
THE UNIVERSITY
*of* EDINBURGH

# Term Selection in Information Retrieval

*K. Tamsin Maxwell*


Doctor of Philosophy

Institute for Communicating and Collaborative Systems

School of Informatics

University of Edinburgh

2016

# Abstract

Systems trained on linguistically annotated data achieve strong performance for many language processing tasks. This encourages the idea that annotations can improve any language processing task if applied in the right way. However, despite widespread acceptance and availability of highly accurate parsing software, it is not clear that ad hoc information retrieval (IR) techniques using annotated documents and requests consistently improve search performance compared to techniques that use no linguistic knowledge. In many cases, retrieval gains made using language processing components, such as part-of-speech tagging and head-dependent relations, are offset by significant negative effects. This results in a minimal positive, or even negative, overall impact for linguistically motivated approaches compared to approaches that do not use any syntactic or domain knowledge.

In some cases, it may be that syntax does not reveal anything of practical importance about document relevance. Yet without a convincing explanation for *why* linguistic annotations fail in IR, the intuitive appeal of search systems that 'understand' text can result in the repeated application, and mis-application, of language processing to enhance search performance. This dissertation investigates whether linguistics can improve the selection of query terms by better modelling the alignment process between natural language requests and search queries. It is the most comprehensive work on the utility of linguistic methods in IR to date.

Term selection in this work focuses on identification of informative query terms of 1-3 words that both represent the semantics of a request and discriminate between relevant and non-relevant documents. Approaches to word association are discussed with respect to linguistic principles, and evaluated with respect to semantic characterization and discriminative ability. Analysis is organised around three theories of language that emphasize different structures for the identification of terms: phrase structure theory, dependency theory and lexicalism. The structures identified by these theories play distinctive roles in the organisation of language. Evidence is presented regarding the value of different methods of word association based on these structures, and the effect of method and term combinations.

Two highly effective, novel methods for the selection of terms from verbose queries are also proposed and evaluated. The first method focuses on the semantic phenomenon of ellipsis with a discriminative filter that leverages diverse text features. The second method exploits a term ranking algorithm, PhRank, that uses no linguistic information and relies on a network model of query context. The latter focuses queries so that 1-5

terms in an unweighted model achieve better retrieval effectiveness than weighted IR models that use up to 30 terms. In addition, unlike models that use a weighted distribution of terms or subqueries, the concise terms identified by PhRank are interpretable by users. Evaluation with newswire and web collections demonstrates that PhRank-based query reformulation significantly improves performance of verbose queries up to 14% compared to highly competitive IR models, and is at least as good for short, keyword queries with the same models.

Results illustrate that linguistic processing may help with the selection of word associations but does not necessarily translate into improved IR performance. Statistical methods are necessary to overcome the limits of syntactic parsing and word adjacency measures for ad hoc IR. As a result, probabilistic frameworks that discover, and make use of, many forms of linguistic evidence may deliver small improvements in IR effectiveness, but methods that use simple features can be substantially more efficient and equally, or more, effective. Various explanations for this finding are suggested, including the probabilistic nature of grammatical categories, a lack of homomorphism between syntax and semantics, the impact of lexical relations, variability in collection data, and systemic effects in language systems.

# Lay Summary

It may seem obvious that in order to improve search, such as the retrieval of documents by a search engine like Google, it helps to automatically teach computers to 'read' and 'understand' language. If we can accurately determine what a query means, and match it to documents that have the same meaning, then we should get more relevant search results.

Meaning is determined automatically by 'natural language processing' (NLP). NLP works by counting linguistic tags assigned to words and phrases, words that appear next to each other, and relations between words. A tag might be 'noun' or 'verb' and there is a relation between '*red*' and '*car*' in the phrase '*red car*'. Tags and relations (called "annotations") help computers to know what text means and are very popular. Their popularity is helped by high quality, free software that automatically annotates text. The popularity and success of annotations can make it seem like they are useful for any NLP task, including search. But search experts found two main problems. First, sometimes annotations do not help and it is hard to identify when this will happen just by looking at a query. Second, annotations are not necessary and there are easier ways to get relevant search results. The easier methods are also more efficient. They are faster and work with a document format that takes up less storage space.

Very good search results are possible with probabilistic combination of different types of annotations and word, or phrase, frequencies in documents. These combinations are more likely to help if queries look like questions that we might ask other people. But in general, it is very difficult to beat search systems that simply count how many times words appear next to each other with any use of annotations.

This dissertation examines whether NLP can improve search by finding 'terms' (combinations of one to three words) that better represent queries. This task is called 'term selection'. I focus on queries that look like questions we ask other people because this is where term selection is more likely to help. However, short queries, such as the ones people usually type into search engines, are also explored. A main contribution of this dissertation is systematic exploration of why people believe annotations will help search performance. I consider whether these beliefs are justified, and use linguistic theory, statistical methods and experiments with search systems to argue that in many cases they are not.

The dissertation is divided into three parts. The first part gives background on search systems, how they use relations between words, and how other people proposed

to select terms. The second part discusses three aspects of what makes a useful ("informative") term in search: linguistic principles, language meaning (semantics), and an ability to distinguish relevant from non-relevant documents for a query. The third part presents two new methods for term selection that deliver search results that are as good, or better, than the best reported results so far. I finish with conclusions about the nature of language processing at web scale, and how we can work around it.

# Acknowledgements

A few people helped to bring this work to fruition and deserve thanks. I am grateful to my supervisor, Jon Oberlander, for his patience, encouragement and guidance. His support was of inestimable value in keeping me on track while giving me the space to pursue my own ideas. I am also indebted to Bruce Croft, my supervisor at the University of Massachusetts, who believed in my ability to achieve something great and changed the course of my PhD for the better. His straightforward and insightful comments will go down in history with a smile. Further thanks goes to Victor Lavrenko, who contributed greatly to the clarification of my ideas with timely feedback, and to Burkhard Schafer, who advised in the early stages of this work and demonstrated with enthusiasm that academia is full of exciting opportunity.

A raised glass goes to my peers and colleagues at the University of Massachusetts, Amherst, and the University of Edinburgh, who made life as a PhD student so thoroughly enjoyable. Their comments, suggestions and tips filled small gaps in my knowledge when I needed a hand, and lightened my burden in more ways than one. Of my colleagues, particular thanks goes to Michael Bendersky for the code to extract collection statistics from an Indri index. To Jeff Dalton for fine dining and an invitation to pitch in on experiments. And to Sam Huston for backing up his interest in this work with the addition of PhRank queries to his weight optimization experiments, enabling comparison with a highly effective weighted sequential dependence model. A warm hand goes to Dan, Alison and Roger who cheerfully worked miracles with compute resources on my behalf. And extra special thanks is reserved for David Fisher, whose witty and insightful responses to questioning made engineering and thesis drafting positively fun.

An unwitting contribution to this dissertation was made by Richard Johansson and Pierre Nugues, whose joint semantic-syntactic parser features in this research. Credit also goes to the authors of the Stanford Dependency Parser, MultiLingua, and Weka. I thank the creators for their generosity in making these resources available.

Finally, in the realms of personal support I would not have made it through without my partner, Gregor, whose understanding and encouragement were pivotal to my sanity and success. I am also thankful to friends who taught me, again and again, to have courage, curiosity and confidence. And to my mum and dad, who encouraged me to keep going and believed in me every step of the way. This is for you.

# Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*author*)

The candidate confirms that the work submitted is his/her own, except where work which has formed part of jointly-authored publications has been included. The contribution of the candidate and the other authors to this work has been explicitly indicated below. The candidate confirms that appropriate credit has been given within the thesis where reference has been made to the work of others.

Work from published papers under joint authorship appears as follows:

1. **Chap 6:** Experiments using human annotated training data, and supporting introductory material appeared at ACL 2013. All work is credited to the candidate. Available at `http://www.aclweb.org/anthology/P13-1050`.

    - K. Tamsin Maxwell, Jon Oberlander and W. Bruce Croft, "Feature-Based Selection of Dependency Paths in Ad Hoc Information Retrieval", in *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL)*, Sofia, Bulgaria, August 4-9, 2013

2. **Chap 7:** All content except experiments with the Croft & Harper model, the Relevance Model, and the expanded PhRank model was presented at SIGIR 2013. All work is credited to the candidate. Available at `http://dl.acm.org/citation.cfm?id=2484096`.

    - K. Tamsin Maxwell and W. Bruce Croft, "Compact query term selection using topically related text", in *Proceedings of 36th international ACM SIGIR conference on Research and development in information retrieval*, Dublin, Ireland, July 28-Aug 1, 2013

# Table of Contents

# 1

# Introduction

*"Language is an elephant, and we are all blind men trying to discover what the elephant is like."*

– Chafe (1968)

## 1.1 Word association in information retrieval

Classic information retrieval (IR) often assumes that words are statistically independent of one another. This assumption is clearly unrealistic since documents are not bags of words, but treating them like bags of words simplifies engineering in IR systems. The word independence assumption avoids difficulties with estimation of word dependence probabilities and any need to develop a single weighting scheme for WORDS and PHRASES[1] (single weighting schemes tend to systematically favour phrases (Gao et al., 2004)).

In practice, an assumption of word independence is also quite effective. The assumption works relatively well in practice because the meaning of phrases can often be interpreted as a function of the meanings of their component words. This is an interpretation of COMPOSITIONAL SEMANTICS for IR. Compositional semantics defines the meaning of a phrase to be a function of the meanings of its parts and the way they are put together. For example, in compositional semantics, the meaning of 'red car' is a function of the meanings of '*red*' and '*car*' and their syntactic relationship (adjective-noun). In IR, the set of documents that is relevant to a query about '*red cars*' can be identified using the intersection of the set of documents that refer to '*red*', and the set

[1] A glossary is included at the end of this dissertation for readers who are unfamiliar with standard terminology in IR and linguistics. Glossary terms appear for the first time in text LIKE THIS.

of documents that refer to '*cars*' i.e. the subset of cars that are red. The combination of probabilities, or scores, assigned to documents on the basis of individual words is used to produce a ranking over documents.

The independence assumption for IR has three basic shortcomings. First, words are not independent in reality; their context determines whether they are more or less likely to occur. Second, some phrases do not have compositional semantics; an assumption of word independence fails to retrieve relevant documents for non-compositional phrases such as '*call letters*' because documents that use these words independently are unlikely to focus on the topic of the phrase.[2] Finally, some phrases do have compositional semantics, but their component words have multiple meanings so an assumption of word independence results in the retrieval of many irrelevant documents. Users are not interested in retrieving documents containing individual words, but documents containing particular senses of words and concepts (Krovetz, 1995, 1997). For example, '*record*' in the phrase '*record online*' might refer to information (a document of record) or audio/voice recording.

For these reasons, research on models and techniques that go beyond a word independence assumption have a long history in IR and many modern IR systems incorporate some notion of statistical dependence or SYNTAGMATIC word association (e.g. ngrams). Models that incorporate word associations are thought to retrieve documents with more PRECISION than models that assume word independence because they more closely specify document content. When applied in conjunction with methods that improve RECALL, such as STEMMING (removal of inflections such as suffixes), word associations also help to avoid the reduction in precision commonly seen at higher levels of document recall.[3] Krovetz (1995) gives an example in which '*department*' is stemmed to '*depart*'. Normally, such inappropriate stemming would retrieve many irrelevant documents, but it has very little effect in the context of the phrase '*justice depart*'. Word associations constrain language context and thereby help to address certain challenges of semantic interpretation including word ambiguity and content specification.

[2] '*Call letters*' refers to the identifier, typically an acronym, for a television or radio station. The meaning of the phrase cannot be inferred easily from the multiple meanings of '*call*' and '*letters*'. As such, documents containing independent instances of '*call*' and '*letters*' are unlikely to refer to radio and television stations.

[3] All relevant documents can be retrieved (100% recall) by simply returning all documents in a retrieval collection, but the retrieved set will have low precision.

### 1.1.1 Semantics and relevance

The impact of modeling word associations in IR is evaluated with respect to standard metrics of precision and recall. However, computation of these metrics depends on a definition of RELEVANCE. Relevance typically assumes a binary relation between a document $D$ and a user REQUEST, represented as a query $Q$ (Lavrenko, 2004), where a request communicates an INFORMATION NEED to another human being, and a query is a formulation used by a search engine as a REPRESENTATION of that request (Mizzaro, 1998).[4] IR literature often makes no distinction between a request and a query, and the relevance of $D$ to an information need is assumed if there is substantial overlap of the semantics for $D$ and $Q$. Yet in reality, relevance is contingent on both the semantic overlap between a request and a query and the interpretation of documents.

Unfortunately, it is not easy to infer the semantic overlap of a request and a query because the number of semantic interpretations in any formal analysis is likely to be exponential (Blackburn and Bos, 2003). Even a request with one word can have multiple meanings. By consequence, it can be argued that a standard definition of relevance in which "$D$ is relevant if there is a substantial semantic overlap between the representations of $D$ and $Q$" (Lavrenko, 2004) is not well defined and it is insufficient to evaluate document retrieval with respect to such a definition. In practice, rather than throw out existing definitions, this argument simply highlights the potential importance of semantics in IR for assessing the utility of specific word associations and words. The selection of desirable word associations from a request demands consideration of two criteria: the accuracy with which associations capture language semantics, and the ability of those associations to discriminate relevant documents according to a standard definition of relevance. In other words, there is a need for both semantic representation (the static interpretation of request semantics, in this case limited by selection of word associations, see Section 4.1), and discriminative ability. TERMS that meet both these criteria are defined in this dissertation to be INFORMATIVE.

### 1.1.2 Linguistics and statistics

Simplifying greatly, there are two ways to identify desired word associations. These can be categorized as statistical or LINGUISTIC (Fagan, 1987). This distinction natu-

[4] There are a many definitions of relevance that consider additional or alternative factors such as user preferences, prior knowledge, uncertainty about an underlying information need, differences in task definition, document like-ability, and whether similar documents have already been judged by a user for relevance (Lavrenko, 2004; Mizzaro, 1998).

rally arises given the separate academic disciplines of mathematics and linguistics that contribute to the detection of word relationships. As applied in IR, the approaches differ largely in their efficiency and the degree to which they mark the details of relationships. Statistical methods leave word relationships unspecified, while the reverse is usually true for linguistic methods. Systems that use linguistic processing tend to focus on accurate description of language complexity. For example, '*information*' and '*retrieval*' in the phrase '*information retrieval*' can have a noun-noun, governor-dependent and/or specific semantic relationship. In addition, linguistic theories often assert some alignment between syntax and semantic interpretation (Koenig, 2005) (although not the strict sense of MODEL-THEORETIC SEMANTICS (Partee, 2001)). An advantage of a linguistic approach is that identified word associations can probably be assumed to represent the semantics of the request.

In contrast, a statistical approach aims to capture patterns in data rather than semantics. Statistical retrieval models fit language data well and are shown to be highly effective. Probability theory gives a rich view of language structure and use (Manning, 2007), and a mathematical paradigm lends itself to the probabilistic detection of spurious word dependencies. This contrasts with the categorical approach frequently taken by SYMBOLIC LINGUISTIC models of SYNTAX: either two words participate in a given relation or they do not. Indeed, syntacticians are sometimes criticized for being overly sensitive to categorization requirements, treating "the least counter-example as a fundamental flaw" (Grefenstette, 1998). Mathematical approaches are also often highly efficient, and scale well to real world IR systems. A substantial amount of research on statistical models concentrates on improving practical implementations.

A statistical approach to IR is often considered preferable to one inspired by linguistics. Yet despite many benefits, a statistical approach has two major disadvantages. First, it accounts for observed data, but does not require the resulting model to be interpretable by humans or bear an obvious relation to accurate linguistic generalization. This can make it difficult to recognise and correct systematic retrieval errors. It also discards an opportunity for interactive query tuning that can improve search performance. Interpretable queries generated by linguistically inspired approaches facilitate amendment and help users to recover when a system fails to retrieve desired documents. They can be particularly useful in domains such as law where search transparency is vital.

More importantly, it is not immediately obvious how to focus a mathematical approach to optimally select informative word associations. Brute force is capable of finding optimal solutions, but is simply impractical given that word associations are

specific to each query. This difficulty is evidenced by more than 50 years of experimentation with word association: if brute force was sufficient, the solution would be clear by now. Machine learning provides an efficient framework for learning word associations, but machine learning algorithms are not always guaranteed to find the optimal solution. Moreover, they can be confused by uninformative, unreliable or redundant features. As has been pointed out, the common weakness for learnt approaches is the lack of guidance on how to select features (Zhai, 2008). The wrong selection can reduce the separability of relevant and irrelevant documents, and make finding a good solution less probable. The critical step is selection of features that constitute the most profitable bias for learning.

Linguistic features enter the frame because they can supply a profitable bias for statistical learning at the same time they provide some basis for semantic interpretation. Non-statistical, rule-based processes operating at a small scale, such as syntactic interactions between individual words, produce patterns in language and thus make useful features. This forms a basis for modeling language in large scale IR systems, and means that systems built up from linguistic interactions at the sentence or word level can perform exceptionally well. In this way, linguistics is related to IR as theory to evidence. It can guide development, provide a principled way to understand the consequences of feature selection, and facilitate insight into when and how word associations are likely to aid retrieval.

However, any benefit from an application of LINGUISTIC REPRESENTATIONS, such as grammatical categories and word relations, depends on correct, or appropriate, assumptions about the organisation of language and alignment with semantic interpretation. Just like any other feature source, linguistics can supply a misleading bias for learning if it does not describe key aspects of language with respect to a particular task. Consequently, it may not consistently deliver significant improvements in retrieval EFFECTIVENESS compared to techniques that use no linguistic knowledge.

The application of appropriate linguistic features is made difficult by the fact that there are many competing linguistic theories. In addition, natural language processing techniques (NLP) can be complex and incur a substantial processing cost, making them impractical for large-scale applications. It can also be argued that linguists collect evidence to determine the principles governing production and understanding of language, while researchers in IR collect evidence to uncover the principles that govern document relevance. By consequence, linguistics does not necessarily reveal anything of practical importance about document relevance.

### 1.1.3 Linguistics in IR

To date, empirical evidence on the value of linguistics for IR is inconclusive (Brants, 2004; Hui, 1988; Lewis and Jones, 1996; Smeaton, 1999; Spärck Jones, 1999). A number of widely read papers on the actual and potential contributions of automated NLP to retrieval in the past 20 years variously concluded that:

- "Document retrieval is not an ideal application for NLP" (Brants, 2004);
- "Linguistically Motivated Indexing is not needed for effective retrieval" (Spärck Jones, 1999);
- "When syntactic methods are used for the generation of content-identifying phrases, the retrieval results are often discouragingly poor" (Salton and Smith, 1990);
- "The impact of NLP on information retrieval tasks has largely been one of promise rather than substance" (Smeaton, 1999).

The problem seems to be that in many cases retrieval gains made using language processing components, such as part-of-speech tagging and shallow parsing (chunking) are offset by significant negative effects. This results in minimal positive, or even negative, overall impact when compared to approaches that do not use any linguistic or domain knowledge (Brants, 2004; Lewis and Jones, 1996; Song and Croft, 1999). It might be concluded that language processing is not well suited to IR, but in the early 2000s when these conclusions were made, available evidence was almost entirely derived from the application of one theory of language that may indeed be problematic in an IR context: PHRASE STRUCTURE THEORY.

Phrase structure grammars emphasize, and are designed for, those aspects of language that adhere to a principle of compositionality: that the meaning of a phrase is a function of the meaning of its parts and the way they are put together syntactically. Phrase structure grammars use a finite set of rules that compose GRAMMATICAL CATEGORIES, including sentences, nouns, verbs, ADVERBS, ADPOSITIONS (preposition or postposition), and their phrasal projections (e.g. noun phrases, verb phrases). For example, a sentence ($S$) can be composed of a noun phrase ($NP$) followed by a verb phrase ($VP$):

$$S \rightarrow NP\ VP$$

A core problem with phrase structure theory for IR is that important word relations can cut across grammatical categories. In addition, discrete category assignments are prone to error (see Chapter 3).

Statistical techniques can overcome specific limitations of phrase structure theory and are highly effective (Bendersky and Croft, 2008; Lease et al., 2009; Park et al., 2011; Xue et al., 2010), but not all linguistic theories share the same limitations. For example, several recent discriminative approaches to retrieval also include features generated by DEPENDENCY GRAMMARS. Dependency theory provides an alternative interpretation of language structure that is nonetheless compatible with phrase structure theory. It describes governor-dependent relations, called DEPENDENCIES,[5] that realize a primarily semantic notion of HEADEDNESS (e.g. a PREDICATE, often a verb, is head of its ARGUMENT, such as a direct object). These dependencies replace compositional aspects of language as the focus for syntactic representation and enable dependency grammars to avoid some of the problems associated with phrase structure.

In addition, statistical approaches aim to capture patterns in data. LEXICALISM is a theory of language that also focuses on patterns in data. It gets its name because it is 'of or relating to words' as opposed to other linguistic units, such as MORPHEMES, PHONEMES and grammatical categories. It assumes that meaning in language is formed irregularly and more or less directly grasped without consideration for how the parts are assembled. It focuses on the relations between words, not the structure of language, and often uses statistical techniques to identify patterns in language. Thus, statistics are not a 'silver bullet' for linguistic shortcomings. On the contrary, lexicalism may go some way to explain the success of statistical techniques.

To put it simply, there is no clean separation between the value of linguistics for IR and the value of statistical approaches in IR. Moreover, a separation that privileges the role of statistics for IR ignores potentially valuable insights from linguistics. It also disempowers researchers seeking satisfactory intuitions about untested IR models that leverage word associations. A better understanding of how linguistics applies to IR can help to determine viable directions for future research and explain past empirical results for the application of NLP to search.

[5]Notice that this is a different definition of dependency than the one used in IR. In linguistics, a dependency is a syntactic or semantic relation between a head (or governor) and a dependent word, morpheme or phoneme. In IR, a dependency is a statistical relation between two words or phrases. The syntactic or semantic nature of this relation is unspecified.

## 1.2 Motivation

In this dissertation, I explore the relationship between language structure, semantics and discriminative ability for word associations in IR. The work is motivated from an IR perspective, and focuses on the selection of query terms given a VERBOSE information need, where a term is any text unit composed of *one or more words*.

Verbose queries express a complex or specific information need, either in natural language or a sequence of more than four content-bearing words. They constitute a growing proportion of all search activity on the web. However, the ranking algorithms of most modern search engines are designed to operate on short keyword queries with one to three words (Croft et al., 2010). Compared to keyword queries, verbose queries contain words that make sense in context but may retrieve irrelevant documents. They are also more likely to contain multiple query facets, so fewer documents meet the search criteria. These characteristics make it difficult for search engines to rely effectively on user click-through information, clickable text in hyperlinks (anchor text), and popularity signals like PageRank (Page et al., 1999) for ranking documents (Clarke et al., 2011). By consequence, the retrieval performance of ranking algorithms tends to deteriorate when applied to verbose queries compared to keyword queries (Balasubramanian and Allan, 2009; Bendersky and Croft, 2008; Croft et al., 2010; Kumaran and Allan, 2007). Paradoxically, queries with less information often perform better than verbose queries for the same information need (Bendersky and Croft, 2008).

A boost in the number of relevant documents retrieved by verbose queries can be achieved by better query representation (well-informed decisions about *what* text units to use) and more accurate IR modeling (well-formulated models of *how* to use text units). This dissertation focuses on TERM SELECTION (*what* units to use) for representation of the most essential aspects of query meaning. The goal is to choose terms that contain only words from within a query itself, where any relationship between the words is unspecified, even if it is leveraged during the process of term selection. Terms are applied in a model inspired by Lewis and Jones (1996) in which terms "can refer to concepts with a range of complexity, while the loose coupling among these items permits efficient and flexible matching". This model describes *how* units are used. Specifically, a source query supplemented with key terms from the query is matched against documents using simple, underspecified word proximities. No document parsing is used even though a query may be parsed to assist with the identification of terms. The choice not to parse documents is based on an assumption that "request develop-

ment...matters much more than document characterization" (Lewis and Jones, 1996).

I choose to work primarily with verbose queries because their sentence-like structure enables the application of both statistical and syntactic methods. Selection of informative terms also becomes more critical as the number of potentially noisy terms increases. Nevertheless, one method proposed in this dissertation also achieves consistently high performance with keyword queries, taking a positive step towards solving the grand challenge for IR: engineering strong performance for all queries.

## 1.3 Task definition

The task in this dissertation is term selection from a complete set $P$ of all possible terms for a query, where a term is composed of one or more words. For example, a representative term for the query '*Is the disease of Poliomyelitis (polio) under control in the world?*' (Robust04 #302) is '*polio control*' with an unspecified dependence between *polio* and *control*. Other terms include '*polio*', '*disease polio control*' and so on. The complete set of terms for a query is the power set $\wp(Q)$ of all content-bearing words in the query, where a content-bearing word is any word not appearing in a pre-defined list of highly frequent 'stop' words (Allan et al., 2000). The aim of query-based term selection (henceforth, simply 'term selection') is to identify a small subset of terms in $\wp(Q)$ that maximizes the retrieval effectiveness of $Q$.

Term filtering is the inverse process of term selection. Term selection identifies a subset of terms in $\wp(Q)$ to retain, and term filtering identifies a subset of terms in $\wp(Q)$ to discard. In either case, a process divides $\wp(Q)$ into two disjunctive subsets, only one of which is used for further processing and query reformulation. Term selection and term filtering can dramatically improve the efficiency of queries, as well as retrieval effectiveness, by focusing on the most important information in a query. Specifically, they select for informative terms.

The techniques in this dissertation involve two stages: one for the identification of candidate terms, and a second stage for selection of terms from a candidate pool. This might be interpreted as a process of term selection followed by a process of term filtering. I will prefer the terminology of *term selection* (or equally, term identification) in all cases since the final retained set of terms is much smaller than the discarded set. This terminology simplifies discussion.

## 1.4 Thesis statement

This dissertation answers the question: what types of word associations are most effective for IR? Specifically, it investigates the following thesis:

*Language structure helps to identify word associations that improve search effectiveness more than associations identified using simple word adjacency.*

## 1.5 Contributions

This dissertation is the most comprehensive study to date on the utility of linguistic methods for ad hoc IR. Contributions can be organised into three categories:

- **Linguistic analysis:** Chapter 3 provides insight into major linguistic theories and their ability to identify informative word associations for IR. In the past, it has been assumed that "subtle differences between competing [linguistic] theories are likely beyond what we can effectively detect" (Lease, 2007). However, such an assumption places faith in conventional methods of linguistic interpretation, instead of critical analysis. Three linguistic theories are presented, along with their fitness for application in IR, the context in which their application evolved, and related research trends in IR. These theories are phrase structure theory, dependency theory and lexicalism. To my knowledge, a similar review of linguistics for IR has not been published before, and constitutes a unique contribution to the field.
- **Conceptual exploration:** Chapters 4 and 5 investigate the conceptual relationships between word association, semantics and discrimination. Chapter 4 explores the extent to which word associations identified by linguistic theories align with request semantics, both in theory and in practice. Chapter 5 presents a thorough empirical evaluation of the ability of both automatically detected and user-nominated word associations to discriminate between relevant and non-relevant documents. Based on this evidence, I conclude that surface syntax is not optimal to identify query terms. Word associations identified by various methods are not strongly related to either request semantics (as operationalized by user nominated terms) or discrimination. In addition, terms that represent request semantics are not strongly related to discrimination. These conclusions

are predicated on the assumption that user-nominated terms encapsulate query semantics and they ignore factors such as user distraction and individual variation in memory, imagination, emotion and social factors. In addition, measures of discrimination are based on the performance of one implemented search system. Nevertheless, the systematic study of the relationships between word association, semantics, and discrimination is a unique contribution to the field. In addition, a theoretical explanation is given for these results for grammatical English. This provides scope for the conclusions and facilitates their extension to similar tasks that focus on text topicality or similarity.[6]

- **Method evaluation:** Two novel methods for term selection in IR are proposed and evaluated. The first method focuses on terms that are both discriminative and strongly aligned with the semantics of a request (Chapter 6). It uses dependency structure and is effective but computationally intensive. Conversely, the method in Chapter 7 uses simple word adjacency and term frequency information to select key terms that are not necessarily either words or syntactic phrases. It achieves performance that is as good, or better, than the best published results with verbose queries on multiple evaluation sets, and highly competitive performance for keyword queries. An important factor in its success is a novel proposal for query biased pseudo relevance feedback, in which feedback is used to select terms from within a query rather than expand to novel terms or weight existing terms. This helps the method to produce unweighted queries that are both concise and interpretable by users, without reliance on linguistic processing of queries or documents. Given that existing research in text-based IR is highly developed and typically improves search performance via complex optimization procedures and language processing, this is a substantial achievement. It demonstrates that a succinct representation of verbose queries can be as effective as long reformulations using term distributions and term weighting.

- **Conclusions for NLP in IR:** General conclusions about the value and role of language processing in IR are presented. In Chapter 8, I contend that word associations only help to identify relevant documents in ad hoc IR when interpreted using statistical or probabilistic techniques. This is supported by examples from

[6] Keyphrase detection and topic modelling are similar to term selection for ad hoc document-based IR and it may be possible to generalize conclusions in this dissertation for these tasks. Alternative conclusions may apply for sentence retrieval, question answering and summarization, as these tasks are more focused on word and term relations, types, and organization.

literature and further experimentation. Chapter 9 goes on to present the conclusion that for ad hoc IR, syntactic analysis is not necessary. Syntax does not help to identify terms that are significantly better than terms that leverage simple word proximity. Given a large text collection, for every syntactic word association there is a word association that carries at least as much information about topical relevance and can be identified from unannotated text. This does not claim that linguistics is a failure in IR, or that word proximity or bigrams are always as informative as syntax. Rather, it highlights that syntax captures essential word relationships, but does not capture *all and only* essential word relationships. Syntax may still be useful for IR tasks focused at the sentence-level (such as question answering), or where fine-grained analysis is critical to success.

## 1.6 Scope

This dissertation explores the interdisciplinary issue of word association and term selection for IR. Related topics include dependence models for IR, query expansion, query transformation and reduction, term weighting, query segmentation, question answering and verbose query handling, all of which have an extensive literature (see Chapter 2). In-depth discussion of all these topics is impractical, and a much larger project than a single thesis. I necessarily limit myself to research highlights in each area, and provide references as a starting point for further exploration by the interested reader.

Inspiration for my methods of term selection originate in linguistics. I draw from a limited number of viewpoints in 20th century linguistics on language syntax and semantics, and restrict discussion to linguistic theories that have been applied in IR. In particular, it must be noted that this dissertation is not an attempt to advance the field of linguistics and does not directly address the field of semantic theory. Linguistic literature is summarized to frame the discussion of word association and relevance for IR, and skims the surface of a vast body of work. I extract the most pertinent information to help readers understand why certain query terms may be more effective than others. For a more in-depth exploration of related work in linguistics, please see the referred literature.

Finally, I address metrics for word association, which are broadly applied in many computational linguistics tasks. Other language processing tasks, such as summarization and information extraction, also resemble term selection, but with one notable exception these tasks are not addressed so as to constrain this dissertation to a rea-

sonable size. The exception is the summarization task of keyphrase extraction. This resembles term selection so closely that it should not be ignored. Moreover, a novel term selection technique proposed in this dissertation is derived from an algorithm for keyphrase extraction.

Overall, this dissertation addresses the value of linguistics for IR, and contributes to an analysis of the strengths of structural language features for the selection of word associations for IR. I do not inspect all possible types of word associations that might be effective or attempt to prove the superiority of any particular linguistic approach over another. The dissertation is rather a general resource for those interested in the intersection of language processing and information retrieval.

## 1.7 Overview of the thesis

This dissertation is divided into four parts, preceded by a background to IR. Readers familiar with IR may prefer to skip to Chapter 3 for novel contributions. The first part discusses three major linguistic theories and their fitness for application in IR, along with relevant history and related research (Chapter 3). Readers familiar with linguistics may choose to skim this Chapter, attending to the Sections on the suitability of each theory for IR. Part two is composed of Chapters 4 and 5. It provides a conceptual exploration of three factors that may influence term selection in IR: semantics, word association and document discrimination. Part three proposes and evaluates two novel methods of query term selection, the second of which demonstrates significant improvement over the best competing query reformulation models (Chapters 6 and 7). Finally, in part four, I present general conclusions about the value and role of language processing in IR, including supplemental experiments in Chapter 8 and discussion in Chapter 9. This is followed by directions for future work. The Chapter breakdown is as follows:

- **Chapter 2** reviews important concepts and terminology in IR that complete an understanding of the challenge of term selection, how it is applied, and how others have approached it before. Fundamental concepts in IR are reviewed, including general frameworks for term selection, types of queries and IR tasks, document collections, and models of word association for IR.
- **Chapter 3** provides background for three linguistic theories that construct different models of language: phrase structure theory, dependency theory and lexical-

ism. It addresses the linguistic principles embodied by terms and gives readers without a background in linguistics a basic understanding of relevant concepts in language representation. Discussion covers the history of linguistics in search and key theoretical points that bear upon term selection in IR.

- **Chapter 4** suggests that semantic relationships between words are one aspect of term informativeness, and that semantics aligns with language structure. It provides a framework for the categorisation of word association methods into three classes with characteristic behaviours in IR systems–syntagmatic, syntactic and statistical methods–and explores the ability of these classes to select user-nominated terms. User-nominated terms are used as a gold standard for the semantic interpretation of queries.
- **Chapter 5** evaluates the ability of word association methods to discriminate between relevant and non-relevant documents. In addition, combinations of terms, and combinations of word association methods, are explored to determine the degree to which discrimination is a property of individual terms and term sets.
- **Chapter 6** explores a novel approach to term selection that leverages the semantic properties of catenae. Catenae are a flexible, simplified notion of paths in a dependency graph that were previously proposed as a precondition for the semantic phenomenon of ellipsis. The approach uses catenae to identify candidate terms, and a supervised machine learning technique to discriminate between informative and uninformative candidates.
- **Chapter 7** explores an unsupervised approach to term selection with a novel term ranking algorithm called PhRank that uses no linguistic information. PhRank ranks terms by representing query context as a co-occurrence network with discriminative weights. Queries incorporating top ranked terms achieve up to 14% performance improvement compared to highly competitive IR models.
- **Chapter 8** uses extensions to PhRank to demonstrate contrasting ways that language features are applied to identify word associations and select discriminative terms. Evidence is used to outline a simple, coherent guideline for the application of language features in novel term selection techniques. Specifically, language features make their largest contribution to search effectiveness when incorporated in statistical or probabilistic techniques.
- **Chapter 9** provides conclusions and directions for future work, including some collaborative experiments.

Conventions in this dissertation include the use of italics to indicate some key points and terminology. A glossary is included at the end of this dissertation for core terminology. Words and terms in the glossary will appear for the first time in text LIKE THIS.

# 2

# Background to Information Retrieval

This Chapter reviews important concepts, terminology and background pertinent to a discussion of term selection strategies in IR. Topics include general frameworks for term selection, types of queries, search tasks, document collections, and models of word dependence. Readers familiar with IR may prefer to skip this Chapter and head straight for a discussion of linguistic principles in Chapter 3.

## 2.1 Ad hoc information retrieval

Most research in IR refines the effectiveness of queries using standard test collections such as the ones made available by the Text REtrieval Conference (TREC). Every year, TREC designs a number of 'tracks' that focus on specific aspects of retrieval. A standard search scenario is *ad hoc* retrieval, in which an IR system is given a previously unseen query and required to produce a ranked list of documents from a static collection. The ranked list must be based entirely on the query, any relevant resources, and features of the document collection. No prior knowledge about the user, the information need, or the query context is available. The search topic is a transient information need (Manning et al., 2008) such as navigation to a particular web page, or search for information on a specific topic.

For each IR track, TREC provides a static set of queries developed for a specific document collection. Human relevance judgments are available for all documents in the collection that are likely to be retrieved by one of the queries. This enables the effectiveness of retrieval strategies to be empirically evaluated and compared, and helps to identify valuable directions for future research.

For modern probabilistic retrieval systems, the goal of ad hoc retrieval is to return a

ranking over documents in order of decreasing probability of relevance. The probability of a document being relevant is independent of the probability of other documents in the collection being relevant. The effectiveness or performance of an IR model, or IR technique, refers to its ability to discriminate relevant from non-relevant documents in order to achieve this goal. I will use the terms *effectiveness* and *performance* interchangeably.

### 2.1.1 Keyword and verbose queries

For ad hoc retrieval tracks, TREC provides static sets of *title* queries. Title queries are *keyword* queries, which means they are typically sequences of three or fewer content-bearing words without obvious syntax. Ad hoc TREC queries are also associated with a *description* topic and a *narrative*. A description topic is a natural language description of an information need that is usually one or two sentences long. A narrative is a short paragraph of text that details an information need and may specifically identify what is *not* relevant to a query.

Description topics contain sentences, so they are often used to study the behaviour of *verbose* queries in IR. Verbose queries express a complex or specific information need and include *long keyword* queries and *natural language* queries. Long keyword queries have five or more content-bearing terms with no clear syntax. They typically appear in commercial query logs, and constitute around 10% of all queries (Bendersky and Croft, 2009). These queries may be an attempt by users to adapt natural language to fit the abilities of search engines. Search engines are optimized to perform well on short keyword queries (Lease, 2010) so users tend to write keyword queries even when an information need can be more accurately expressed in natural language and/or keyword queries do not retrieve the desired documents.

In contrast, natural language queries are grammatical sentences. They may feature in voice-activated search, including spoken queries to mobile devices, and where users expect other users to answer their questions, such as web question and answer (Q&A) sites. Although natural language queries make up a small proportion of all queries, an average query length of 30 words on Q&A sites such as *Wondir* and *Yahoo! Answers* suggests that users may be willing to write complex questions to have their information needs satisfied (Croft et al., 2010).

An example of a TREC title (keyword) query, long keyword query, natural language query, and TREC description topic and narrative are given below. Note that the

standard IR terminology does not distinguish between *requests* and *queries*. A request is an expression of an information need before it is transformed into a query for submission to a search engine. At a practical level, all but the keyword query below are requests. I will follow standard practice and use 'query' interchangeably to refer to a 'request', but not vice versa:

- **Title (keyword) query**: '*schengen agreement*'
- **Long keyword query**: '*signatories schengen agreement border control implementation*'
- **Verbose natural language query**: '*Signatories of the Schengen agreement to eliminate border controls in Western Europe, and their actions.*'
- **Description topic**: '*Who is involved in the Schengen agreement to eliminate border controls in Western Europe, and what do they hope to accomplish?*'
- **Narrative**: '*Identify the actions of signatories of the Schengen agreement such as: measures to eliminate border controls (removal of traffic obstacles, lifting of traffic restrictions); implementation of the information system data bank that contains unified visa issuance procedures; or strengthening of border controls at the external borders of the treaty area in exchange for free movement at the internal borders.*'

A distinction is usually made in IR literature between natural language queries and *questions*, even though the former often take the form of a question in the linguistic sense. A question expresses an information need in question-answering (QA) tasks and seeks a specific, small chunk of information such as found in a sentence or phrase. In contrast, a query aims to retrieve relevant documents that contain desired information, or are 'on topic'. For this reason, natural language queries are typically referred to as verbose queries, or description topics. I will also use this terminology.

Verbose queries can be transformed into long keyword queries by the removal of STOPWORDS and/or STOP STRUCTURE. Stopwords are highly frequent words that appear in many documents, irrespective of the topic of a text, e.g. {*with, the, do, if, he*}. The length and content of stopword lists vary, but a small list contains around 37 words. A more comprehensive list contains around 420 words. The stoplist used in this dissertation contains 418 words as applied in the INQUERY search engine (Allan et al., 2000) (see Appendix A). For TREC description topics, a supplemental list of 18 words is applied that identifies words commonly used to frame TREC topics e.g. {*describe,*

*information, document*} (Allan et al., 1995) (see Appendix A). For example, TREC description topics often begin, '*Find documents that describe*', or '*Find information about*'. Like stopwords, such stop structures are phrases that do not provide any information about the topic of a text. They are typically used in queries, and often begin at the first word (Huston and Croft, 2010). An example stop structure is, '*Can any one help me out with a*'.

Finally, complete documents themselves can be considered a form of very verbose query. Retrieval using a long text as a query is known as query-by-document, or *associative document retrieval*. Examples include several legal search tasks, such as patent retrieval and invalidity search (Fujii et al., 2007). Patent retrieval poses real challenges for search engines (Iwayama et al., 2000). Extensive use of highly generic language, acronyms, technical terminology and novel words obfuscates document semantics, while a demand for high recall usually comes at the expense of precision.

### 2.1.2 Question answering

Question answering (QA) is concerned with the automatic retrieval of answers to natural language questions that are fact-based and extractive, rather than answers that require the generation of new knowledge. A pipeline architecture is typically used with modules for question analysis and answer analysis. A question analysis module reformulates a question into an ad hoc query for relevant document passages. A better reformulation retrieves more relevant passages, so techniques that improve verbose query effectiveness, such as term selection, can also improve overall QA performance provided they do not sacrifice recall for the sake of precision. Recall is important in QA because lexico-syntactic features of a query are used to identify answers. This means that the way text is written makes some answers easier to identify than others.

The process of QA typically begins with categorization of a question focus e.g. factoid, list, relationship, or definition.[1] The focus is often a named entity, such as a number, person or date, and identifies the desired answer type. Dependencies between entities in a question are also considered, although the QA community generally concentrates more on answer analysis and extraction than question analysis (Prager, 2006). Candidate passages in documents are identified using ad hoc retrieval and assigned an

[1] In the TREC QA track that ran from 1999-2007 (Dang et al., 2007), questions were classified into four types: *factoid* questions can be answered by a few words, often just a noun phrase; *list* questions are answered by a set of entities, or factoids; *relationship* questions define a relationship by which one entity affects another; and *other* questions are correctly answered by any factoid about an entity. *Definition* questions have also been used in place of *other* questions (Voorhees, 2005).

answer type that is checked for semantic alignment with the expected answer type. Passages that do not contain the expected answer type are discarded, and remaining passages are normalized in preparation for passage ranking (Pasca and Harabagiu, 2001). Ranking compares the syntactic or semantic structures of the question and candidate answers and identifies passages from which answers are extracted. Finally, optional postprocesses are performed, including answer merging and verification. These rely on answer redundancy or sense-checking against an external resource (Prager, 2006).

A common approach to answer selection and ranking compares the dependency parse for a document sentence and a question. Specifically, a matching function compares a dependency path between any two stemmed terms $x$ and $y$ in a question $A$ with any dependency path between $x$ and $y$ in a sentence $B$. The match score for $A$ and $B$ is computed over all dependency paths in $A$, and used to rank or select sentences. For example, Lin and Pantel (2001) present a method to derive paraphrasing rules for QA by analyzing paths that connect two nouns; Echihabi and Marcu (2003) align all paths in questions with trees for heuristically pruned answers; Punyakanok et al. (2004) define a tree edit distance between questions and answers that accounts for all relations between words; Shen et al. (2005) consider a constrained set of relations between noun phrases (NPs) and verbs, and between pairs of NPs, in a simplified dependency structure; Cui et al. (2005) score answers using a variation of the IBM translation model 1 for paths between open class words that are not in the same NP or verb phrase (VP); Wang et al. (2007) use a quasi-synchronous machine translation model to map all parent-child paths in a question to any path in an answer (a one-to-many mapping); and Moschitti (2008) explores syntactic and semantic kernels for QA classification. More recently, Heilman and Smith (2010) present a novel tree edit distance algorithm, and Surdeanu et al. (2011) learn to rank non-factoid answers for QA using many features including parent-child dependency relations. Alternative approaches use term frequencies with PROXIMITY MEASURES, lexical templates for shallow patterns, or logical analysis. Answer ranking with machine learning can also apply diverse features including semantic hypernyms and synonyms (Surdeanu et al., 2008).

Detailed overviews of QA can be found in Prager (2006) and Kolomiyets and Moens (2011). In general, analysis of both questions and answers relies heavily on language processing techniques such as parsing and named entity recognition (NER). Word sense disambiguation, COREFERENCE RESOLUTION and other techniques may also be applied. For fact-based questions, slightly longer questions tend to provide more accurate answers (Croft et al., 2010), but the most descriptive questions are also the most

difficult. The probability of identifying a direct match for a complex question structure in a single passage is quite low. In addition, effective and reliable methods for the reduction or decomposition of long questions are still being developed (Prager, 2006). The work in this dissertation may contribute indirectly to this goal.

## 2.2 Models of word association

Text processing methods can be leveraged prior to search in order to predict or select multiword terms that better represent an information need. However, much research in IR focuses on understanding the behaviour of IR systems, rather than preprocessing techniques that are independent of IR systems (Spärck Jones and Tait, 1984). As such, foundational work on term selection for IR is inextricably linked to the development of IR models that go beyond a word independence assumption.

IR models can incorporate word association in the blueprint for the retrieval process, either by restriction of query terms or by application of statistical data to estimate word dependencies in queries and documents. IR models are typically less discerning in the identification of informative word associations than query-based techniques because they identify *all* associations of a certain class, such as bigrams or governor-dependent relations, rather than only the most informative associations selected using multiple discriminative features. Nevertheless, these models are part of a large body of prior work on word association in IR.

This Section presents major approaches to IR modeling, with a focus on adaptations that incorporate word dependence. Vector space models, probabilistic models, inference networks and linear feature-based models are covered. Vector-space models were the focus of most IR research in the 1960s and 1970s. They are typically less effective than modern alternatives but are still in use, in part for their simplicity and intuitive appeal (Croft et al., 2010). Probabilistic models were introduced in the late 1970s and remain highly competitive and widely used today. A recent systematic comparison of proximity-based dependence models found that a variant of the probabilistic model BM25 can sometimes outperform all other models tested (Huston, 2013). Inference networks are not widely applied, but have the advantage of flexibility. They can incorporate many diverse types of evidence about document relevance and are used to implement linear feature-based models. Linear feature-based models are often state of the art, and weighted versions can consistently outperform competing alternatives (Huston, 2013).

### 2.2.1 Vector space models

The classic vector space model (Salton, 1971) treats queries and documents as points in semantic space. Points that are close together in this space are semantically similar, and documents are sorted in order of increasing distance between the points that respectively identify a query and each document in a collection (decreasing semantic similarity) (Turney and Pantel, 2010).

The model divides search into two stages: indexing and retrieval (Zhai, 2008). In the original vector space model, index terms are mutually independent words, and each word is associated with a TF.IDF weight. Documents are represented by a vector $\vec{d}_k$ of index words, and each query is represented by a corresponding vector $\vec{q}$ of query words. A document vector is a linear combination of word vectors, and the set of all word vectors in the document collection generates the document space. During retrieval, a similarity coefficient is used to approximate relevance of a document vector to a query vector. The similarity coefficient might be the inner product, a function of the angle between the vectors, or a function of a projection of the vectors (Salton et al., 1975). The cosine measure is commonly used:

$$cos(\vec{d},\vec{q}) = \frac{\sum_{j=1}^{n} d_j q_j}{\sqrt{\sum_{j=1}^{n} d_j^2 \sum_{j=1}^{n} q_j^2}} \ . \tag{2.1}$$

Phrases and other dependent word units can also represent documents during indexing. Salton and McGill (1983) suggested the use of multi-word units, or compound terms, in addition to individual words, and explored indexing with phrases determined by statistical word co-occurrence. Terms identified by statistical and syntactic rules were also evaluated for the classic vector space model by Fagan (1987). However, syntactic phrases were deemed to be too low frequency to make good document identifiers (Lewis, 1992). Moreover, it was also shown that retrieval performance using syntactic phrases was comparable to performance using less complex, and therefore preferable, statistical word associations (Salton, 1989).

Variants on the classic vector space model also incorporate word dependence. In the generalized vector space model (GVSM) (Wong et al., 1985) a document vector is a linear combination of $2^n$ term vectors, and the cosine similarity takes correlation between terms $t_i$ and $t_j$ into account using the dot product between $\vec{t_i}$ and $\vec{t_j}$ as follows:

$$cos(\vec{d},\vec{q}) = \frac{\sum_{j=1}^{n}\sum_{i=1}^{n} d_j q_j \vec{t_i}\vec{t_j}}{\sqrt{\sum_{j=1}^{n} d_j^2 \sum_{j=1}^{n} q_j^2}} \quad . \tag{2.2}$$

The correlation between terms $t_i$ and $t_j$ can be determined in many ways, such as reference to statistical co-occurrence in a large body of text, or correlation in an external semantic resource (Tsatsaronis and Panagiotopoulou, 2009).

### 2.2.2 Probabilistic models

Probabilistic IR models are dominant today due to their strong theoretical foundation for reasoning about uncertainty (Croft et al., 2010). These models summarize uncertainty so that even if they do not comprehensively model the retrieval process, for example by including all possible word dependencies, they can provide good predictions about document relevance (Norvig, 2011).

Probabilistic IR models assume a set of relevant documents, and a corresponding set of non-relevant documents. They are based on the Probability Ranking Principle (PRP) (Robertson, 1977), which holds that an optimal ordering of documents is achieved if documents are ranked according to the probability of relevance for any user with an expressed information need. Let random variables $D$ and $Q$ represent a document and a query respectively, and let the binary random variable $R$ indicate whether $D$ is relevant to $Q$. The relevance of $D$ to $Q$ can take one of two values: $r$ (relevant) and $\bar{r}$ (not relevant). It has been shown that ranking documents according to $P(R = r|Q,D)$, the posterior probability of a document $D$ being in the relevant set, maximizes average retrieval precision (MAP, see Section 2.5) (Croft et al., 2010).

Probabilistic IR models include classic probabilistic models, language models for IR, and the Relevance Model (Lavrenko and Croft, 2001), all of which have variants that incorporate word dependence. Word dependence is also present in several probabilistic models that assert a word independence assumption. Cooper (1991) shows that the mathematical definition of word independence is statistically incompatible with assertions of conditional independence of words given document relevance or its absence. Instead, Cooper argues that models asserting both these claims are founded on an assumption of "linked dependence" in which the degree of statistical dependence between two words in a set of relevant documents is associated in a constant way with their degree of statistical dependence in the corresponding non-relevant set. This

may seem trivial, but the assumption of linked dependence accounts for certain dependencies between words that might otherwise be hard to explain. For example, words that create COMPOUNDS such as '*database*', '*data-base*' and '*data base*' are expected to have a constant degree of association in large collections of documents (Krovetz, 1995), regardless of whether they are relevant to a particular query. (Lavrenko, 2004) asserts that a weaker claim of "proportional interdependence" is sufficient, such that on average, words in a document have as much interdependence in a set of relevant documents as they do in a corresponding non-relevant set. This claim allows some words to be more dependant in a relevant set so long as the dependencies are offset by other word dependencies in the non-relevant set.

The rest of this Section provides further details on classic probabilistic models and language models for IR, two of the most frequently used classes of IR models. The presentation largely follows the one found in Croft et al. (2010).

#### 2.2.2.1 Classic probabilistic models

The first classic probabilistic model is the *binary independence model* (BIM) (Robertson and Spärck Jones, 1988). When word independence is assumed, the BIM is simply a Naive Bayes binary classification that aims to classify documents as relevant or non-relevant. Using Bayes rule, a document is considered relevant if:

$$P(D|r)P(r) > P(D|\bar{r})P(\bar{r}) \ .$$

This is equivalent to saying a document is relevant if:

$$\frac{P(D|r)}{P(D|\bar{r})} > \frac{P(\bar{r})}{P(r)} \ .$$

Document ranking is sufficient for IR, so documents are ordered according to the quantity on the left side of the equation. This is the likelihood ratio, hence probabilistic models are also known as document-likelihood models. In other words:

$$P(r|Q,D) \stackrel{rank}{=} \frac{P(D|Q,r)}{P(D|Q,\bar{r})} \ .$$

BIM assumes word independence, and represents documents as a vector of binary features $D = (w_1, w_2, ..., w_t)$ corresponding to each word in the vocabulary. $w_i = 1$ if the word is in the document, and 0 otherwise. Documents are then ranked according to:

$$P(r|Q,D) \stackrel{rank}{=} \prod_{w \in D} P(w|Q,R) \quad \text{, where}$$

$$P(w|Q,R) = \frac{P(w|Q,r)}{P(w|Q,\bar{r})} \quad .$$

This is equivalent to:

$$P(r|Q,D) \stackrel{rank}{=} \sum_{w:d_w=1} log \frac{p_w(1-s_w)}{s_w(1-p_w)} \quad ,$$

where $p_w$ is the probability that a word $w_i$ occurs (has the value 1) in a document in $r$, $s_w$ is the probability that a word $w_i$ occurs in a document in $\bar{r}$, and $1-p_w$ is the probability that a word $w_i$ does *not* occur (has the value 0) in a document in $r$.

Probabilistic models differ in how they estimate the probability of document relevance $P(r|Q,D)$ and how they approximate the relevant set of documents. The well-established Okapi BM25 model (BM25) (Spärck Jones et al., 2000) improves greatly over BIM with three additional features: *tf.idf* weighting, a variable for document length, and model tuning parameters. These features are carried by additional variables in the discriminant function that have no effect on core estimates of $p_w$ and $s_w$. $P(r|Q,D)$ is approximated heuristically using a simple and effective word frequency formula (Robertson and Walker, 1994).

An alternative to the classic, generative approach estimates probabilities directly using a discriminative (regression) model. This estimate is usually based on a subset of query features also observed in documents (for example, see the Markov random field model in Section 2.2.4). A common weakness for this approach is the lack of guidance on how to select features (Zhai, 2008). Another alternative uses a query to characterize a set of relevant documents during an initial phase of retrieval. The PSEUDO RELEVANT documents identified by this process are used to calculate document rankings. The Relevance Model (Lavrenko and Croft, 2001) is a variant of this approach.

In general, the integration of word dependence in probabilistic models is an established direction of research. An early example is the tree dependence model that improves the accuracy of estimates for $P(w|Q,R)$ by substituting each probability $p_w$ and $s_w$ in the calculation of $P(w|Q,R)$ with a conditional probability (van Rijsbergen, 1977, 1979a). Word dependencies are identified using a maximum spanning tree (MST) over mutual information between words.[2] With the exception of the head node of the tree,

[2] A maximum spanning tree (MST) is a dependency tree represented by a directed, acyclic graph $G$ in which each vertex is associated with a word, and mutual information scores between words weight

this means that every word is characterized as dependent on one other word with which it has the greatest mutual information. The model outperforms a simple independence assumption (Harper and van Rijsbergen, 1978), even though mutual information is not a particularly good measure of word collocation when co-occurrence counts are small (see Section 4.5.7.1).

The tree dependence model conditions each query word on one other word, and thus approximates the true conditional probability that would account for all preceding words. The BLE model (referring to the Bahadur-Lazarsfeld Expansion it applies) (Croft, 1986) incorporates all dependencies between query words.[3] Let the estimate of $P(w|Q,r)$ assume word independence. Dependence is incorporated using the form:

$$P'(w|Q,r) = P(w|Q,r)(1+A) \ ,$$

where $A$ is a measure of the strength of word dependencies in the query, calculated using expected mutual information between words. The method is equally applicable to query dependencies identified using language processing techniques.

For practical purposes, the tree dependence model is similar to a BLE model when correlation parameters for third and higher order dependencies (terms with three or more words) are set to 0, and second order dependencies are subject to the constraints described by a maximum spanning tree. However, higher order correlations are not always negligible (Losee Jr., 1994) and a more effective model might take this into account.

The generalized term dependence model extends the tree dependence model to account for higher order dependencies by decomposing a query tree into subtrees connected by third order dependencies. It performs slightly better than the tree dependence model (Yu et al., 1983), and both models achieve small performance improvements over a standard BM25 model (Croft et al., 2010).

#### 2.2.2.2 Language models

A language model (LM) estimates a multinomial distribution over individual words and word sequences such that each term is associated with a probability of occurrence (Ponte and Croft, 1998; Song and Croft, 1999). The distribution can be applied to gen-

each edge. The score for a specific dependency tree is the sum of its weighted edges. A MST is one of an equivalent set of trees meeting the global constraint that the sum of its edge weights is maximized.

[3]The full Bahadur-Lazarsfeld Expansion specifies dependencies between all subsets of words and can result in an exponential number of expression components.

erate new text and determine the probability of generation. Language models naturally model word dependence, and have been studied extensively.

Word sequences encoded in language models are frequently adjacent terms, such as bigrams and trigrams (Song and Croft, 1999), but sequences can also be identified using proximity measures (Na et al., 2008),[4] syntactic relations, as in the dependence LM (Gao et al., 2004) and dependency structure LM (Lee et al., 2006), semantic relations such as those found in WordNet (Cao et al., 2005), and arbitrary relationships, as with the graph LM (Maisonnasse et al., 2007).

A benefit of language modeling in IR is that the resulting models integrate indexing and retrieval. This means that an improvement in indexing is not required to improve performance of the model (as with the vector space model). Language models also replace heuristic *tf.idf* weights with formal probability estimates. One of three measures is used to rank documents (Croft et al., 2010):

- The probability of generating a query given a document language model. This is known as the *query likelihood* model. It computes $P(Q|D)$, together with the posterior probability of a document $P(D)$. Documents are ranked according to $P(D|Q) \propto P(Q|D)P(D)$.
- The probability of generating a document given a query language model. This approach is similar to the Relevance Model. In the Relevance Model, a distinction is made between relevant and non-relevant documents, and a query is said to share its relevance model with a relevant document.
- The similarity between a query language model and a document language model. A ranking over documents is produced based on the Kullback-Leibler (KL) divergence, or cross entropy, between a query language model and a document language model.

The main challenge with language models is that documents and queries can be very short, resulting in sparse data and poor estimates of term probabilities. Data smoothing, model interpolation and backoff strategies are typically used to improve performance (Ponte and Croft, 1998). Smoothing especially ameliorates the impact of common high frequency words (similar to stopwords) that might otherwise receive

[4]Proximity has been determined by the separation in text of pairwise word combinations (Rasolofo and Savoy, 2003), non-overlapping word sequences of any length (Song et al., 2008a) and statistical measures. See recent surveys by Cummins and O'Riordan (2009) and Tao and Zhai (2007) for more information.

higher probabilities than discriminative words. Longer documents require less smoothing, and dirichlet smoothing is often used because it accounts for document length. Standard unigram query likelihood with dirichlet smoothing performs comparably with the BM25 model, and query likelihood can outperform BM25 when more sophisticated smoothing techniques are applied (Croft et al., 2010).

A second challenge with language models in IR is that the word order used to represent concepts in a query may differ from the word order used to represent the same concepts in documents. Several models address this challenge during calculation of term counts so that alternative word orders are implicitly captured in document LMs. Heuristic adaptations include relaxation of the word order constraint, while continuing to enforce term adjacency (Srikanth and Srihari, 2002), and use of proximity measures to identify dependent terms (Na et al., 2008). Governor-dependent relations and semantic relations that are independent of surface word order can also be used to identify word sequences. These alternatives involve substantial offline data processing, but can translate into slight improvements in IR effectiveness compared to a bigram LM approach (Gao et al., 2004).

Interpolation of alternative LMs can address the limitations imposed by any single word dependence representation. Examples include the multi-dependency LM (Cai et al., 2007d) that uses a hybrid dependency structure composed of syntactic dependency, syntactic proximity dependency, and word co-occurrence, and a hybrid LM that incorporates semantic WordNet relations and word co-occurrence (Cao et al., 2005). Dependence can also be defined between units other than individual words. For example, Srikanth and Srihari (2003b) use a concept language model in which a query is represented as a sequence of concepts (terms) identified by syntactic analysis.

Finally, the Relevance Model (Lavrenko and Croft, 2001) used in Chapter 7 is a prominent approach to language modeling that assumes relevant and non-relevant classes of documents, and constructs a language model that simultaneously represents a query and all documents that are relevant to the query. The general idea is that a query is only a verbal estimate of an underlying information need. Relevance is viewed as a generative process, and queries and relevant documents are random observations from that process. The retrieval procedure involves two passes. First, initial retrieval with a query identifies a pseudo relevant document set. Second, a number of top retrieved documents are used to estimate term probabilities in the relevant set. The original term probabilities in the query model are interpolated with term probabilities in the model built from the relevant documents. This effectively obtains a smoothed, expanded

model of the query that represents the latent query topic. Documents are ranked by computing the KL-divergence between the relevance model and each document model.

Notice that in a language modeling approach to IR, it is inappropriate to simply add selected terms to a query, or adjust query term weights. This is because the new query would conceptually no longer be a sample from the language model (Zhai, 2008).[5] Instead, the Relevance Model interpolates term probabilities in an original query model with term probabilities in a model built from the relevant document set, effectively obtaining a query expanded with novel words that were not in the original query. Latent concept expansion (LCE) is a generalization of Relevance Model expansion that can generate multi-word terms as well as individual words (Metzler and Croft, 2007b).

### 2.2.3 Inference networks

An inference network for IR models retrieval as an evidential reasoning process in which the goal is to estimate $P(I|D)$, the probability that a user's information need is met by a document (Turtle and Croft, 1991). Evidence for this proposition is an arbitrarily complex function represented in a Bayesian inference network. The inference network is a directed acyclic graph (DAG) in which nodes are events that are classified into three types: observation of a document (the document node); evidence that an information need has been met (representation nodes); and combinations of evidence (query nodes, and the node representing the information need). The inference network is rooted at the document node, and belief is propagated through the network in order to determine the probability that the information need is met.

Inference networks are highly flexible because they can incorporate arbitrary types of textual and non-textual evidence (Croft et al., 2010; Turtle, 1995). Examples of evidence include word occurrence, document type, citations, and location-specific word occurrence and co-occurrence. The latter are defined by proximity operators and appearance in document sections such as title, heading etc. (Croft et al., 2010). Arcs in the graph represent probabilistic dependence between events, and the binary value of a node is a function of belief in the parent nodes or the prior probability of observation. In the case of the root document node, this probability is usually set at $\frac{1}{collection\ size}$. The probability of an information need being met is assessed one representation node at a time by propagating belief through the network. The score at the node for the infor-

[5] The query is a sample drawn from a distribution, and adding words to a query alters the observations in the sample. Rather than alter the observations, it is preferable to alter the model that captures belief about how the observations are generated.

mation need combines evidence from the query nodes, and is used to rank documents.

The highly effective IR system used in this dissertation (Indri version 4.12)[6] combines an inference network with language modeling by using language models to estimate arc probabilities in the network (Metzler and Croft, 2004). These probabilities represent whether a node feature such as 'word $x$ adjacent to word $y$' would be generated by (is TRUE for) a model of a document (Croft et al., 2010). Where a feature does not appear in a document, it is assigned a default probability. Figure 2.1 depicts an example of a combined inference network and language model for retrieval (Croft et al., 2010). In the Figure, $D$ is the document node, the parameters used to estimate language models are represented by $\mu$, and the language models are represented by $\theta$. Feature nodes $f$ are document features, such as a query term whose words are adjacent in document text, and belief nodes $b$ are used to combine beliefs (probabilities) within the network. Finally, the information need node $I$ is a belief node that combines all evidence in the network. The value at $I$ is the basis for document ranking.[7]

### 2.2.4 Linear feature-based models

A linear feature-based retrieval model is a discriminative probabilistic model that ranks documents according to $P(r|Q,D)$, the probability of relevance given a query and a document. This is achieved with a weighted linear combination of features. Features are typically document scores calculated with respect to a query $Q$ using one or more existing retrieval functions (Gao et al., 2005; Metzler and Croft, 2005). For example, taking a language modeling approach, a feature might be the score for a document using a unigram query likelihood model. Other features might be document scores computed for the same query representation using document language models that capture various forms of dependency. The query might also have several representations, each of which contributes its own document score.

The motivating intuition behind linear feature-based retrieval is that a combination

[6] `http://www.lemurproject.org/`

[7] An inference network defines a joint probability distribution over a collection of random variables. To estimate $P(I|D)$, term representation beliefs first must be computed for each node $f_i$ using the language model for document $D$. The probability of term relevance given a document, $P(r|D)$, is estimated by marginalizing out the node for the language model using the expectation over the posterior $P(\theta|D)$. This assumes that the expected value of the model prior, $P(\theta)$, is equal to $P(r|C)$, the probability of relevance given the collection model. Various belief operators are then applied at each belief node $b_i$, corresponding to query operators specified in the query. In Indri query language, operators other than *#not* assume $n$ parent nodes, each with belief $b_i$ and, for weighted operators, weight $w_i$. The formulas for belief operators are combined with term representation beliefs to determine the exact ranking function used for any structured query. Belief flows through the network to determine $P(I|D)$.

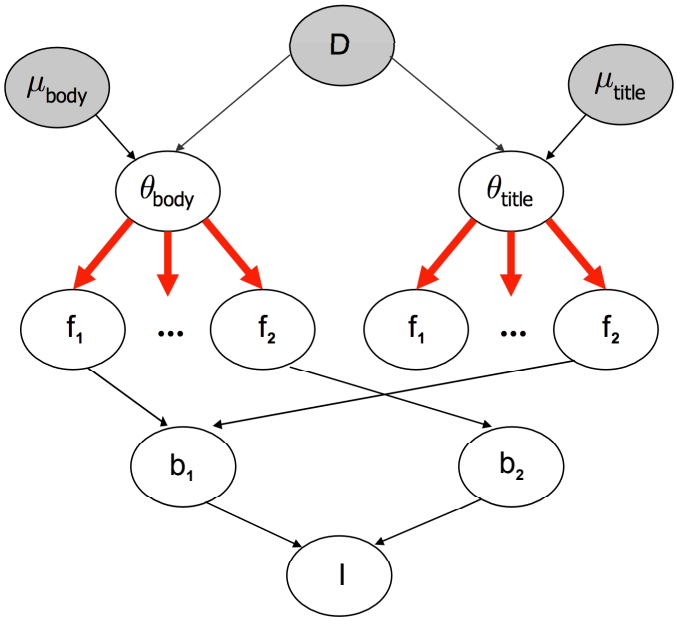



Figure 2.1: A graphical model depiction of a typical inference network for IR.

of many sources of document evidence improves retrieval performance compared to a model that uses only one source of evidence. Multiple sources of evidence help to identify an optimal document ranking because an optimal ranking is more stable than errors made by any particular retrieval strategy. A similar intuition underlies inference networks and discriminative learning to rank techniques (Liu, 2009). Essentially, these are meta-retrieval models that combine different document ranking strategies, and rank documents using multiple query representations (Zhai, 2008).

A combination of different IR strategies makes feature-based models an excellent framework for the exploration of synergistic effects between established and novel retrieval strategies. The IR models applied in this dissertation are linear feature-based models with text features interpreted as evidence nodes by the Indri retrieval engine (see previous Section). The resulting queries can be highly effective. A model that considers a unigram query representation, together with a bigram query representation, is more effective than a unigram language model alone (Song and Croft, 1999). Moreover, recent work using linear feature-based models demonstrates significant improvement over competitive baselines (Balasubramanian et al., 2010; Bendersky and Croft, 2008; Metzler and Croft, 2007a; Park et al., 2011; Xue et al., 2010).

The Markov random field (MRF) model for IR (Metzler and Croft, 2005, 2007b) is a straightforward, robust linear feature-based model that incorporates word dependence. Formally, it is constructed from an undirected graph $G$, in which vertices are random variables and edges represent dependence between variables. A random vari-

able in the graph is independent of all nodes with which it does not share an edge, and dependent on variables with which it does share an edge (the Markov assumption). Random variables in $G$ represent query words and a document under consideration. A clique $c$ in $G$ is a subset of dependent words, or terms, as defined on $G$. Let $C(G)$ be the set of all cliques in $G$. For the purposes of ranking, the posterior probability of a document $D$ being relevant to query $Q$ is derived as (Metzler and Croft, 2005):

$$\begin{aligned} P_\Lambda(D|Q) &= \frac{P_\Lambda(Q,D)}{P_\Lambda(Q)} \\ &\overset{rank}{=} logP_\Lambda(Q,D) - logP_\Lambda(Q) \\ &\overset{rank}{=} \sum_{c \in C(G)} log\psi(c;\Lambda) \ , \end{aligned}$$

where $\Lambda$ represents uncertainty that a document is relevant to a query, and $\psi$ is a positive, real-valued potential function that represents the contribution of clique evidence to document relevance. $P_\Lambda(Q)$ is the same for every document, so the model factorizes $P(D|Q)$ into a sum of factors $\psi(c;\Lambda)$ usually parameterized as:

$$\psi(c;\Lambda) = exp[\lambda_c f(c)] \ .$$

Here, $f(c)$ is a feature function, or feature, and $\lambda_c$ is a feature weight. As with other linear feature-based IR models, features are typically document scores calculated with respect to a query. In practice, cliques often behave like clique sets: groups of cliques associated with the same potential function. In this case, the clique feature value is the sum of the feature values for each element in the set.

The primary baseline model in this dissertation is the sequential dependence (SD) variant of the MRF model (Metzler and Croft, 2005). SD is one of the most successful IR models, and is widely used as a competitive benchmark for retrieval performance. It assumes dependence between adjacent words in surface query text, and is known to perform as well as, or slightly better than, a bigram language model for IR. For most collections, a weighted variant of SD outperforms all other major IR models (Huston, 2013). The SD model uses a linear combination of three cliques, where each clique is prioritized by a weight $\lambda_c$. The first clique contains individual words (query likelihood *QL*) with default $\lambda_1 = 0.85$. The second clique contains query terms that are evaluated by the sequential and ordered appearance of component words in documents (*ordered* window *#1*) with default $\lambda_2 = 0.1$. The third clique uses the same terms as

clique 2 but searches for them in an *unordered* window size of 4 multiplied by the number of words in a term (*#uw8*) with default $\lambda_2 = 0.05$. The difference between ordered and unordered windows is shown below for the term '*united states*'. A *#1* ordered window would only match example 2.5, while an unordered window *#uw4* would match examples 2.3 - 2.5.

(2.3) '*the states united against the federal initiative...*'

(2.4) '*a spokesperson for the united presbyterian church states...*'

(2.5) '*the united states government signed...*'

Note that for a term with only one word *w*, the operators *#1(w)* and *#uw8(w)* equate to a search for the word *w* in a document.

There are two basic shortcomings of the SD model. First, it only assumes first-order word dependence. The full dependence variant of the MRF model for retrieval assumes that all query terms are dependent on each other in some way. However, many of the assumed dependencies do not contribute positively to IR effectiveness, and the SD variant closely approximates performance of the full dependence variant (Metzler and Croft, 2005). Second, each element in a clique that behaves as a clique set contributes equally to the final document score. This is clearly inappropriate since some elements in the clique set (e.g. query terms) are likely to be more discriminative than others. A straightforward improvement applies non-uniform weighting to reflect the contribution by each element. Optimization of weights via a supervised learning approach significantly improves model performance (Lease, 2009).

Combined query representations in linear feature-based models can be particularly effective when weights and features are optimized automatically (Lease, 2009; Metzler and Croft, 2005, 2007a). However, the hazard with these models is that queries can quickly become lengthy and complex with many different forms of evidence. The challenge is to improve on their results using simpler, and more efficient, query reformulations. This is the challenge addressed by term selection.

## 2.3 Techniques for query reformulation

A variety of query-based techniques can be applied to improve document rankings. These QUERY REFORMULATION techniques start with a query $q$, and typically use knowledge of the query, the document collection, and external resources to develop a query

$q'$ that improves the representation of the information need. Techniques start with simple transformations such as tokenization, stopping (removing words on a stoplist) and stemming of both queries and documents to ensure that query and index terms match. They also include more sophisticated term selection, query expansion, query reduction and term weighting methods.

Term selection is the focus of this dissertation and can be applied to both title queries and description topics. However, term selection is not the most appropriate reformulation technique for title queries since they often respond better to smoothing and query expansion (Huang et al., 2010; Mei et al., 2008; Zhai and Lafferty, 2001). Conversely, verbose queries are less focused and prone to query drift when using query expansion. By consequence, their reformulation focuses on term weighting (Bendersky et al., 2010; Lease et al., 2009), term selection (Balasubramanian et al., 2010; Bendersky and Croft, 2008) and query reduction (Huston and Croft, 2010). These techniques become critical as the number of potentially noisy query terms increases. Other query reformulations entail specialized IR systems, such as query augmentation with metadata to specify query semantics (Chu-Carroll et al., 2006), and are not the focus of this work. The rest of this Section considers several query reformulation techniques that are relevant to IR models discussed in later Chapters.

### 2.3.1 Term weighting

In this dissertation, I compare IR models that use novel term selection methods with previously reported models that use term weighting. Term weighting aims to assign higher numeric values to terms that are good discriminators of relevant documents. This quantifies term importance and improves document ranking. Weights are usually based on term frequencies in documents, document collections, query logs, and other textual resources, and may be determined heuristically or by machine learning.

Term weighting is similar to term selection because it can eliminate terms with poor discriminative ability by reducing their weights to zero. It is also complimentary to term selection because there is a trade off between the accuracy of term selection and the need for term weighting. If term selection is poor, and multiple terms are selected, then terms that are not discriminative can be down-weighted to improve results. Conversely, term weighting may not affect queries with only a few well-selected terms.

In addition, term weighting is closely associated with IR modeling. Older IR models incorporate term weights directly, and language modeling for IR replaces explicit

term weights with term probabilities (see Section 2.2.2.2). Some IR models are applied to classes of terms, such as bigrams, and perform term selection by default.

The most widely-used and popular weighting schemes are *tf.idf* formulas (Zhai, 2008; Robertson and Spärck Jones, 1988) that multiply term frequency (*tf*) and inverse document frequency (*idf*). Term frequency is a measure of salience: the more frequently a term appears in a document, the more likely it signifies information about that document. Very long documents are more likely to contain higher counts of any term, so frequency scores are normalized for document length. Inverse document frequency measures discriminatory ability: a term appearing in very few documents is better able to discriminate between relevant and non-relevant documents than a term appearing in a large number documents. A standard formula for *idf* is $log\frac{N}{n}$, where $N$ is the number of documents in a collection, and $n$ is the number of those documents that contain a specific term. The two most well-known *tf.idf* weighting schemes are the one used in the BM25 retrieval model (Robertson and Walker, 1994) (see Section 2.2.2.1) and an improved version of *tf.idf* that uses pivoted document length normalization (Fang et al., 2004; Singhal, 2001). The latter varies normalization to correct for error between the estimated and actual document relevance scores on training data. There are many more variants of *tf.idf* weighting. At a basic level, they all combine some measure of term occurrence in a specific document, and occurrence in a collection of documents.

A direction for recent research is predictive term weighting based on collection or external resource statistics, rather than RELEVANCE FEEDBACK or pseudo relevance feedback (Robertson and Spärck Jones, 1988). Predictive weighting optimizes term weights using supervised learning techniques typically trained on metrics of IR effectiveness (see Section 2.5). This approach is known to deliver performance gains with verbose queries for many IR models (Bendersky and Croft, 2008; Bendersky et al., 2011; Balasubramanian and Allan, 2009; Balasubramanian et al., 2010; Huston and Croft, 2010; Kumaran and Allan, 2007, 2008; Kumaran and Carvalho, 2009; Lease et al., 2009; Lioma and Ounis, 2008; Shi and Nie, 2010; Xue et al., 2010). Some of the comparison models in this dissertation use predictive term weighting.

Term weighting can also incorporate word dependence information. Whereas standard *tf.idf* weights use word frequencies, context-dependent weighting considers word associations in a window of text surrounding a query word. The procedure is intended to mimic a user reading text in the vicinity of a query term to decide whether the document portion is relevant to the query (Dang et al., 2010; Wu et al., 2008).

### 2.3.2 Query expansion

Two query reformulation models presented in Chapter 7 use pseudo relevance feedback for query expansion. Query expansion suggests, or adds, words (or rarely, terms) to a query in order to better describe an information need and overcome vocabulary mismatch. The basic approach selects expansion words that are closely associated with a query word $q_i$ in documents relevant to query $q$. Alternatively, expansion words may be associated with $q_i$ in an external resource such as an ontology or thesaurus. Typically, expansion words do *not* appear in the original query.

Approaches to query expansion may be syntactic, semantic or based on word co-occurrence. In the 1960s, many experiments used a manually, or automatically, constructed synonym dictionary or thesaurus to choose expansion terms, where the thesaurus contained terms from the same controlled vocabulary used to index documents (Salton, 1963). Expansion terms were associated with original query terms in the thesaurus. Syntactic relations were also a basis for identification of closely associated words in an external knowledge base or plain text. For example, Grefenstette (1992) identified candidate expansion words as adjectives and nouns modifying occurrences of query terms in documents, either syntactically or through VALENCE relations. More recently, associated terms have been identified using thesauri, ontologies and other semantic resources, including formal description logics and Semantic Web resources (Meij et al., 2009). Unfortunately, it is difficult to construct a semantic resource that is sufficiently complete for IR. Outside of narrow domains, individual words can be highly ambiguous, and techniques that use general resources, such as WordNet, have not been shown to be effective (Croft et al., 2010).

Query expansion based on statistical word co-occurrence uses word sequences in a query log, a document collection, or a subset of documents. A straightforward example of co-occurrence-based query expansion derives from the Hyperspace Analogue to Language (HAL) (Lund and Burgess, 1996) IR model. Candidate expansion words in a pseudo relevant text are ranked by combination and comparison of word co-occurrence vectors (Bruza and Song, 2002). Expansion words can also be identified by statistical word association measures, such as the DICE coefficient, mutual information, expected mutual information, t-score, $\chi^2$ coefficient, and log likelihood ratio (see Section 4.4). These measures can be used independently or in combination with syntactic methods. For example, Vechtomova and Karamuftuoglu (2004) explored noun phrases filtered

by statistical collocability (C-value[8], mutual information or average *tf.idf* score). The measures can also define co-occurrence syntactically and semantically.

Statistical approaches to query expansion are robust but can impose high computational overhead when applied to very large text collections of variable quality (Croft et al., 2010). In addition, for global co-occurrence data, they do not distinguish between different semantic senses of words. This can bias expansion towards words that appear frequently in a collection because they have many senses, but are poor discriminators between relevant and non-relevant documents (Peat and Willett, 1991). When this occurs, expansion can result in queries that perform no better, or even worse, than unexpanded queries (Peat and Willett, 1991). The consistent selection of expansion words that improve IR effectiveness without undue computational burden is difficult.

To help ensure that query expansions reflect an information need, word association techniques can be applied to a subset of relevant documents. This subset is identified explicitly with relevance feedback or implicitly with pseudo relevance feedback. Expansion using pseudo relevance feedback is practical but can be unpredictable due to the variable quality of top-ranked documents. For this reason it is more reliable to suggest alternative queries when an initial query does not perform well, some of which may be expansions.

The best approach to practical query expansion today is probably query suggestion by lookup in a large query log (Croft et al., 2010). This approach is applied in modern commercial search engines and leverages intuitions about query patterns. For example, it may be assumed that queries from the same session are related, that users click through to the same documents from related queries, and that for a query of size $n$, related queries are the most probable or 'trending' (rapidly increasing in popularity) query ngrams of size $n+1$. Unlike most expansion strategies, a multi-word term (the query) is assumed to be the important unit of meaning, not independent words.

Query-to-question generation is also related to query expansion, but organizes the informational content of a query syntactically at the same time that the content is augmented with new terms (Zhao et al., 2011). A potential benefit of query-to-question generation is a better understanding of the inverse process of query reformulation and reduction, including the identification of important word dependencies. However, this opportunity has yet to be explored empirically.

[8]C-value is a measure of the stability of an ngram in a corpus.

### 2.3.3 Term selection

This dissertation focuses on query biased term selection (henceforth, 'term selection'), the identification of text units in queries that best represent an information need and help to improve retrieval effectiveness. For example, given the query, '*What are the pros and cons of term limits?*' (Robust04 #699), search performance can be improved by a factor of almost 20 by recognising '*term*' and '*limits*' as the phrase '*term limits*'.[9]

In term selection, some words in an original query may be omitted. Query segmentation is a form of term selection in which every query word is assigned to a term and used in query reformulation. It is ideally suited to keyword queries and is well-studied. Recent segmentation techniques include a method for learning noun phrases by making a binary decision at each possible segmentation point (Bergsma and Wang, 2007); ranking all possible query segmentations by their probability according to a concept language model constructed from English Wikipedia (Tan and Peng, 2008); segmentation using mutual information scores of word combinations in query logs (Risvik et al., 2003); and a simple and effective segmentation method based on corpus frequencies (Hagen et al., 2011). This last method inspires term features applied in Chapter 6.

Term selection has a long history in IR research, in which terms are variously described as query phrases (Fagan, 1987), dependent terms (Park et al., 2011), key concepts (Bendersky and Croft, 2008), and sub-queries (Kumaran and Carvalho, 2009; Xue et al., 2010). In the 1980s, the dominant approach to term selection identified syntactic phrases in queries that matched syntactic phrases used for document indexing. For example, the FASIT system (Dillon and Gray, 1983) matched combinations of grammatical categories, such as adjectives and nouns, against pre-defined category patterns. Later, Fagan (1987) took a more comprehensive approach with terms identified by both statistical collocation and syntactic rules. He determined that syntactic terms are comparable to less complex, and therefore preferable, statistical terms (Salton, 1989).

Following these linguistically inspired methods, other techniques combined constituents with governor-dependent relations, statistical collocations or term frequency data (see Chapter 3 for a description of linguistic structures). Examples of combinatory approaches include selection of terms composed of the heads of phrase structure constituents connected by grammatical relations or certain neighbouring relations (Lewis

[9] In Indri query language, the query '*#combine(pros cons term limits)*' has a mean average precision (MAP, see Section 2.5) score of 0.0100, whereas the query '*#combine(pros cons #1(term limits))*' has a MAP score of 0.2080.

and Croft, 1990); selection of terms identified by predicate-argument structures involving nouns (Strzalkowski and Carballo, 1993); and selection of word sequences in particular part of speech blocks on the basis of collection frequency (Lioma and Ounis, 2008). More recently, noun phrases have been combined with dependency relations in a weighted model for sentence retrieval (Balasubramanian and Allan, 2009), and used with a graph-theoretic representation of mutual information between query terms (Kumaran and Allan, 2007).

The structures from which constituents and governor-dependent pairs are identified have also been used for term selection, rather than the word combinations themselves. For example, in the late 1980s and early 1990s, a Constituent Object Parser (COP) was used to match headed phrase structure parses for queries and documents (Metzler and Haas, 1989). Likewise, Tree Structured Analytics (TSAs) were used to match binary trees constructed from governor-dependent relations and base grammatical categories (Sheridan and Smeaton, 1992; Smeaton et al., 1995; Smeaton, 1999).

In general, research in term selection has two major trends. First, selection has moved progressively away from simple, constituent-based approaches while syntactic matching using governor-dependent relations continues to evolve, especially in question-answering (QA). Recently, dependency path-based techniques have been inspired by work in machine translation that matches many syntactic paths simultaneously (Wang et al., 2007). For example, in ad hoc IR, Park et al. (2011) model document relevance as a translation problem between the dependency parse structures of queries and documents. A similar approach is taken by Zhou et al. (2011) in matching user questions to pre-existing questions (and consequently, answers) in a Community Question Answering (CQA) database.

The second trend is toward methods that use features of constituency, headedness and term frequency, rather than explicit linguistic elements or the structures from which these elements are derived. These methods leverage machine learning and mathematical models that can readily incorporate syntactic and non-syntactic features. For example, the key concept (KC) model (Bendersky and Croft, 2008) classifies noun phrases using a decision tree with frequency-based features. Xue et al. (2010) explore a conditional random field model with syntactic and non-syntactic features. Other approaches rank query terms and subqueries using dependency tree relations (Park and Croft, 2010) and other syntactic features (Kumaran and Carvalho, 2009; Park et al., 2011; Xue et al., 2010). Frequency-based features are prominent in many approaches, and are sometimes used exclusively. For example, Balasubramanian et al. (2010) use

frequency-based features to reduce queries to a single term, and Huston and Croft (2010) take a similar approach to remove 'stop structure' for search in a standard web interface.

Finally, work in entity identification for queries is related to term selection because entity names are often informative terms. Luo et al. (2013) use syntactic and semantic features, including entity types, to identify Mandatory Matching Phrases in question answering, and Guo et al. (2009) use Weakly Supervised Latent Dirichlet Allocation (WS-LDA) to identify named entities in a commercial query log. However, entity identification is distinct from term selection. Entity names are restricted to single entities with greedy scope, and are properly resolved with respect to external resources. In contrast, informative terms may contain concepts with several entities, can be identified without reference to external resources, and are often more effective if reduced to essential elements e.g. '*thatcher*' rather than '*Prime Minister Margaret Thatcher*'.

## 2.4 Retrieval collections

Text REtrieval Conference (TREC) corpora are standard test-beds for IR evaluation and are used in this dissertation. TREC is an on-going forum organised since 1992 by the National Institute of Standards and Technology (NIST) and the US Department of Defence to facilitate statistical evaluations of retrieval systems (Voorhees, 2005). TREC corpora consist of a collection of documents, a set of query topics, and relevance judgments with respect to those topics for documents that were among the top $k$ returned by IR systems when the test-bed was developed. Although these judgments are not exhaustive, it can be assumed that most of the relevant documents are known.

Three TREC collections are used in this dissertation to provide queries that vary substantially in length and known difficulty. Together, they provide a diverse platform for experiments (see Table 2.1). Robust 2004 consists of 250 queries focused on poorly performing, or hard, topics. This means the topics are known to be difficult for IR systems, often due to the way the query is expressed or because the number of relevant documents is very low (like searching for a needle in a haystack). The document collection is relatively small with approximately 528,000 documents. Topic 672 is excluded from Robust04 evaluations in this dissertation as the collection contains no relevant documents for this topic. WT10G uses a larger collection of over 1.5 million web pages and 100 query topics. The much larger GOV2 has 150 queries and 25 million web pages crawled from websites in the `.gov` domain.

| Name | # Docs | Topic Numbers |
|---|---|---|
| ROBUST04 | 528,155 | 301-450, 601-700 (-672) |
| WT10G | 1,692,096 | 451-550 |
| GOV2 | 25,205,179 | 701-850 |

Table 2.1: TREC collections and topics

## 2.5 Evaluation metrics

Many evaluation measures have been proposed and are currently employed in IR (Manning et al., 2008). At the most basic level, the effectiveness of an IR system, or model, is measured in terms of *precision* and *recall* (van Rijsbergen, 1979a). Precision measures the ability of a system to retrieve only relevant items from a collection, and recall measures the ability of a system to retrieve all relevant items in a collection:

$$Precision = \frac{number\ of\ relevant\ items\ retrieved}{total\ number\ of\ items\ retrieved}$$

$$Recall = \frac{number\ of\ relevant\ items\ retrieved}{number\ of\ relevant\ items\ in\ collection}$$

Both precision and recall are defined on unordered sets of retrieved documents, and interpreted at different cut-off and balance points using interpolation, or via graphical comparisons and definitions. The following metrics are reported in this dissertation and computed using TrecEval, the evaluation utility provided by TREC. Descriptions below are based on NIST standards (Manning et al., 2008; Voorhees, 2001).

- **Average precision:** This metric is calculated by averaging the precision after each relevant document is retrieved. It summarizes performance over all documents in a collection, and rewards systems that rank relevant documents earlier (high) in the retrieved set. It is thus well suited to the evaluation of open domain search, where users care almost exclusively about the top ranked results. However, for queries with many relevant documents, a long tail of lower ranked documents can have a substantial impact. The Mean Average Precision (MAP) is calculated over all queries in a set, and demonstrates exceptionally good stability and discrimination between systems (Manning et al., 2008). However, it weights queries equally, so it does not reveal variation in performance for queries that have a great many, or only a few, relevant documents. MAP can vary widely

across queries, and must be calculated on a fairly large and diverse test set in order to be meaningful.

- **Precision at k:** This metric is suitable for evaluation of IR tasks where only a limited number of results are pertinent to a user. For example, in open domain search, ten results are presented per web page, so a user might be interested in precision at $k = 10$. Precision at the threshold of $k$ documents is summed over all queries in a set, and divided by the number of queries. Precision at $k$ is not a good measure for performance across a set of queries because queries with more relevant documents tend to have higher precision at $k$.

- **R-Precision:** A summary statistic that represents one point on the precision-recall graph. This is the precision after $R$ documents have been retrieved, where $R$ is the number of documents relevant to a query. R-Precision de-emphasizes rank order and adjusts for the size of a set of relevant documents. This makes it particularly useful when very large, or small, numbers of documents are relevant. A drawback is that it requires all relevant documents to be known. The average R-Precision over many queries is calculated as the mean of the R-Precisions for each query in a set. R-Precision is highly correlated with MAP.

- **Normalized discounted cumulative gain (NDCG):** This metric scores a ranked list of documents compared to some optimal ordering, where the notion of relevance is continuous rather than binary. NDCG is *cumulative* because it is averaged over the top $k$ documents. For example, NDCG for the top 15 documents ($k = 15$) is referred to as NDCG15. It is *discounted* so that higher ranked documents are weighted to have more effect on the resulting score. This assumes that for a given ranked document, a user cares more about its relevance than any document with a lower ranking. Finally, NDCG is normalized across queries so it reflects overall performance.

## 2.6 Conclusion

This Chapter reviewed concepts, techniques and IR models related to word dependence and term selection for IR. An important practical issue with word dependence in IR is the potential for a dependence model to generate a very large number of search terms. This can make dependence models cumbersome. Moreover, a dependence assumption can lead to problems with data sparsity and increased estimation errors in statistical

models. As a result, gains in IR effectiveness achieved with the inclusion of word dependencies may not outweigh losses. In fact, most dependence models for IR have not brought significant improvements in retrieval effectiveness (Gao et al., 2004).

Nevertheless, several highly successful IR techniques select informative terms using a variety of linguistic and non-linguistic features. These techniques include query segmentation, reduction, expansion and term weighting in addition to explicit term selection. A key question is how the effectiveness of term selection can be further improved.

The next Chapter considers the first of three factors that can affect whether a term is informative for IR, namely the linguistic principles that underlie word association. The focus is on three major theories of language previously applied in IR: phrase structure theory, dependency theory and lexicalism. The presentation of linguistic theory also informs later discussion of novel techniques for term selection.

# 3

# Linguistic Principles for Term Selection

In the terminology of IR, word dependence is a statistical relationship that is linguistically unspecified.[1] Words may be treated as dependent if they are adjacent in surface text or have a statistical association (see Section 4.4). This Chapter presents an alternative view in which word dependence as defined in IR reflects word associations identified by linguistic theory. A WORD ASSOCIATION is a relation between two or more words, e.g. $x$ and $y$, such that the presence of $x$ provides information about the semantic meaning of $y$, the grouping of $x$ and $y$ to form the semantic meaning of some larger unit, or the possibility that $y$ will be observed.[2]

This view removes an unnecessary dichotomy that is sometimes set up between linguistic and statistical approaches for IR, nurtured by the notion of a contrast between 'linguistic processing' and any approach that does not use syntactic annotations (specifically, approaches that rely on word co-occurrence). The dichotomy is misleading. Word distributions form the basis of statistical approaches to language processing and are also integral to some forms of linguistic interpretation. Grammatical categories, frequently used as syntactic annotations, are based on distributional similarities in text. In addition, many statistical approaches to indexing, query representation, and query expansion are applications of a lexicalist theory of language (see Section 3.3) stripped of theoretical linguistic implications. Put simply, neat separation of statistical and linguistic approaches to language neglects the history of linguistic theory.

[1] *Word dependence* carries a specific syntactic-semantic interpretation in linguistics (governor-dependent relations - see Section 3.2).

[2] Semantics is properly a branch of linguistics, but this is sometimes overlooked in IR, e.g. "linguistics and semantics" Bilotti et al. (2010)

This Chapter explores the possibility that word dependence reflects linguistic principles, without prejudice to whether the word associations described are predominantly syntactic, statistical or otherwise. By consequence, the aspects of language accentuated by linguistic theory influence the utility of word associations for IR. Consideration of the linguistic principles that ground word association generates insights upon which to build and adjust term selection strategies. It also facilitates understanding of how users produce queries from an information need. A purely statistical or mathematical interpretation of dependency does not provide such comprehensive guidance.

This Chapter provides information on three linguistic theories that allow different representations of language. This information is necessary to explain relationships between the theories, and where appropriate their relationships with statistical text processing. The theories represent distinct schools of linguistic thought and describe major frameworks for identification of word associations in IR. Alternatives and variant theories are mentioned, but are not in focus because they have not been applied in IR. The theories considered are:

- **Phrase structure theory:** a Chomskyan system of recursive, deterministic rules for combining a fixed, universal set of grammatical categories in a logical way. Grammatical categories are based on the distribution of abstract word categories, and are classified according to the role they play in a sentence. For example, a thing might be an actor, acted upon or used in an action. Phrase-structure grammars are simple, elegant models of the language people *should* speak, rather than the language they *do* speak (Norvig, 2011). In this way they are PRESCRIPTIVE rather than DESCRIPTIVE.[3]

- **Dependency theory:** a direct model of binary asymmetrical relations between words called *dependencies*. The manner in which these dependencies are identified varies, but they share a core notion of valency. Valency refers to requirements and restrictions placed on the type and number of word arguments and COMPLEMENTS. There is no difference between phrase structure grammars and dependency grammars with respect to their ability to distinguish and describe GRAMMATICAL sentences. However, there is no evidence that IR requires judgments of grammaticality. Dependency grammars are simple and compatible with

[3]Phrase structure theory is prescriptive, even though it is based on Structuralism, which is often called 'descriptive linguistics'. Structuralists identify grammatical categories by the ways in which they differ from other categories. In doing so, they *describe* the categories present in language. However, the use of grammatical categories in fixed phrase structure rules is prescriptive, not descriptive.

word distribution analyses. They also capture semantic information and are descriptive of language, not prescriptive. For these reasons they may be preferable to phrase structure grammars for IR.

- **Lexicalism:** a theory holding that language cannot be understood in an abstract, idealized form. Lexicalism analyzes words and phrases as they are used in the real world, taking distributions of words and phrases with respect to other words and phrases, rather than distributions of grammatical categories with respect to other grammatical categories (Cowie, 2005; Hunston and Francis, 2000). Lexicalism is strongly descriptive and maintains that meaning is primarily a property of phrases, rather than individual words. By consequence, meaning is partly determined by lexical context. Moreover, the probability of meaning can be determined by analysis of context in training data. As discussed in Section 3.3, statistical approaches for IR are most closely related to lexicalism, although the heritage is not widely acknowledged.

This Chapter contains one Section for each of the three linguistic theories. Each Section begins with a general definition of the approach, major variants of the theory, and its structures for language representation. This description helps to characterize terms compared in this dissertation, but might be skipped by a reader familiar with linguistics. The ensuing subsections place linguistic theory firmly in the context of IR. They identify theoretical issues and viewpoints that bear on the suitability of each theory for term selection in IR. They also present salient research trends and the context in which they evolved. These subsections should provide sufficient contextualized review for the reader familiar with linguistics who is interested in the application of language processing to IR. Finally, the theory Sections are followed by broad remarks about the expected efficacy of linguistic theories for IR.

## 3.1 Phrase structure theory

### 3.1.1 First principles

Phrase structure theory emphasizes, and is designed for, those aspects of language that adhere to a principle of compositionality: the meaning of a phrase is a function of the meaning of its parts and the way they are put together syntactically. These parts, called constituents, are units of text that are processed together, and composed into larger

constituents to form a hierarchical structure. They are classified into grammatical categories based upon the complementarity and predictability of their distributions with respect to other constituents. These categories define the roles that constituents play in a sentence (Bloomfield, 1933; Tsarfaty, 2010).

For example, we might observe that a very large set of word combinations can appear after the verb '*used*', such as '*used the technique*' and '*used the method in the book*'. The same set of word combinations might also be observed before the verb '*used*', as in '*the technique used*', '*the method in the book used*'. Based on these observations, we could hypothesize the existence of a grammatical category that defines all the word combinations that can appear before or after a verb like '*used*'. This category identifies the role that these words, or groups of words, fill in a sentence or construction. In this example, the grammatical category is a noun phrase. Other constituents, as well as combinations of constituents, can be discovered in a similar way.

Grammatical categories determine roles in a sentence, so any constituent of a given type can be substituted for any other constituent of the same type, and the resulting sentence will be grammatically correct regardless of whether it makes sense semantically. This is a defining feature of phrase structure, and gives rise to Noam Chomsky's famous example of the grammatical, but non-sensical sentence, "Colorless green ideas sleep furiously" (Chomsky, 1957).

Phrase structure is derived from the STRUCTURALIST idea that every element of language can be understood through its relation to other elements. Structuralism was the dominant school of linguistics before Chomsky, and broadly focused on the taxonomic classification of language phenomena such as terms, phonemes and morphemes, much as biology once focused on the taxonomic classification of animals (Graffi, 2005).

Phrase structure was accidental in some Structuralist grammars (Searle, 1972), but Chomsky made the derivations of sentences using phrase structures explicit in his ground-breaking publications on syntax in 1957 and 1965 (Chomsky, 1957, 1965). The number of sentences that can be uttered is infinite, rendering Structuralism inadequate to classify clauses and sentences, and thus explain language data. To resolve the issue, Chomsky developed generative phrase structure rules that take a canonical form, with a constituent on the left, and an acceptable, ordered decomposition of smaller constituents on the right, such as:

$$VP \rightarrow V \;\; NP$$

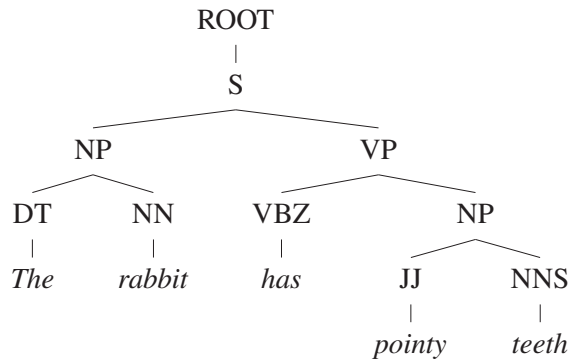



Figure 3.1: A phrase structure tree.

These rules recursively represent language, and use combinations of lower-level constituents to interpret higher-level constituents. This results in a highly economical rule set, or phrase structure grammar, that can be used to generate grammatical sentences.

Like the rules in many linguistic theories, phrase structure rules completely determine the sentences that are, and are not, acceptable in a language. As a set, they describe a phrase structure grammar that can be used to analyze a sentence.

A sentence analyzed with phrase structure rules produces a parse taking the form of a tree with one root node. A phrase structure tree is a labeled, ordered, directed, connected graph in which terminal nodes (nodes with no children) represent words and non-terminal nodes represent constituents (Zwicky and Isard, 1963) (Figure 3.1). A word sequence is a constituent according to a phrase structure tree if there is some node $A$ that dominates every word in the sequence and no other words (Jacobson, 2005). For example, in Figure 3.1, '*pointy teeth*' is a noun phrase constituent that is immediately dominated by the verb phrase '*has pointy teeth*'. The notion of dominance is central to the identification of constituents and some terms used in this dissertation. Informally, a node $c_i$ dominates a node $c_j$ if there is a downward sequence of edges connecting $c_i$ and $c_j$ in the graphical representation of a phrase structure tree. More formally (Zwicky and Isard, 1963):

> **Definition:** *Dominance is the reflexive and transitive closure of the arc relation between two nodes $c_i$ and $c_j$ in a set of nodes C, such that if dominance is a relation R on C, then for all nodes $c_1$, $c_2$, $c_n$ in C, whenever $c_1$ R $c_2$ and $c_2$ R $c_n$ then $c_1$ R $c_n$.*

The node $c_i$ *immediately dominates* the node $c_j$ if $c_i$ dominates $c_j$ and no other node $c_x$ intervenes between them. For example, in Figure 3.1, '*teeth*' is immediately dominated by a noun phrase (NP) node (*pointy teeth*) and is dominated by a verb phrase (VP) node

(*has pointy teeth*) as well as a node representing the entire sentence. No single word dominates another (e.g. '*teeth*' does not dominate '*pointy*' or vice versa).

### 3.1.2 Varieties of phrase structure grammar

There are at least three aspects in which phrase structure grammars differ. First, they vary in how they handle sequences of text that appear to be constituents but have discontinuous elements (Jacobson, 2005). Some grammars relax the requirement that phrase structure rules describe sequential elements (Bach, 1979), but most are modelled on Chomsky's transformational grammar (Chomsky, 1957).

Transformational grammar retains the essential continuity of constituents that are units of contiguous words, or contiguous sequences of smaller constituents. Movement rules are used to accommodate discontinuity. They are part of the syntax of language and effectively shift part of a constituent (that is itself a smaller constituent) to another site within the sentence. They describe a sequence of derivations, or transformations, that map one phrase structure tree into another. These transformations occur across multiple levels of syntax, moving from DEEP STRUCTURES[4] that typically align with sentence semantics, to the final surface syntax that accounts for observed word order and sound.

Transformational grammar is often called generative grammar because it aims to succinctly describe, or generate, all grammatical sentences in a language. However, many grammars are generative, so the more descriptive term is 'transformational grammar'. Chomsky's theories have evolved several times since the introduction of transformational grammar in 1957, but all of Chomsky's subsequent theories, including Government and Binding (GB), the Minimalist Program (MP) (Chomsky, 1965, 1981, 1995), and the Principles and Parameters model of language, are variations of transformational grammar. Transformational grammar is therefore a prototypical, or 'standard', phrase structure grammar, and the most common one applied in IR.

Phrase structure grammars also vary in their branching constraints (Jacobson, 2005). In a prototypical phrase structure tree, a node can immediately dominate at most two other nodes. Grammars obeying this constraint are in Chomsky Normal Form (CNF), and can be described by relatively efficient computational parsing algorithms. Other, non-prototypical grammars such as Head Driven Phrase Structure Grammar (HPSG)

[4]Chomsky has since abandoned the idea of deep structure, or D-structure and its counterpart S-structure, in favour of Logical Form (LF) and Phonological Form (PF) (see Chomsky (1995) on Minimalism), although the concept of transformation remains central.

(Pollard and Sag, 1994) permit more than two immediately dominated constituents.[5]

A third point of divergence centers on whether the relationship between sibling nodes (nodes that are immediately dominated by the same node) is symmetric or asymmetric. In standard phrase structure grammars the relationship is symmetric. However, there are several influential theories in which this is not the case. In Categorial Grammar (Steedman, 1999), and related theories, grammatical categories are functions that map from one grammatical category to another, and the structure of a sentence is a representation of how grammatical rules work to define the sentence. For example, a verb phrase can be defined as a verb looking for a noun phrase to its right. The verb and the noun phrase are sibling nodes. Hence, the relation between sibling nodes is asymmetric because the noun phrase is the argument to the verb.

### 3.1.3 Parsers applied in this dissertation

The phrase structure parser used in this dissertation is an unlexicalized probabilistic CONTEXT FREE GRAMMAR (PCFG) parser (Klein and Manning, 2003a), using a grammar that is not in Chomsky Normal Form and assumes symmetric relations between sibling nodes. The parser was chosen for speed and because it is commonly used in IR research (Balasubramanian and Allan, 2009; Bendersky and Croft, 2008; Park et al., 2011; Xue et al., 2010; Zhao and Callan, 2010). The PCFG parser assigns a conditional probability to each expansion rule in the grammar based on observations of the rule in the Penn Treebank, a collection of gold-standard parsed data from the *Wall Street Journal* (Marcus et al., 1993). The probability of a parse tree is the product of the probabilities of all the rules used to expand nodes in the tree, and the tree with the highest probability is selected as the best parse. The PCFG parser does not leverage transformations or deep structure.

### 3.1.4 Suitability for IR

Transformational grammar is remarkably successful at describing grammatical rules underlying the construction of sentences. However, zealous enthusiasm for Chomsky's framework is counterbalanced by evidence and analysis from the fields of biology, psychology, and childhood development that vigorously question the degree to

[5]Grammars such as HPSG and Combinatorial Categorial Grammar (CCG) are non-prototypical in more ways than one. They remove transformations, and are sometimes considered lexicalist due to the inclusion of additional information in the lexicon. For more information see Jacobson (2005).

which Chomsky's ideas reflect any reality of human language processing. Oller (2008) claimed that Chomsky's approach, "suffered for decades from nagging failures to provide fully workable descriptions in any general domain of syntax", and that alternative theories, "are fundamentally preferable to the Chomsky-inspired approach... in cases where empirical data are examined at close range". Chomsky's theories have also been criticized by his followers in their attempts to resolve failures of his approach while retaining the thrust of his ideas (Oller, 2008).

Chomskyan theory has been repeatedly adjusted and adapted in response to criticism, and made to incorporate ideas from other areas of linguistics. For example, X-bar theory parallels dependency grammars (Chomsky, 1970; Haegeman, 2005) (see Section 3.2). The continuously shifting nature of Chomsky's theories makes it difficult to pin down a counter-argument to his ideas. Nevertheless, from an IR perspective several broad objections can be made to the core system of phrase structure rules defined over grammatical categories.

First, transformational grammar does not provide a complete account of all real-world language data. Chomsky sees language as a self-contained formal system, and believes linguistics to be properly confined to the study of an idealized language that is present in the human mind, independent of actual communication. This idealized language represents human language *competence* and is associated with "an ideal speaker-listener, in a completely homogeneous speech-community, who knows its language perfectly" (Chomsky, 1965). Chomsky contrasts language competence with language *performance* observed in text documents and speech recordings. Language performance is distorted by the limitations of human memory and perception, and the influence of imagination and social habit. Consequently, it is only of interest as a means to deduce the one, unique 'true' (or psychologically real) form of language. For Chomsky, language data are not the words we see and hear, they are the grammatical judgments of native speakers (Chomsky, 1965).

In short, phrase structure grammars emphasize idealized language and grammatical structure that may not adequately describe the reality of autonomous language encountered in IR. Queries are not necessarily well-formed sentences, and many open domain documents, such as forum posts and blogs, contain numerous grammatical irregularities. On this ground, phrase structure grammar is not expedient for IR.

Another concern is the degree to which Chomsky's discrete grammatical categories are central to human language. It is not clear that all languages share the same underlying grammatical structure, and indeed that any language can be neatly described

in this way. Chomsky argues that there must be a common structure underlying all languages because they reflect the same inherent human language ability (Chomsky, 2006). Grammatical units account for the intuitions of native speakers, and explain the productivity and systematicity of language. Based on his arguments, the leading view in modern linguistics is that all languages have some form of constituent structure (Jacobson, 2005). There is a set of grammatical categories from which languages may choose, and these categories are cross-linguistic (Haspelmath, 2007).

However, empirical data reveals that words cannot reliably be assigned to single grammatical categories (Lakoff, 1987). Francis (1993) notes that, "any one of a huge range of the more frequent words in English... has a unique grammatical profile, which certainly cannot be encapsulated by calling the word in question an adjective or a noun or a preposition". Instead, some items fit a given grammatical category better than others. In addition, even very basic grammatical categories do not seem to be crosslingual (Dryer, 1997; Haspelmath, 2007). For example, Mandarin Chinese does not appear to contain adjectives as a separate part of speech. By consequence, it seems that language is better described by *prototypical* categories, rather than *discrete* categories. There is a probability distribution over categories, and phrase structure parsing flattens this distribution by representing only the most general, or probable, case.

By consequence, when probabilistic IR techniques use grammatical categories they must compensate for an inadequacy of the underlying conceptualization of discrete categories at the same time that they optimize over data. This is particularly striking since there is no compelling reason to expect that grammatical categories should be central to IR.

Indeed, there is evidence from IR that suggests grammatical categories are not central to search processes. For example, stemming improves the match between queries and documents by discarding information about grammatical categories. In stemming, words that are inflections (e.g. plurals, tenses) or derivations (e.g. transformations of a verb into a noun) are normalized to facilitate matching of words with similar semantic meaning, e.g. {*argument, arguments, argue, argues, arguing*}. Stemming conflates grammatical categories such as verb and noun to remove irrelevant distinctions.

Brants (2004) also observed that low level grammatical categories, in the form of parts-of-speech, can be unreliable features for the description of single words. This is because gains in IR effectiveness attributed to the separation of meaning using grammatical categories are often offset by losses incurred by separating words with similar meaning. On the one hand, part-of-speech tags can indicate distinct meanings for

words with the same spelling, as in the example of *building*/Noun, which refers to an architectural construction, and *building*/Verb, which refers to an action of creation or improvement. On the other hand, any distinction made between *snow*/Noun and *snow*/Verb may have a negative impact on IR because events of snowing, and snow that falls from the sky, identify roughly the same semantic space.

In response to linguistic evidence, Lakoff, one of Chomsky's most vocal critics, argues that the foundations of phrase structure theory are called into question by the failure of grammatical categories as originally conceived. He asserts that the theory of categorization depends on an "objectivist paradigm", such that two items should be in the same category if they share properties, yet "The use of the objectivist paradigm in empirical semantic studies is simply an article of faith... the objectivist paradigm does not even come close to working" (Lakoff, 1987). This objection indicates that, as stated by Hockett (1968):

> "[Chomsky's] views are largely in error, but they are too powerful to be shrugged aside. It is necessary to meet Chomsky on his own ground. When we do this, we discover that, even if he is wrong, his particular pattern of error tells us some things about language that were formerly unknown".

With respect to IR, debate about the validity of Chomsky's category assignments may explain conflicting results observed with the application of features of phrase structure grammars in IR techniques. Specifically, if grammatical categories have a prototypical or probabilistic nature, then we expect that probabilistic approaches to language processing may be successful (Manning, 2007) where discrete approaches are not. Indeed, successful applications of phrase structure for IR, such as machine learning with grammatical categories, are probabilistic. Less successful approaches, such as noun phrase indexing, use discrete category assignments. Thus, the nature of grammatical category assignments may account for variable success with phrase structure representations for IR.

### 3.1.5 History of phrase structure for IR

Chomsky inspired a generation of young linguists coming out of graduate school with an elegant, homogeneous theory of language that replaced the complex discovery procedures of Structuralism. This framework could generate, and thus explain, complex language data using a few simple rules and a limited number of grammatical categories. It brought natural language closer to the formal language of mathematics, and took "a

major step toward restoring the traditional conception of the dignity and uniqueness of man" (Searle, 1972).

The publication of Chomsky's "*Syntactic Structures*" (Chomsky, 1957) in 1957 sparked a revolution in linguistics from which arose the heady conviction that "man's languages could be reduced to mathematical principles" (Grefenstette, 1998). This took linguistics by storm, and spread enthusiasm for this revolutionary new theory to IR. In the 1960s it seemed that, "Phrase structure, as the name indicates, accounts for the most important word groupings," and that, "These groups are also those which make up the basic components to be included in a useful [semantic] information graph" (Salton, 1964b).

A hypothesis began to develop within the IR community that noun phrases are the ideal text unit to represent documents, while use of phrase structure annotations cemented their acceptance as the dominant method for 'linguistic processing'. Noun phrases were thought to be good indicators of semantic content while other phrases were thought to be misleading or too general (Salton, 1964b). Work with noun phrases for document indexing appeared as early as 1958 (Baxendale, 1958) and by the mid 1960s, phrase structure was applied in the SMART system (Salton, 1966).

In 1966, shortly after Chomsky's second major publication, a pivotal National Academy of Science (NAS) report condemned the unfulfilled promises of machine translation while praising Chomsky's "revolution in linguistics" (Searle, 1972). The report precipitated a sharp increase in government funding for work on phrase structure theory (Nevin, 2010; Marcotty, 1996). It also slowed progress in IR (Committee, 1966; Marcotty, 1996) by causing a drop in funding for IR, particularly for methods that used non-Chomskyan, or statistical, natural language processing techniques (Spärck Jones and Abbate, 2001).

By 1990, the fashion for Chomskyan linguistics meant that both prepositional and noun phrases were used to represent the texts of documents and search requests, even though the IR community had discovered quite early on that statistical phrases tend to produce more correct indicators of document content (Salton, 1966). Indeed, Salton (1964b) pointed out that:

> "Some important linguistic phenomena do not fit into a phrase structure model, even if extended to handle special cases such as discontinuous constituents... There is no way in a phrase structure model to relate, for example, two semantically identical sentences of which one is in the passive and the other in the active voice."

Nevertheless, the association of queries and documents with elements of phrase structure led to some improvements in retrieval effectiveness, and this reinforced the (possibly incorrect) belief that such syntactic annotations are the most appropriate focus for further 'linguistic' research. For example, the FASIT system showed improvement in IR effectiveness when using pre-defined patterns of grammatical word categories for document indexing compared to stemmed, or thesaurus-based, indexing (Dillon and Gray, 1983). The results were reported for only 22 queries run against a small database of 250 documents. More comprehensive experiments by Fagan (1987) showed that phrases identified with syntactic rules are comparable in effectiveness to statistical phrases, but are more complex, and therefore less preferable as a means of language analysis (Salton, 1989).

Some retrieval strategies combined phrase structure with headedness information from dependency theory, but these studies were usually small and used a discrete notion of grammatical categories (Brants, 2004; Fagan, 1987; Lewis and Jones, 1996; Salton and Smith, 1990; Song and Croft, 1999). They did not deliver the hoped-for gains in effectiveness. For example, Spärck Jones and Tait (1984) converted constituent word patterns into dependency representations (a many to one operation), and applied pattern matching in reverse to infer variant expressions (a one to many operation, from dependency representation to surface word order). The process was not particularly robust. It generated phrases such as '*retrieval by information*' for the source phrase '*information retrieval*'.

Metzler and Haas (1989) proposed to match parse trees in queries and documents using a type of headed phrase structure parsing, but results with this Constituent Object Parser (COP) were never reported. Phrases defined by Lewis and Croft (1990) were composed of the heads of phrase structure constituents that occurred within the same clause boundary connected by a grammatical relation. These phrases were extracted from 1425 documents and supplemented by a limited number of heuristic phrases created from the heads of neighbouring constituents, such as a verb phrase adjacent to a noun phrase. The approach captured many dependency relations between constituent heads but did not produce significant improvements in IR effectiveness on the CACM retrieval collection[6] (3204 documents and 50 queries). Further experimentation with phrase clusters fared no better.

The lack of substantial improvements with these techniques resulted in criticism

[6]CACM is a collection of abstracts of articles published in the 'Communications of the ACM' journal between 1958 and 1979.

of natural language processing, particularly with respect to the application of noun phrases for search applications. Typical sentiments included:

- "A major reason why previous attempts have had limited success at using syntactic information to improve information retrieval performance is that they have often not utilized the appropriate aspects of syntactic description. There seems to be relatively little to be gained from determining, for instance, whether two terms are in the same noun phrase." (Metzler and Haas, 1989);

- "The linguistic sophistication of the phrase generation process appears to have little effect on the performance of the resulting phrase representation." (Lewis and Croft, 1990);

- "Unfortunately, no matter how the problem is simplified, the analysis of noun phrase constructions, which is chiefly needed in information retrieval, is especially difficult, and all the various attempts to come up with general rules for noun phrase understanding have been unsuccessful... When syntactic methods are used for the generation of content-identifying phrases, the retrieval results are often discouragingly poor." (Salton and Smith, 1990).

Overall, it appears that methods that do not assume any syntactic knowledge are surprisingly hard to beat. It has been suggested that syntactic phrases are too low frequency to make good document identifiers (Lewis, 1992). They may also provide a suboptimal level of granularity. Query segmentation shows that informative word associations do not necessarily align with constituents. For example, '*San Jose international airport*' is a single noun phrase, but not all documents referring to the international airport in San Jose will use that exact phrase, especially if the airport is mentioned repeatedly. Rather, it can be usefully segmented into '*San Jose*' and '*international airport*'.

In the face of such unfavourable indicators, alternative applications of phrase structure in IR were explored. Recent techniques mix phrase structure parsing with discriminative, frequency-based criteria, and often take a prototypical view of grammatical categories. Results have been encouraging. Srikanth and Srihari (2003a) show that concepts consisting of contiguous words in constituents outperform concepts identified by bigram word co-occurrences. The Key Concept retrieval model by Bendersky and Croft (2008) filters noun phrases with decision tree learning using frequency-based features from external resources, and applies them in a weighted linear feature model. This weighted model outperforms a dependence model for IR that uses simple bigram word co-occurrences. Approaches using low-level grammatical categories have also improved over a query likelihood baseline. For example, Lease et al. (2009) use part-

of-speech tags, frequency counts and other features in a learning to rank framework for estimating term weights.

Despite their success, these approaches ignore the more fundamental problem that grammatical categories are identified from surface representations of text. These structures cannot make certain semantic distinctions, as observed by Chomsky and others (Chomsky, 1957). For example, Searle (1972) notes that it not not clear from the syntax alone whether the sentence '*I like her cooking*' means: *I like what she cooks, I like the way she cooks, I like the fact that she cooks*, or *I like the fact that she is being cooked*. We can conclude that because the semantics of queries and documents are pivotal in IR, the grammatical features of surface representations might not offer any benefit over a non-linguistic approach with respect to accurate semantic description (see Chapter 4 for more on the semantics of phrase structure grammars).

Unfortunately, as the attention of the IR community is drawn away from examination of linguistic theory, and towards probabilistic approaches to NLP, the failure of syntax to adequately capture semantics is often ignored. Instead, the utility of grammatical categories in computer algorithms reinforces their popularity. As Lakoff (1987) observes:

> "One of the reasons why the classical theory of categorization is becoming more, rather than less, popular, is that it is built into the foundations of mathematics and into much of our current computer software. Since mathematical and computer models are being used more and more as intellectual tools in the cognitive sciences, it is not surprising that there is considerable pressure to keep the traditional theory of classification at all costs. It fits the available intellectual tools, and abandoning it would require the development of new intellectual tools. And retooling is no more popular in the academy than in industry" (Lakoff, 1987).

The deficiencies of phrase structure grammars in the context of IR suggest that they are primarily used for four reasons. First, their familiarity and dominance in modern linguistics; second, the suitability of discrete categories for feature-based computational algorithms; third, the ubiquity of parsers trained on phrase structure annotations;[7] and fourth, the variable gains observed when features of phrase structure representation are used in probabilistic approaches. Notably, these probabilistic approaches typically include discriminative frequency-based features of text as well. Phrase structure remains a dominant means of language representation in ad hoc IR today.

[7] Most automatic parsers used for IR were trained on the Penn Treebank (Lease, 2007; Marcus et al., 1993).

### 3.1.6 Summary

Grammatical categories are familiar, well-defined and clearly understood. The rules of phrase structure grammars provide an elegant framework for the grammatical analysis of language, and the categories they use are well-suited to computational algorithms, including machine learning and optimization frameworks. These benefits have helped to make phrase structure grammars the dominant method of language representation for IR, as well as in other areas of natural language processing.

However, phrase structure grammars have a number of disadvantages for IR. First, they were never intended to offer a complete account of all language data. In Chomsky's view, manifestations of language in the real world, such as the data input to IR systems, are unimportant except in their provision of evidence for idealized language competence.

Moreover, there is the question of whether grammatical categories adequately describe language. It is not clear that languages share the same underlying grammatical structure, or that words and phrases participate in discrete category assignments. The former may be a concern in cross-lingual IR. The latter may adversely affect the performance of techniques that use a discrete notion of grammatical categories. In contrast, a probabilistic framework that captures a prototypical notion of grammatical categories helps to avoid occasional mistakes associated with the application of phrase structure in IR.

Probabilistic approaches that use phrase structure also perform reasonably well. However, phrase structure theory does not focus on semantics, so it fails to identify certain word associations with nearly equivalent meaning. The semantic characterization of phrase structure grammars is discussed further in the next Chapter. For now, it is sufficient to make an analogy between the use of grammatical categories for IR and the application of a statistical model to a task in which the data features of interest do not match the underlying model assumptions. Such a statistical model may be robust enough to perform well even if the model is not a good approximation for the data distribution. However, the application is inherently flawed, and a better model will likely result in improved task performance. Likewise, we might expect that the results achieved in IR with an inappropriate model of word association will be somewhat lacklustre and variable, even if the model is an excellent fit for some other application. A model that better captures the distribution of relevant word associations in IR should achieve improved, or more stable, retrieval effectiveness.

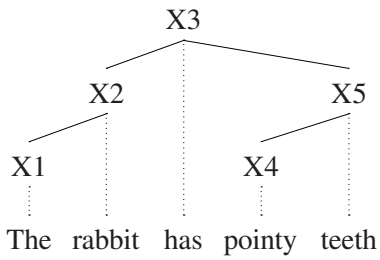



Figure 3.2: A labelled dependency tree.

## 3.2 Dependency theory

### 3.2.1 First principles

Modern dependency theory is the culmination of a long heritage in language analysis. It is based on the dominance of one word over another, a notion formalized by Frenchman Lucien Tesnière with his posthumous publication in 1959 (Fraser et al., 1993; Nivre, 2005; Tesnière, 1959). Tesnière's dependency grammar appeared just two years after Chomsky's first book on transformational grammar, and inspired the development of further dependency grammars, including Functional Generative Description (FGD) (Sgall et al., 1986), Meaning Text Theory (MTT) (Kruijff, 2005) and Word Grammar (Hudson, 1987).

Dependency grammars assume that syntactic structure is composed of lexical elements linked by binary asymmetrical relations called dependencies. This structure forms a directed graph, in which each arc holds between a *head* (or *governor*) and a *dependent* node. The graph has only one root node that is not dominated by any other node, and is usually acyclic, taking the form of a tree in which each node only has one governor (Figure 3.2). The most salient feature of dependency structures is the lack of non-terminal phrasal nodes seen in phrase structure trees. In a constituency framework, words combine with a sister relation to form greater units, whereas in a dependency framework, words combine with a parent-child relation to form greater units in which no higher node is generated.

The definition of a dependency relation varies between dependency grammars, but always contains a core notion of valency. Valency refers to requirements and restrictions placed on the type and number of arguments given to a word. It is related to TRANSITIVITY that describes the syntactic requirements and restrictions for verbs. For example, ditransitive verbs, such as '*gives*', fill two syntactic slots for objects, resulting in phrases like '*David gives John data*'. Here, '*John*' is the direct object of '*gives*', and

'*data*' is the indirect object. Valents overlap with transitive relations for many verbs but differ in several important ways.

First, valency accounts for the subject position whereas transitivity does not. The subject is usually the actor in a sentence, so for the sentence '*David gives John data*', '*David*' is the subject and '*gives*' has three valents, namely '*David*', '*John*' and '*data*'. Second, valency is not restricted to verbs. Nouns and adjectives derived from verbs may partly retain the valency of their verb forms. For example, the adjective-noun combination '*farming operations*' has '*operations*' as the governor, and '*farming*' as the dependent. This is derived from the valency of '*operate a farm*' that has the verb '*operate*' as the governor and '*farm*' as the dependent. Valency also accounts for types of arguments, whereas transitivity does not.

Dependencies in dependency grammars can be syntactic and semantic (Nivre et al., 2007; Schneider, 2007). Syntactic structures include head-modifier relations, such as adjective-noun and noun-noun combinations. These combinations are derived from the valencies of verbs, as we just observed for the adjective-noun combination '*farming operations*'. Noun-noun combinations, such as '*farm operations*', can be derived in a similar way.

Semantic structures described by dependency grammars include head-complement relations. Head-complement structure is synonymous with predicate-argument structure identified during semantic role labelling, except in a minority of linguistic theories that do not interpret sentence subjects as complements (like *David* in the example above). This alignment enables two observations. First, dependency parsing has a semantic interpretation because predicate-argument relations have a semantic interpretation. Specifically, semantic assignments for predicate-argument structures include agent, source, theme and goal (Jackendoff, 1972). Second, semantic role labelling can be considered a subtask of dependency parsing in which the types of semantic roles are identified. Note, however, that a sentence may contain semantic relations that a parse for predicate-argument structure does not detect. This is because arguments must be predicted, or required, by a predicate to be detected by predicate-argument structure. Not all semantic dependents are predicted.[8]

[8] Arguments are a special case of default dependency relations, known as adjuncts, that are not required. Some adjuncts with a semantic function are not arguments. For example, the adjunct '*last year*' in the sentence '*who ran last year?*' is a temporal locator (a type of complement), and not an argument.

### 3.2.2 Varieties of dependency grammar

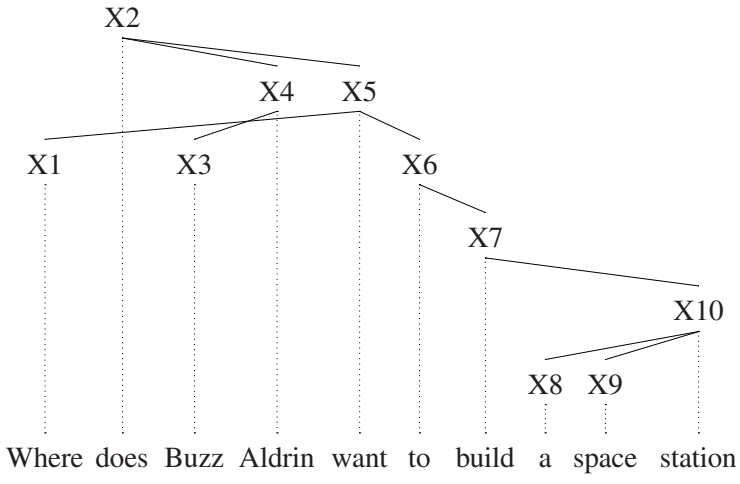
Dependency grammars differ in a number of ways. One major difference is the number of levels used for language analysis. A dependency parse generated by an automated parsing system is usually a monostratal structure that combines syntactic and semantic dependencies in a single layer. Monostratal dependency grammar was first proposed by Tesnière (Tesnière, 1959) and produces a parse structure that is usually conceived of as syntactic, even though it clearly has semantic properties.

Monostratal dependency grammars are ubiquitous in automated parsing, but most dependency grammars distinguish between syntactic and semantic dependencies. Multistratal dependency grammars are inspired by the distinction between deep structure and surface structure in transformational grammar. They recognize semantic and syntactic relations in separate layers of the dependency representation, and may reserve an additional layer for morphological dependencies. Morphological dependencies occur when morphemes are linked by dependencies instead of words (Mel'čuk, 1988), and are particularly appropriate for languages such as Arabic that have a rich morphology.

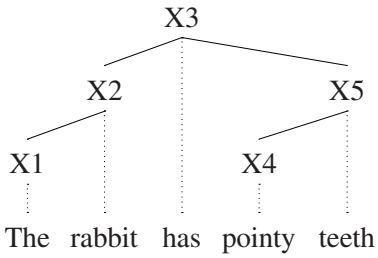
A second difference concerns the way that dependency grammars handle relations between discontinuous words. Most dependency grammars allow the order of nodes in a dependency structure (from left to right) to diverge from the surface word order, although whether word order nonetheless represents a constraint on headedness is "hotly debated" (Nivre, 2005) (see the Adjacency Principle in this Section). In addition, most dependency grammars used for automated parsing are *projective*. Projective grammars do not permit word dependencies to CROSS. For every edge between a governor node $h$ and a dependent node $d$, the node for a word $w$ occurs between $h$ and $d$ in surface text only if $w$ is also dominated by $h$. In Figure 3.3 (b), the dependency tree is projective because, for example, *pointy* occurs between *has* and *teeth* in surface word order, but *has* is the governor of *pointy* and also the governor of *teeth*. Most practical systems for dependency parsing are projective because projectivity constrains the parse space and makes parsing more efficient.

In contrast, most theories of grammar are *nonprojective* because it is fairly common to observe linguistic constructions that cannot be represented by a projective parse (Nivre, 2005). In the Penn Treebank (Marcus et al., 1993),[9] for example, 7.6% of sentences include at least one non-projective link (Johansson and Nugues, 2008). Figure 3.3 (a) shows a nonprojective dependency tree with crossed dependencies.

[9]The Penn Treebank is an annotated corpus that contains over 4.5 million words from the *Wall Street Journal*.




(a) *Nonprojective dependency tree.*




(b) *Projective dependency tree.*

Figure 3.3: (a) A nonprojective dependency tree may have crossing dependencies; (b) A projective dependency tree does not contain crossing dependencies.

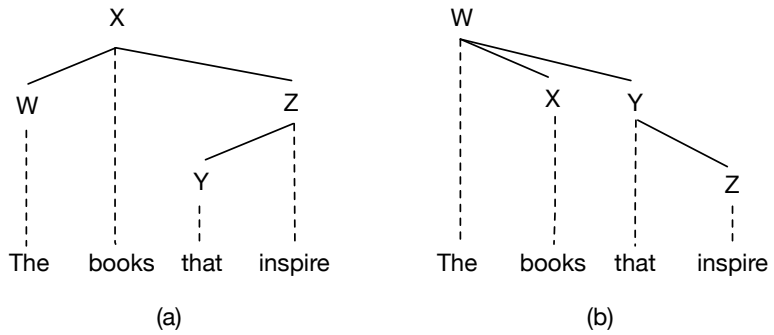



Figure 3.4: The choice of how governor and dependent roles are assigned within a dependency relationship impacts the dependency structure; (a) Tesnière's framework (Tesnière, 1959); (b) Hudson's framework (Hudson, 1987)

The third major source of variation concerns how headedness is defined, which in turn affects the structure of a dependency parse. Headedness is determined with reference to valency for straightforward cases of head-modifier and head-complement relations. However, dependencies involving function words are problematic, and it is common to observe variation in the output of automated dependency parsers for the same sentence (Buyko and Hahn, 2010). As shown in Figure 3.4, different assignments of governor and dependent roles result in different dependency structures.

Most automated dependency parsers use a variant of Tesnière's monostratal dependency grammar, and also follow his approach for the definition of dependencies involving function words. In Tesnière's framework, a content word is the nucleus of its function words, so a noun is the governor of its determiner, a verb is the governor of its auxilliary, and so on (Tesnière, 1959). This is shown in Figure 3.4 (a) for the phrase '*the books that inspire*'. In contrast, the monostratal dependency grammar proposed by Hudson (1987) reverses Tesnière's assignment of governor-dependent labels for some word combinations. His conception is shown in Figure 3.4 (b).

Hudson argues that function words can govern content words based on a semantic criterion and requirements of word order. The semantic criterion holds that if $X$ is a semantic governor, and $Y$ is the dependent, then $X+Y$ specifies a *kind of* the thing described by $X$. For example, *books* is the governor of '*those books*' because the phrase describes a kind of book, and *read* is the governor of '*will read*' because the phrase describes a kind of reading. In Figure 3.4, *that* is considered the semantic governor of '*that inspire*' because in the context '*the books that inspire*', the phrase '*that inspire*' indicates a kind of purpose or reason, and not a kind of inspiration. This view contrasts with Tesnière's approach in which the content word '*inspire*' is the governor of the function word '*that*'.

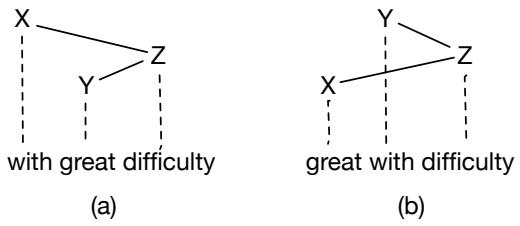



Figure 3.5: The acceptability of the word order in (a), and the unacceptability of (b), is determined by the Adjacency Principle. Figure adapted from Hudson (1987).

Hudson's argument about word order is based on his *Adjacency Principle* which asserts that a dependent should appear as close to its governor as possible in the linear order of text (Hudson, 1987). This principle is similar to the ordering principle for well-formed strings in dependency grammars described by Robinson (1970). It states:

> **Adjacency Principle:** *If A is the head of B, and some word C separates them, then it must be the case either (i) that C is subordinate to A, or (ii) that C is (subordinate to) the head of both A and B.*

In this definition, *A* is subordinate to *B* if *A* is dependent on *B*, or if *A* depends on *C* and *C* is subordinate to *B* (Hudson, 1987). The consequences of this principle with respect to determination of acceptable word order are illustrated in Figure 3.5. Both (a) and (b) show that *difficulty* depends on *with*, and *great* depends on *difficulty*, as determined by the subphrases '*with difficulty*' and '*great difficulty*'. Hudson argues that the word order in (a) is acceptable because it adheres to the Adjacency Principle while the word order in (b) does not. Specifically, in (a), *difficulty* is separated from its governor by *great*, but *great* is subordinate to the same governor, satisfying clause (i). In (b), *difficulty* is separated from its governor by *with*, but *with* is not subordinate to the same governor, or subordinate to a head of both *great* and *difficulty*. For this reason, (b) is unacceptable.

An argument for determiners being the governors of nouns can be presented along similar lines, with reference to word order requirements for phrases like '*the red books*'. Namely, if '*books*' is the head of '*the*', then the dependency between them would be the same as between '*red*' and '*books*'. There would be no explanation for the observation that a determiner always precedes an adjective (i.e. not '*red the books*'). This view, of course, contrasts with Tesnière's approach in which the content word *books* is the governor of the function word *the*.

The point is that reasoned arguments exist for assignments of governors and de-

pendent roles in dependency grammars that sometimes differ markedly. There are also a significant number of potential syntactic and morphological criteria for headedness that might affect the choice of governor-dependent pairs. Zwicky (1985) examined eight such criteria for the identification of headedness for phrase structure trees. Six criteria are syntactic or morphological and are listed in Table 3.1. The remaining two criteria are semantic. They hold that a base constituent is the governor of another base constituent if it is interpreted this way in a dependency grammar, and that a constituent is a governor if it is a functor, such as a predicate, that takes semantic arguments.

In short, there is a common core understanding of dependency based on valency relations, but a sentence can have multiple dependency interpretations that differ significantly in certain respects. In fact, it is not necessary that a dependency grammar identifies all words as participating in governor-dependent relations, resulting in incomplete syntactic analyses (Nivre, 2005).

A final difference is the number and type of dependency relations defined. Most theories label dependency types, but the labels are often discarded when dependencies are used in computational language processing (Nivre, 2005). In addition, the differences are usually small. Most annotation schemes define around 50 relations. LINK GRAMMAR identifies 106 relation types, and is very similar to a dependency grammar, but differs in three ways: it can be cyclic, there is no notion of a root word, and its links are undirected (Buyko and Hahn, 2010; Sleator and Temperley, 1993).

### 3.2.3 Parsers applied in this dissertation

The two dependency parsers used in this dissertation vary in their projectivity constraints, the types of relations that they define, and the structure of the dependency parses that they produce (their headedness criteria). Both implement monostratal dependency grammars based on Tesnière's framework, such that function words are the dependents of content words.

The Stanford probabilistic context free grammar (PCFG) dependency parser (henceforth, the Stanford parser) is applied in most experiments in this dissertation. It combines dependency relations extracted from a headed phrase structure parse produced by the unlexicalized, projective PCFG parser described in Section 3.1.3, and a separate dependency parser (Klein and Manning, 2003b). CFGs assume node expansions to be independent of one another, but the selection of an expansion rule can be profitably constrained by dependency structure. For example, if a verb can take a location as

| Headedness criteria | Example |
|---|---|
| *H* determines concord, such that co-constituents must agree. | In the phrase '*two books*', '*books*' must agree with '*two*'; it is incorrect to use '*two book*'. |
| *H* is the morphosyntactic locus that bears inflectional marks of the syntactic relations between the *C* and other syntactic units, such that inflections on *H* percolate information up to higher order constituents. | In '*read these books*', the inflectional marking of person and number occurs on '*read*' (read/reads), indicating that '*read*' is the governor. The inflections indicate how other syntactic units must agree with *C*, ruling out statements like '*he read these books*'. |
| *H* is subcategorized with respect to its sisters. A constituent is subcategorized when its ability to appear with other constituents is constrained, while the other constituents are not so constrained. | '*Give*' is subcategorized to occur with either *NP NP* as in '*give John the book*' or *NP* to *NP* as in '*give the book to John*', therefore '*give*' is the governor of the sentence '*give John the book*'. |
| *H* selects the morphological form of its sisters. | The morphological form of a personal pronoun '*them/they*' would depend on the head verb '*read*', as in '*read them*' and not '*read they*'. |
| The distribution of *H* is identical to that of *C*. | The name '*John F. Kennedy*' appears in roughly the same contextual distribution as '*Kennedy*', indicating that '*Kennedy*' is the governor. |
| *H* is obligatory, such that its removal forces a construct in which it appears to be recategorized. | The removal of '*give*' in the construct '*give the book*' forces it to be recategorized as a noun phrase, rather than a verb phrase. |

Table 3.1: Syntactic and morphological criteria proposed by Zwicky (1985) for a governor *H* and a dependent *D*, forming a construct *C*.

a semantic dependent then it is more likely that a prepositional phrase will attach to that verb. For this reason, the approach combines the output of a phrasal parse and a dependency parse in a factored model.

The second parser is the top-performing system from the Conference on Computational Natural Language Learning (CoNLL) 2008 shared task for joint syntactic-semantic analysis (Johansson and Nugues, 2008) (henceforth, the joint dependency parser) and uses the CoNNL dependency format as applied by most native dependency parsers. Like the Stanford parser, it has syntactic and semantic subsystems, but performs joint dependency parsing and semantic role labeling, rather than phrase structure parsing. The syntactic component is a bottom-up projective dependency parser using pseudo-projective transformations, which means that non-projective links are lifted up a parse tree to achieve projectivity, and trace labels are used to recover non-projective links at parse time. The semantic component uses global inference on top of a pipeline of classifiers. The final parse is selected by re-ranking a short list of candidate parses generated by each component. This approach aims to improve performance on separate tasks of dependency parsing and semantic role labeling by performing them together. The parser is applied in Chapter 6 where semantic labels are used in feature generation.

These parsers differ in the number and type of relations they identify. The Stanford parser identifies 48 dependency types that can be output in a collapsed format (de Marneffe et al., 2006). This format narrows the distance between content nodes by the removal of prepositions, conjunctives and other function words. For example, the phrase in (3.1) below is collapsed to the phrase in (3.2). The collapsed dependency format is used for both parsers in this dissertation. The output of the joint dependency parser is converted to a collapsed format in a separate step.

(3.1) $words \overset{nmod}{\longrightarrow} in \overset{pmod}{\longrightarrow} documents$

(3.2) $words \overset{prep\_in}{\longrightarrow} documents$

The joint dependency parser uses the CoNLL-X dependency format, which is applied by most dependency parsers. This is based on a constituent to dependency conversion of the Penn Treebank and applies 54 dependency types (Nivre et al., 2007). The main difference between the CoNLL-X and Stanford dependency formats is the representation of passive constructions and auxiliary and modal verbs. The Stanford format takes auxiliaries to be the governors in passive constructions, whereas the CoNLL-X format assigns this function to main verbs (Buyko and Hahn, 2010).

Certain semantic relations identified by the Stanford parser may not be available with the CoNNL representation. The main difference between the two outputs is the representation of passive and tense verb constructions, including auxiliary and modal verbs. The Stanford parser takes main verbs to be governors, whereas the CoNLL format takes auxiliaries to be governors. Figure 3.6 (a) shows that '*heard*' is the root of the tree generated by the Stanford parser, while (b) shows that '*were*' is the root of the tree generated by the joint dependency parser. The pair '*types heard*' is directly available from the Stanford parser, whereas the joint dependency parser identifies '*types were*'. In the case of query $q$, the collapsed governor-dependent pairs are the same for both parsers. This often occurs because support verbs are typical stopwords. However, there may be differences between the parse output for some queries.

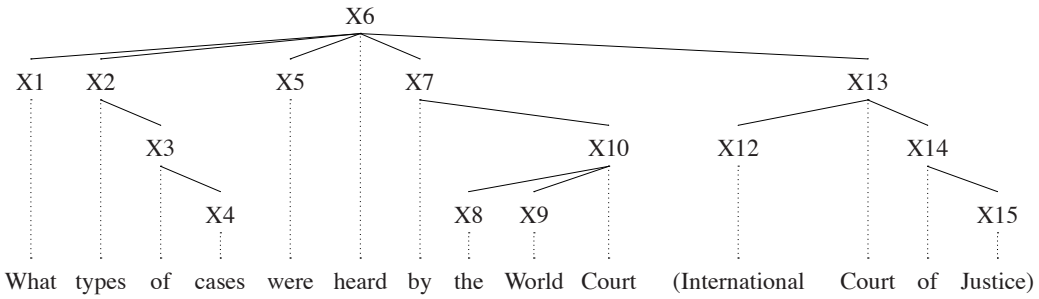
### 3.2.4 Suitability for IR

For a long time, dependency grammars were marginalized from mainstream linguistics, at least in part due to the belief that they are weakly equivalent to phrase structure grammars (Nivre, 2005). Interest in dependency grammars was primarily stimulated by Chomsky's work on X-bar theory (Chomsky, 1970; Hudson, 1987) that adopts a core notion of dependency. X-bar theory assumes that all major syntactic and functional grammatical categories project the same structure of a head $X$ and its complement or specifier.

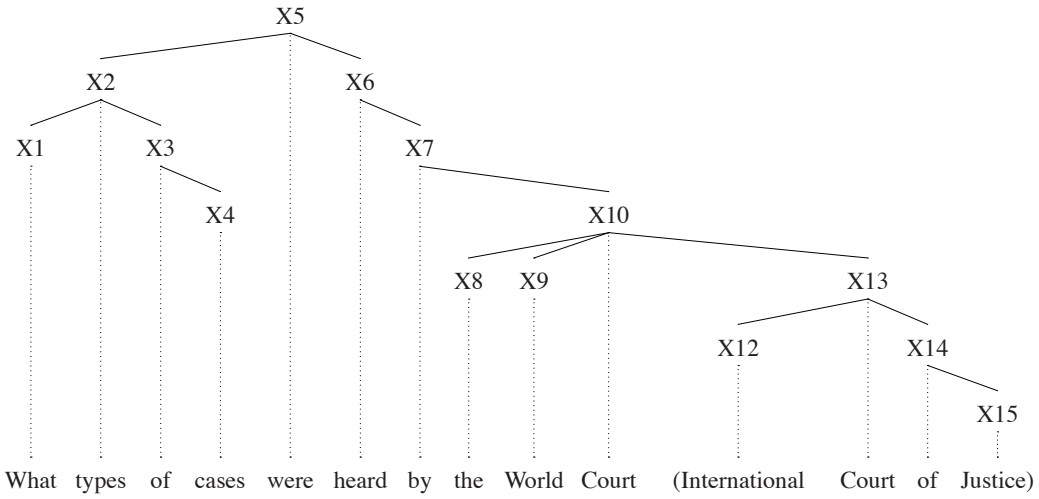
Chomsky's proposal precipitated a broad desire within the linguistics community to integrate the intuitive semantic ideas of dependency grammars with the logical account of syntax provided by phrase structure grammars. A quite natural assumption was that there is little to be gained from preferring one general account of language to the exclusion of another (Chafe, 1968). Language analysis is properly applied to both entities, such as constituents, and relationships between them, as targeted by dependency grammars.

This way of thinking had the advantage of spurring attempts to account for significant observations of both theories. It was observed that given criteria to determine headedness, constituent trees can be converted into a dependency structure. Moreover, constituents can be recovered from a dependency structure by interpreting each node in a dependency tree as the head of a constituent. A dependency constituent is composed of all words in a dependency subtree headed by a given node (Rambow, 2010).

The downside of this conversion between phrase structure and dependency trees




(a) Stanford format




(b) CoNNL format

| | |
|---|---|
| types cases | heard types |
| world court | heard court |
| court court | court international |
| court justice | |

(c) Head dependent pairs

Figure 3.6: Illustration of (a) a Stanford dependency parse, (b) a CoNNL dependency parse, and (c) collapsed governor-dependent pairs extracted from either parse.

was a suggestion that dependency parsing offers no practical benefit as an independent alternative to phrase structure parsing. This view was reinforced by Gaifman's widely-read paper that claimed dependency grammars to be weakly equivalent[10] to phrases structure grammars (Gaifman, 1965; Nivre, 2005). Attempts to integrate dependency theory into the dominant phrase structure framework tended to overestimate the importance of syntax, and underestimate the benefit of a practical focus on semantics, at least for some language processing tasks. Specifically, most accounts focused on a rationalization of semantic dependencies within a structure originally intended to describe surface word order, rather than a rationalization of how surface word order might result from a structure of semantic dependency.

The difference may be crucial for applications such as IR that seek to leverage semantic information. The former approach, taking phrase structure to be primary, requires some external knowledge in the form of transformation rules to move from a semantic representation to the surface appearance of text. The latter approach, as illustrated by Hudson's Adjacency Principle and a lexicalist approach to semantics (see section 3.3), benefits from an assumption that no complex additional knowledge is required to explain surface word order. Rather, word order is implicit in semantic structure given a normal constraint that related things should be close to one another in time or space.

It seems obvious that the latter approach might be preferred for a system that matchs the semantics of queries and documents without knowledge of complicated transformation procedures. However, such an approach requires confidence that there is a benefit to language analysis with dependency grammar quite independent of a phrase structure framework. It is precisely the intuition that dependency grammars are essentially interchangeable with phrase structure grammars that held back such independent investigation of dependency grammars for IR.

Importantly, it has since been demonstrated that the weak equivalence claimed for phrase structure grammars and dependency grammars may have been misinterpreted. Phrase structure grammars do not subsume dependency grammars, or vice versa (Abney, 1995). This is because there is no unique equivalence between the two. Figure 3.7 shows that the dependency trees in (a) and (b) are different, but induce the same unlabeled phrase structure tree in (c).

A current view is that both phrase structure grammars and dependency grammars are equivalence classes of Headed Context Free Grammars (HCFG) (Abney, 1995).

[10]Two grammars are weakly equivalent if they characterize the same set of strings.

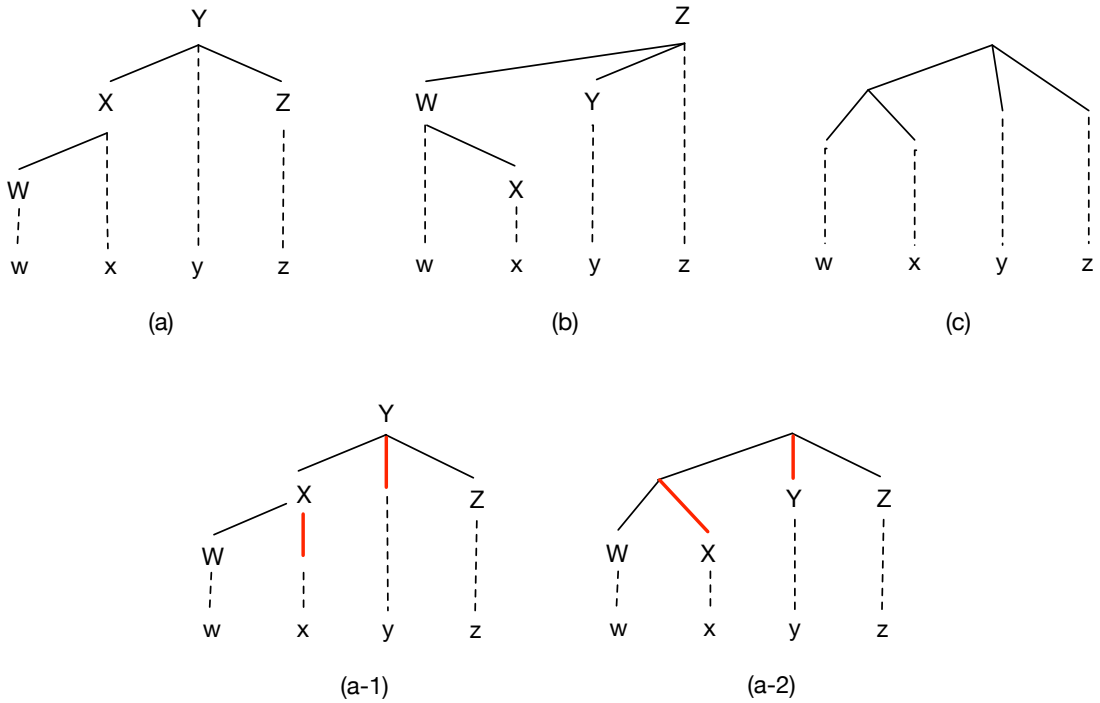



Figure 3.7: Phrase structure grammars and dependency grammars are not uniquely equivalent. The different dependency trees in (a) and (b) are both represented by the same phrase structure tree in (c) (Figure taken from Abney (1995)). The derivation of (c) from example (a) is shown in (a-1) and (a-2). Phrase structure representations always assign words to leaf nodes. So, for every non-leaf node that represents a word, add an edge and move the word symbol to the leaf position. The structure in (c) results.

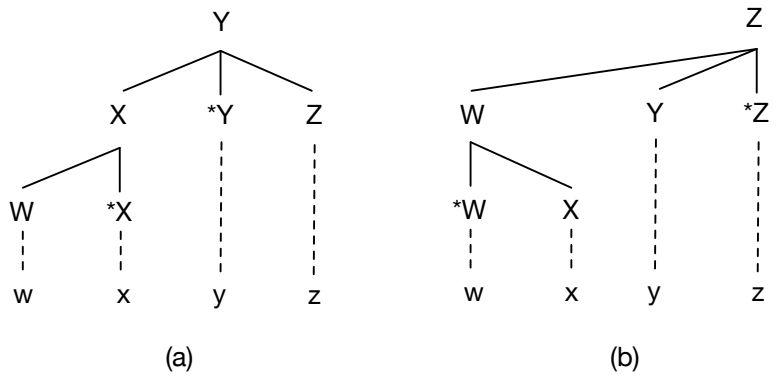



Figure 3.8: A phrase structure tree labeled with head information corresponds to a unique dependency parse (Figure taken from Abney (1995)).

Context free grammars, such as phrase structure grammar, are represented by a finite set of rules of the form $X \rightarrow Y$ where $X$ and $Y$ are low level grammatical categories (Hays, 1964). HCFG grammars include information about both headedness and word order. Standard phrase structure grammars abstract away from the choice of governors in an HCFG, and dependency grammars abstract away from word order.

A HCFG corresponds to a unique dependency parse and a unique phrase structure parse. Likewise, a phrase structure tree labeled with head information corresponds to a unique dependency parse, as the example in Figure 3.8 shows. However, in order to generate a dependency parse from a phrase structure parse, the equivalence relations of the phrase structure grammar must be extended with a convention for marking heads that is not native to phrase structure grammar.

Of course, whether dependency grammars correspond with HCFGs or phrase structure grammars is pragmatically inconsequential if there can only be one choice of governor for any language construct. However, as previously discussed, there are many potential criteria for headedness. This many-to-one relationship of dependency trees to phrase structure trees supports an argument for the value of dependency analysis independent of a phrase structure framework.

At the same time, it identifies uncertainty associated with dependency structure. A nebulous model of dependency is less appealing as the basis for a well-defined IR model compared to statistical word associations, or the apparently well-defined, discrete categories of phrase structure grammar. Fortunately, in practice, many variations in dependency structure are irrelevant for IR because they focus on relations in which either the governor or the dependent is a function word that is discarded during query and document processing. For example, articles (e.g. *the*), complementizers (e.g. *that*),

auxiliary verbs (e.g. *have, be*), and prepositions (e.g. *in, for*) are all stopwords that are often removed from text. If dependencies involving these words are collapsed, as proposed in the Stanford dependency representation, then many differences between dependency structures are resolved. For example, the major differences between the structures of Tesnière's dependency grammar, and Hudson's non-standard analysis of governor and dependent assignments shown in Figure 3.4 are no longer apparent. This observation lifts a barrier to the exploitation of dependency grammars for IR on the basis that dependencies are not sufficiently well-defined.

### 3.2.5 History of linguistic dependency for IR

Dependency grammars were acknowledged by the IR community soon after the publication of Tesnière's work on formal dependency theory in 1959 (Tesnière, 1959). Initially, dependency structure was perceived to offer two benefits: a means of word association (governors are associated with their dependents), and an alignment with the deep language structure of transformational grammar. This alignment meant that dependencies could be used to improve matching processes for queries and documents. Normalization was possible for spurious differences in surface text representations generated by constituent structure and word order (Smeaton et al., 1995). For example, there are differences in the constituent structure of '*natural language processing*' (noun phrase) and '*processing of natural language*' (verb phrase), but a consistent (collapsed) dependency relationship between '*language*' and '*processing*', and between '*natural*' and '*language*'.

Salton's experiments with SMART in 1964 were among the first to use dependency grammars (Salton, 1964b) with a parser inspired by experiments in Russian and English (Kuno and Oettinger, 1962). This parser seems to draw upon extensive use of dependency trees in Meaning-Text Theory (MTT) developed in the USSR in the early 1960s (Mel'čuk, 1981). The presence, or absence, of dependencies was used to filter phrases identified by statistical co-occurrence, so that only co-occurring pairs of concepts that were also linked by governor-dependent relations were retained. The method was part of the indexing and syntactic phrase matching system, and leveraged dependency structures for word association and phrase normalization.

The disappointing performance of this technique in the SMART experiments, and a misconception about the subsumption of dependency grammars by phrase structure grammars, had a detrimental impact on nascent interest in dependency grammars for

IR. Salton concluded that phrase structure grammars and dependency grammars "can be used interchangeably for present [IR] purposes" (Salton, 1964b). The rapid rise and expansion of Chomsky's transformational grammar further bolstered interest in phrase structure grammars at the expense of interest in dependency grammars. The IR community perceived language to be primarily constituent-based, and although dependencies were sometimes used to normalize differences in text representations (Smeaton et al., 1995), or identify word relations within and between constituents (Lewis and Croft, 1990; Smeaton and van Rijsbergen, 1988; Strzalkowski and Carballo, 1993), they were typically applied in conjunction with phrase structure representations.

Examples of the joint application of phrase structure and dependencies include the COPSY system that was part of the TINA project at Siemens. This used dependency relations to normalize the content of noun phrases, where dependency relations were determined using heuristic rules. The resulting phrases were found to be useful, especially when combined with "more general terms" (Schwartz, 1990). Tree Structured Analytics (TSAs) were another combination of constituents and dependency relations. These idiosyncratic binary tree representations encoded syntactic relations between two sibling nodes at their parent node, and were identified with a constraint grammar using dependency relations and base grammatical categories (Smeaton et al., 1995). TSAs could determine the semantic closeness of phrases in queries and documents. Unfortunately, they were less effective than baseline *tf.idf* weighting (Sheridan and Smeaton, 1992; Smeaton et al., 1995; Smeaton, 1999). Experiments by Strzalkowski and Carballo (1993) were more encouraging. They identified pairwise predicate-argument associations between nouns and low-level grammatical categories, such as noun-noun, noun-adjective, and verb-noun pairs. The resulting phrases were used as word contexts, where word contexts for query words had two applications: they were used in reformulated queries, and compared to the contexts of document words to identify words for query expansion. This was similar to the query expansion method implemented by Grefenstette (1992) using only phrase structure.

Of course, there were exceptions in which dependency relations were used exclusively. Wang et al. (1985) constructed a relational thesaurus for IR using predicate-argument relations, along with non-dependency relations such as synonymy and part-whole relationships. Dependency-based language models for IR (Gao et al., 2004) also do not use constituent information. However, such examples are relatively uncommon.

For the most part, after the SMART experiments, dependency structures were not used independently of phrase structure grammar until the advent of syntactic language

modeling for IR in 2004. Gao et al. (2004) was the first to adopt this approach, successfully employing a statistical link grammar (Sleator and Temperley, 1993) to identify binary relations between words. The link grammar was implemented using linguistic constraints, instead of a full parsing system. This enabled parsing of ungrammatical sentences and avoided a requirement for pre-processing text with part-of-speech tags. A similar model using a dependency grammar was later implemented by Lee et al. (2006), and extended by Cai et al. (2007b) with a multi-dependency approach. The latter used three types of dependency-based language model to describe both queries and documents. All these approaches improved IR effectiveness compared to a standard bigram language modeling approach.

The success of syntactic language modeling for IR has not significantly shifted the trend of using both constituents and dependencies simultaneously in ad hoc IR. This is largely because probabilistic combination of multiple, assumed noisy, sources of linguistic information achieves strong performance. Instead, to maximize the benefit of feature combination, the recent move has been towards sophisticated machine learning frameworks that improve IR performance by using language features from different representational models of text. For example, one successful strategy learns optimal rankings over candidate subqueries with a conditional random field model trained on features that include both phrase structure and dependency representations (Xue et al., 2010). Another approach uses similarly diverse features to model document relevancy as a translation problem between queries and documents (Park et al., 2011).

These approaches are not without their downside. In ad hoc IR, the implementations can be time-consuming, and may result in long, awkward queries or require parsing of an entire document collection. In addition, features that do not capture desired word relationships for IR can bias a machine learning algorithm *away* from an ideal solution, resulting in sub-optimal performance.

Complex processing is more practical in question answering (QA) because syntactic matching techniques are applied to only a subset of pseudo-relevant sentences or documents. A popular approach in QA computes the similarity between dependency parse trees for answer extraction and answer ranking (Wang et al., 2007) (see Section 2.1.2). Dependency parse features have also been applied to sentence ranking using a function based on the separation distance of component terms in a dependency parse (Cai et al., 2007a). The use of such features in QA is generally found to deliver considerable improvements in performance, presumably because it models the semantic content of sentences (Surdeanu et al., 2011).

### 3.2.6 Summary

Dependency grammars have been largely marginalized from mainstream linguistics, at least partly due to a claim that they are weakly equivalent to phrase structure grammars. In fact, phrase structure grammars do not have a one-to-one correspondence with dependency grammars, but it is relatively easy to convert from one language representation to another. The availability of parsers that output both phrase structure and dependency information, and the ability of dependency structure to normalize certain syntactic variations, has led to a focus on techniques that use features of both constituency and dependency.

A point of difference for dependency grammars and phrase structure grammars is their focus on semantics. Yet even dependency grammars fail to detect all semantically related words. This is discussed further in the next Chapter (see Section 4.4.1), but for example, '*nuclear protest*' is arguably an informative search term for the query '*French protests against testing of nuclear warheads*' (adapted from Robust04 #620), yet '*nuclear*' and '*protest*' are not related by either phrase structure or the most probable dependency parse.[11] They are also not adjacent in text. The application of dependency grammars is therefore well-motivated, but does not identify all semantic content.

Dependency relations used without features of phrase structure are a promising avenue for further exploration. They have significant practical advantages, including the fact that they are not constrained by word order. Some also use lexical forms other than words, such as lemmas or morphological units (Mel'čuk, 1988). This means that they are well equipped to represent the semantics of agglutinative languages, that form most words by joining morphemes together (Nivre, 2005). Dependency-based analysis also presents a constrained parsing problem that results in gains in parsing efficiency compared to phrase structure analysis. While there is a trade off between accuracy and efficiency for any parsing algorithm, non-projective dependency parsing can be sufficiently restricted to make effective parsing possible with a complexity of $O(n)$, given a sentence of length $n$ (Nivre, 2003, 2005). This is compared to $O(n^5)$ for the default configuration of off-the-shelf phrase structure parsers (Cer et al., 2010).

In addition, there are useful applications for simple bilexical governor-dependent

[11] '*Nuclear protest*' is a reasonable search term given that it is commonly used to describe protests against governmental nuclear policy. Selection of this term can be based on world knowledge. If '*nuclear warheads*' is parsed as a compound noun, the variant '*protest nuclear*' is accessible. To see this, consider a query on '*protests about the bus station refurbishment*', where '*bus station refurbishment*' is a compound noun. It is possible to identify all three query terms {'*protest bus*', '*protest station*', '*protest refurbishment*'}. The term '*protest nuclear*' is less accessible if '*nuclear*' is parsed as an adjective for '*warheads*', as it is in the most probable dependency parse.

relations. For example, dependency trees can be used to measure distances between words. Moreover, with respect to the information provided by phrase structure grammars, dependency trees can be interpreted to provide almost as much information using roughly half the number of nodes and edges. By Occam's Razor, a simpler theory should be preferred if a more complex theory does not offer any advantage.

Perhaps most importantly, dependency grammars have a natural compatibility with distributional analyses of word co-occurrence that feature in IR techniques such as language modeling. Consider that distributions of word co-occurrence are regularly used for tasks that identify words with meaningful relations (e.g. statistical word association measures). It is precisely the meaningful relations between individual words that is the focus of dependency grammars. By consequence, governor-dependent relations are a suitable unit of co-occurrence for these tasks. In some sense, dependency grammars even generate a need for statistical analysis of word co-occurrence to validate headedness criteria (Hays, 1964).

In contrast, phrase structure grammars have little affinity with analyses of word co-occurrence because they constrain many words to co-occur with complex constituents, not with individual words. For example, a noun (N) might co-occur with a verb phrase (VP) as in '*John*/N – *studied information retrieval*/VP' or a verb (V) might co-occur with a noun phrase (NP) as in '*studied*/V – *information retrieval*/NP'. In fact, the only words available for co-occurrence analysis in phrase structure grammars are those that appear together in simple constituents composed of elementary units, such as the noun-noun combination '*information retrieval*'. As a result, statistical word associations identified using word co-occurrence are not compatible with phrase structure theory.

Admittedly, a lack of consensus on the 'right' assignment of governor and dependent roles, and a complete set of typed dependency relations, is a disadvantage for dependency grammars. However, this is symptomatic of uncertainty about how function words participate in dependency relations. One solution is to omit function words in IR. Alternatively, it may not be necessary for a dependency parse to exhaust syntactic analysis, with the benefit that dependency grammars can be applied to real-world data that is not grammatically correct (see for example, Gao et al. (2004)).

Practical advantages have helped to generate interest in dependency parsing for general language processing tasks, and recognition of these advantages is slowly stimulating further critical investigation of language processing by the search community. In light of their strengths, dependency grammars appear to be an appropriate and useful model of language for IR.

## 3.3 Lexicalism

### 3.3.1 Definition of lexicalism

Lexicalism, also known as *lexical theory*, is broadly concerned with "the frequency and therefore importance of lexical phrases, the varying degrees to which lexical phrases are open to variation in wording, the function of lexical phrases, and the importance of lexical phrases to a model of language that gives lexis and grammar equal priority" (Hunston and Francis, 2000). The word *lexical* is used because it is 'of or relating to words', as opposed to other linguistic units, such as morphemes or phonemes. In addition, it is associated with *lexicology* in which lexical items are studied primarily through the construction and use of dictionaries and thesauri. Lexical items are often, but not always, words e.g. '*data base*' is a lexical item. Lexicalism champions the importance of language patterning as applied in practical lexicology, and assumes that word patterns relate to semantic meaning (function).

Lexicology can be defined in a narrow or a broad sense (Cowie, 2005). The core definition pertains to *lexicography*, the art of compiling, writing and editing dictionaries, and the development of components and structures linking the data in dictionary entries. A slightly broader definition includes *lexical semantics*, the study of the meanings of words as might be considered when writing a thesaurus. It may also include *phraseology*, the study of phrases and IDIOMS.[12] Idioms are phrases in which an exact combination of words is fixed, and the meaning of a whole expression cannot be predicted from the usual meanings of the component words.

A broad definition of lexicology includes *corpus linguistics* and *empirical semantics*. Corpus linguistics is a modern approach to the exploration of lexical items in large corpora, and empirical semantics is the study of semantics using corpus linguistics. These extended areas of lexicology have helped to drive advances in automated language processing. They have made huge strides since the 1980s when it became common for linguists to have access to computers powerful enough to handle large text collections, and dominate many areas of computational language processing today.

Corpus linguistics and empirical semantics are not part of the core definition of lexicology for the simple reason that lexicology has a much longer history. However, lexicography, lexical semantics, phraseology and corpus linguistics are generally treated

[12] See Nunberg et al. (1994) for a complete discussion of idiom.

as branches of lexicology in Britain, where corpus linguistics was primarily developed by the linguists Halliday and Sinclair. Elsewhere, these areas may be considered separately (Cowie, 2005). I will refer to lexicology (and *lexicalism*) in the British sense because large collections of text are necessarily employed in IR, making corpus linguistics of primary importance. Moreover, phraseology is pivotal to the study of word associations.

Lexicalism has many related meanings, but as a theory of language it assumes that meaning in language is formed irregularly and more or less directly grasped without consideration for how the parts are assembled. On a practical level, this relates to the lexicalization of grammar, where lexical items are assigned functions for the analytic formation of greater units. The hypothesis is that grammatical information is specified in, and projected from, the lexicon. The governor-dependent information in a dependency structure, or connections between words in a phrase structure, are encoded as requirements and restrictions on the type and number of arguments and complements for words, instead of grammatical rules that are separate from words.

Lexicalists use unrestricted language data to discover patterns of language use, the extent to which they are used, and the contextual factors that influence such patterns. In contrast with MENTALIST theories, such as Chomsky's account of syntax, lexicalism is objective (Rehman, 2010). However, there is a similarity between lexicalism and phrase structure theory in that both are based on analyses of contrasting lexical environments. These analyses determine the similarity of small text units, such as words, by looking at their co-occurrence frequencies with other words and phrases in text.

To clarify the lexicalist approach, and how this differs from the distributional analysis that grounds phrase structure theory (Structuralism), consider the following characterization. Let $A$ and $B$ be unique text units and $C$ and $D$ be elements of a textual environment. Simplifying greatly, we can classify $A$ as being similar to $B$, or in the same substitution class, if we find occurrences of both $CAD$ and $CBD$ in a text collection to be equally likely (in practice, of course, classification is based on occurrences in many different textual environments using word distributions in text, but we will ignore this for a moment). For example, if $A$ = '*reads*', $B$ = '*studies*', $C$ = '*Jane*' and $D$ = '*biology*', we might determine that $A$ and $B$ belong to the same class because we find attestations of both '*Jane reads biology*' (British English) and '*Jane studies biology*' (American English) to be approximately equivalent. In the simplest case, we might classify $A$ and $B$ into the same substitution class if we find occurrences of both $CA$ and $CB$, or $AD$ and $BD$.

The major difference between Structuralist and lexicalist analyses is the way text units and textual environments are represented. In Structuralism, substitution classes are grammatical categories. If we wish to identify whether *A* and *B* are in the same grammatical category, then *C* and *D* are represented either as grammatical categories, or abstract representations of language such as 'inflected word'. For example, if we find that attestations of both '*N reads N*' and '*N studies N*' (using *N* for noun) are equally likely, then we might determine that *A* and *B* are in the same grammatical category, in this case the category of a verb.

In lexicology, the substitution classes are semantic meanings. If we wish to identify whether *A* and *B* have similar meanings, then *C* and *D* are represented as words, phrases, or semantic concepts. The analysis stays close to the observed data and does not perform prior analysis such as assignment of grammatical categories. If we find that attestations of both '*Jane reads biology*' and '*Jane studies biology*' are equally likely, then we might determine that *A* and *B* have a similar dimension of meaning ('*reads*' is used instead of '*studies*' with respect to college level academic work in British English).

Based on the similarity of these discovery procedures, we can draw a parallel between constituents as the structures that emerge from phrase structure theory, and collocations as the structures that emerge from lexicalism. The notion of collocation is one of the key contributions of lexicalism, and is based on an analysis of contrastive lexical environments for words. A more specific and comprehensive definition of collocation is the subject of the next Section.

### 3.3.2 Definition of collocation

Collocation is identified by many names, including lexical phrases, composites, sentence stems, phrasemes, formulaic language, conventionalised language forms, prefabricated language chunks, phrase patterns and more (Hunston and Francis, 2000). Many definitions have been proposed that focus on syntagmatic word relations and phrase frequency, as well as intuitive ideas about semantics and idiosyncratic constraints on word combinability.

Simplifying greatly, collocations are syntagmatic word associations that can be used to identify PARADIGMATIC meaning. Syntagmatic relations hold between ordered sequences in space or time, such as words in text or speech (e.g. including, but not limited to, ngrams). They are traditionally used to define phrases that might require

their own entry in a lexical resource such as a dictionary. For example, the phrase '*pat down*', meaning an act of searching a person for concealed items, requires its own dictionary entry. Paradigmatic relations hold between members of conceptual sets, such as {*run, runs, running*}, {*teacher, student, lesson*} and {*cat, bat, sat*}. A paradigm is a set of word forms, or word meanings, that share some function and can be substituted for each other in some context. For example, synonymy is a paradigmatic relation in which two or more lexical items have the same semantic sense. However, members of a paradigmatic set do not necessarily share the same meaning. They might have different meanings yet be substitutable in some lexical or phonetic environment, e.g. in British English, '*high school teacher*', '*high school student*', '*high school lesson*'.

Contrastive lexical environments, or collocability, are used to identify the function shared by each member of a paradigmatic set. In simple terms, aspects of meaning are determined by word sequences, and word sequences can be used to identify important word relations including:

- **Synonymy:** when two lexical items have the same meaning. Synonymy is important in IR because it identifies variant expressions of the same concept;
- **Polysemy**: when one word form has two or more related, but separate, meanings. Polysemy is important in IR because related word senses constitute partial representations of a full concept (Krovetz, 1995);
- **Homonymy**: when one word form refers to more than one lexical item with a distinct meaning, such as '*bank*' referring to both an entity that manages money, and a mound of earth that contains a river. Homonymy is important in IR because it separates unrelated concepts.

The most notable early empirical investigations of lexical patterns were made by Harold Palmer for learners of English as a foreign language in Japan (Palmer, 1933). Palmer's findings were highly influential in education (Smith, 1999), but the first discussion of collocation in linguistics is usually attributed to the British linguist John R. Firth (van der Wouden, 1997). Firth defined collocation to be a type of syntagmatic word association that captures a word's ability to combine with other words (Firth, 1935). He examined a specialized set of phrases, rather than lexical compatibility as a whole (van der Wouden, 1997), yet it transpired that his definition is relevant to a wide range of word associations.

A word's ability to combine with nearby words was subsequently a focus for two of Firth's students, Halliday (1966) and Sinclair (1966). Halliday illustrated the impor-

tance of syntagmatic relations with an example of two adjectives: *strong* and *powerful*. These adjectives have a similar semantic meaning, so if we extrapolate from syntactic or semantic criteria, then we expect them to appear in the same lexical environments. For example, we expect to find instances of '*powerful tea*' as well as '*strong tea*', and '*strong car*' as well as '*powerful car*' in any large corpus of text (Halliday, 1966). However, it can be observed that people prefer to say '*drink strong tea*' rather than '*drink powerful tea*', and '*drive a powerful car*' rather than a '*drive a strong car*'.

The fact that large text collections do not reveal certain expected word patterns indicates that language, as proposed by Smadja (1989), "cannot be accounted for on pure syntactic or semantic grounds. These are lexical constraints that need to be introduced in order to filter out such oddities when producing English". We can conclude that there must be some syntagmatic criteria that influence these significant word patterns, and that word sequences have some primary role in language.

The syntagmatic aspect of collocation is important, but unfortunately, word sequences alone are insufficient to identify all and only the significant word associations in text. Lexically related words are not necessarily adjacent or contiguous (Sinclair, 1966). In addition, syntagmatic relations offer no way to distinguish between *collocations* that form significant word pattens, and *co-occurrences* that do not. Guidance is required on how many intervening words are permitted between collocates, and whether collocations are formed by all word pairs that are separated by some distance.

Halliday suggests a solution to this predicament, namely that collocation requires both "linear co-occurrence together with some measure of significant proximity, either a scale or at least a cut-off point...such as significant deviation of the probability of occurrence from the unconditioned probability" (Halliday, 1966). His idea is that a collocation is composed of a central word and any number of words that co-occur with it within an arbitrary distance. The degree to which the words constitute a collocation is the frequency, or probability, with which the central word co-occurs with other words in the combination. This may be calculated relative to the frequency, or probability, that the central word co-occurs with any word not in the combination (Gledhill, 2000).

Following this lead, many modern definitions of collocation incorporate a statistical notion of significance (as do many measures of word association, see e.g. Church and Hanks (1990)). However, there is no agreement on precisely how such definitions should be made. Two common interpretations of statistical collocation define it as a sequence of adjacent words that frequently appear together, or possibly an interrupted sequence of words that appear together more often than expected (Church and Hanks,

1990; Gledhill, 2000). The benefit of such statistical definitions is that they are not limited to a single linguistic constraint so they have the flexibility to systematically identify many varied expressions in text.

The downside of a purely frequentist approach is that it can identify many trivial and superfluous word combinations. It can also inadvertently exclude significant collocations from consideration. As a result, while a statistical approach improves over a simple, syntagmatic definition, it can also fail to distinguish between collocations that are significant word associations and co-occurrences that are not.

The solution to this difficulty requires reference to additional criteria that usually rely on opposing concepts of collocation and free word combination:

- **Collocations:** combinations described by a degree of fixedness. Fully fixed combinations behave like words that must be learned as separate elements of language and interact with other sentence components as single units (Fillmore et al., 1988; Kjellmer, 1984). For example, '*the Statue of Liberty*' is a fixed combination that behaves semantically as a single term;
- **Free combinations:** combinations formed according to grammatical rules that have fully compositional semantics and may be productive, such that their components form the basis of novel combinations. Free combinations come together without idiosyncratic constraints, and include syntactic phrases such as '*read a book*'.

It is not obvious how to assign boundaries between these categories. A third category, such as 'partially fixed', is often created for flexible and transitional collocations. These are combinations that are neither fully fixed, nor completely free.

The lack of a clear distinction between collocations and free combinations led Howarth (1998) to suggest that there is a continuum of collocations, from those that behave more like phrases, to those that behave more like single terms. Collocations thus lie on a scale from *flexible* to *fixed* based on the relationship between their overall meaning and the degree to which component terms are restricted (Howarth, 1996). All collocations are somewhat fixed, and a privileged position is often assigned to idioms as quintessential, non-productive fixed combinations (Wood, 1981).[13]

The somewhat fixed presentation of collocations led Sinclair (1987a, 1991) to suggest that language contains a large number of semi-preconstructed phrases, and these

[13] Idioms are identified as collocations in some accounts (Gledhill, 2000; Kjellmer, 1984), but not in others (Cruse, 1986; Mel'čuk, 1998; Wood, 1981).

phrases are selected and used as single units during communication. They help to ensure that language output is natural, in contrast with syntactic structures that ensure output is grammatical.[14] Specifically, his principle of idiom asserts that meaning is phrase-based, rather than being derived from the meanings of parts and the way they are put together. Sinclair (1991) states:

> "The principle of idiom is that a language user has available to him or her a large number of semi pre-constructed phrases that constitute single choices, even though they might appear to be analysable into segments."

Further criteria for a distinction between collocations and fixed combinations are intuitive and fuzzy. Benson (1989) argues that collocations are not merely recurrent word combinations, they must also be arbitrary, such that the constraints on word combinability do not reflect grammatical or semantic considerations. For example, it is somewhat arbitrary that people use the phrase '*strong tea*' but not '*powerful tea*'. Similarly, the semantic idiosyncrasy of idioms led van der Wouden (1997) to suggest that collocations must be idiosyncratic.

Semantic coherence was a defining characteristic for Mel'čuk (1998). Mel'čuk took the view that many collocations behave as semantic functions operating between two or more words, one of which retains its standard meaning. Language is organized into a typology of semantic functions that are lexical in nature, such as intensity, quantity, operation and function. For example, '*reckless abandon*' and '*theatre of war*' are, respectively, collocations of intensity and location.

The diversity of these definitions indicates that the phenomenon of collocation is untidy and diffuse (Mel'čuk, 1998). Syntagmatic relations, statistical frequency, arbitrariness, idiosyncrasy, and semantic coherence all may be pertinent attributes of collocation, with the consequence that any single definition is unlikely to be complete. Even syntactic criteria cannot be entirely ruled out. Syntax does not predict the observed frequency of certain fixed phrases and lexical preferences. It also cannot explain collocation because the words in both collocations and free combinations are syntactically related. However, this does not preclude the participation of collocated words in syntactic relationships. In fact, Mel'čuk (1998) introduces the possibility that there is a deep syntactic, or dependency, relationship between collocated words.

[14] Chomsky used the sentence "Colorless green ideas sleep furiously" (Chomsky, 1957) to demonstrate the substitutability of words in a grammatical category, even if the resulting sentence does not make semantic sense. It has been contended that without the principle of idiom, "colorless green ideas would indeed sleep furiously" (Gledhill, 2000).

In his account, Mel'čuk claims that collocations can be represented either by lexical functions, or by government patterns such as those applied in Meaning-Text Theory (MTT). Lexical functions reflect deep syntactic roles, and government patterns are formed by dependency relations. Admittedly, Mel'čuk's definition of collocation excludes certain set phrases that he calls '*pragmatic phrasemes*' or '*pragmatemes*'. These compositional phrases have a specific and uniquely identified meaning and include technical terminology, as well as terms like '*caesar salad*', that are identified as collocations by standard definitions. Nevertheless, pragmatemes are also likely to participate in dependency relations, so they do not contradict the possibility that collocations have a syntactic interpretation.

The insight that can be drawn from analysis of collocation is that there may be no single approach to language analysis that uniquely identifies desirable word associations for IR. Further, any approach to language analysis, whether it focuses on syntax, semantics or statistics, might identify desired word associations when used in combination with other language features. For example, collocations cannot be accounted for by syntax, but a syntactic method in combination with statistical features may be sufficient to approximate a desired set of terms.

### 3.3.3 Suitability for IR

Lexicalism is well-suited to IR for four reasons, two of which are shared with dependency grammars. First, like dependency grammars, lexicalism focuses on language semantics. However, lexicalism emphasizes global, as well as local, semantic relations (at the scale of collections, documents and sentences), while dependency grammars largely focus on local relations (within a single sentence). In this respect, lexicalism is especially well suited to language analysis for IR.

Second, both lexicalism and dependency theory share a FUNCTIONALIST view that actual attestations of language are fundamental to understanding its organizational form.[15] This makes them appropriate for the analysis of real-world data. In fact, analysis of real-world data is pivotal in lexicalism since it does not use annotations employed in other linguistic theories, such as parts-of-speech tags and assignments of headedness. Sinclair, one of Firth's students, claimed that the use of annotations, or tags, leads researchers to discover patterns in annotations that hide patterns in text data and reinforce a priori beliefs about their suitability (Sinclair, 1987a). This results in a

[15]This is one aspect of functionalism. The view is derived from an assertion that language cannot be separated from its purpose as a means of communication.

"vicious methodological circle" (Stubbs, 2009) in which relationships are discovered between tags and outputs of interest, leading to more annotations, and so on. Sinclair demonstrated that multi-word units of meaning can be discovered through empirical observation of recurrent syntagmatic word patterns in large text collections without the use of tags (Krishnamurthy, 2005; Sinclair, 1987b).

Lexicalism also has two possible advantages over dependency grammars for IR. First, it views semantic relationships to be essentially phrase-based.[16] This makes it a suitable foundation for word dependence models in IR. In addition, it recognizes that syntagmatic relations between words may be at least as important for the interpretation of meaning as syntactic relations and the semantics of individual words. This means it can account for the success of what are often thought of as 'non-linguistic' methods.

The suitability of lexicalist theory for IR is demonstrated by the success of techniques developed without knowledge of lexical theory, but that nonetheless reflect lexicalist insights. For example, stemming is well-established as a means to normalize English vocabulary and often improves IR effectiveness. This is predicted by lexical theorists, who point out that lexical relations do not respect grammatical categories. One reason stemming is effective may be that it removes irrelevant distinctions between grammatical categories. More pointedly, corpus analysis also predicts that stemming may be detrimental for IR in certain circumstances. Different forms of a canonical word (e.g. {*eye, eyes, eyed*} for the canonical '*eye*') can have distinctly different frequencies and collocates. They also may have different meanings. Collapsing such forms to the same stem makes it more difficult to discriminate between relevant documents. Sinclair provides the example of the singular '*eye*' versus the plural '*eyes*' (Stubbs, 2009). '*Eyes*' collocates with colors like '*blue*' and '*brown*', and has a literal meaning. In contrast, the singular '*eye*' appears in expressions of monitoring and evaluation, such as {*keep an eye on, turn a blind eye, in the public eye, in her mind's eye, more than meets the eye*}. This suggests that it is inappropriate to conflate certain word inflections by stemming.

Lexical analysis is rich with examples in which a similar divisions in meaning can be inferred from variants in word form, although to my knowledge such empirically-driven stemming has not been evaluated in IR. The stemmer developed by Krovetz (1995) handles many special cases in English, but does not address this point. It identifies irregular inflections that should be stemmed to the same form, for example '*matri-*

[16] It is possible to represent phrase-based relationships in dependency grammar, but this is not the norm. Most dependency grammars represent relations between words.

*ces*' and '*matrix*'. It also identifies words that should not be stemmed to the same form because they have different meanings, even though they appear to be inflections of the same word, for example '*suited*' and '*suites*' do not share '*suit*' as their stem. However, it does not identify regular inflections that should not be stemmed to the same form if we wish to preserve contextual meaning, such as '*eye*' and '*eyes*'.

More recent developments in IR also reflect lexicalist ideas. In lexicalism, identification of highly frequent phrases across many different texts led to questions about their nature and purpose. It was suggested that they "serve text-management functions, such as signalling narrative structure, topicalization, point of view, and the like. They do not denote anything in the world, but signal the attitude of the speaker and a textual contrast" (Stubbs, 2009). In IR, such highly frequent phrases are known as *stop structure*. Their removal is shown to improve the effectiveness and efficiency of verbose queries (Huston and Croft, 2010; Spärck Jones and Tait, 1984) precisely because they do not denote anything of importance.

In summary, lexicalism predicts major experimental findings in IR even if its predictions do not always translate perfectly to a search context. For example, patterns of word co-occurrence studied by lexicalists are domain independent (Krishnamurthy, 2005), while queries can be domain specific. In addition, lexicalists are often interested in collocations that include stopwords, while stopwords are often discarded for IR. Nevertheless, lexicalism is well suited to language processing for search tasks.

### 3.3.4 History of lexicalism for IR

There are two peculiarities of lexicalism in IR. The first is that research applying lexicalist principles goes back to the foundations of classic IR in the 1960s even though there was, and continues to be, very little awareness of lexical theory in the search community. The second is low awareness of lexicalism itself. It is sufficiently salient that in place of a history of lexicalism in IR, I consider the lack of history and the context in which it eventuated.

Examples of applied lexicalist principles in IR are numerous and include word normalization, phrase-based indexing, proximity matching, dependency models,[17] co-

[17]Here, dependency models are discussed in the sense of IR models (see Section 2.2). Lexicalism is not related to linguistic dependency theory except in that dependency grammars define relationships between words (they are lexicalized). Lexicalism assumes that meaning in language is more or less directly grasped without consideration for how the parts are assembled, and studies statistical patterns of language use. In comparison, dependency grammars aim to describe all grammatical sentences in a language, and are concerned with the way in which words are assembled to form sentences.

occurrence-based term selection, and the use of lexical context to define meaning in semantic space models for IR. Many IR techniques are presented as statistical or mathematical, but assimilate lexical theory in the use of statistical collocation and word distributions (Sinclair, 1987b). Alternatively, techniques are presented as purely heuristic, but embody ideas about phrase-based meaning as described in the principle of idiom (Sinclair, 1991). In general, any technique that leverages patterns, or frequencies, of term co-occurrence, syntagmatic sequences or word context (which is typically described by adjacent and co-occurrent words) also reflects lexicalist ideas.

In the 1960s, context was used to overcome word mismatch between queries and documents. A typical approach converted from observed words to controlled terms by means of a manually, or automatically, compiled thesaurus or synonym dictionary (Salton, 1963). Related words were clustered, or classified, into synonym sets using word co-occurrence, and assigned a normalized representation. For example, word associations for IR were computed with an application of the chi-squared formula with a correction for small sample sizes (Stiles, 1961), word-word binary matrices (Meetham, 1963) and a co-occurrence-based similarity matrix (Needham, 1965).

Collocations were also applied in phrase-based indexing from the 1960s through the 1980s. Statistical collocations were found to make more effective indexing terms than syntactic phrases (Fagan, 1987). However, phrase-based indexing has largely fallen out of use because there is no conclusive evidence that it improves retrieval effectiveness compared to word-based indexing (Croft et al., 1991). Proximity measures that leverage a simple linear sequence of words together with a measure of significant proximity are often used instead. These also share characteristics with collocations. Significant proximity is defined by a window of pre-specified length and collocation is commonly defined as a combination of words that co-occur with some significance.

Some dependence models also use syntagmatic word associations, such as ngrams, that are central to lexical theory. Ngrams are a feature of language models (Song and Croft, 1999) and the sequential dependence (SD) model (Metzler and Croft, 2005), among others. In fact, the SD model uses a combination of ngrams and proximity operators, both of which reflect lexicalist ideas. In addition, lexical patterns feature in semantic space models for retrieval and query expansion (Bai et al., 2005; Bruza and Song, 2002). For example, the Hyperspace Analogue to Language (HAL) (Lund and Burgess, 1996) measures word similarity using a matrix in which each row is a vector of co-occurrence counts for a word $x$. Counts are determined by moving a sliding window over a large text corpus.

Lexicalism has a natural affinity with IR but the foundational work of lexicalists is generally ignored by the search community.[18] This is quite surprising since IR research in Britain evolved in the 1960s at the same time as early experiments on linguistic collocation by Firth's students. In addition, Halliday was fortuitously co-located at the CLRU with prominent IR researchers, including Karen Spärck Jones, a highly influential leader in the area of language processing and IR. Given a shared interest in data-driven language semantics, it seems natural that there might have been cross-pollination of ideas. Halliday was certainly aware of, and cites, concurrent work in IR (Halliday, 1966). However, this was not reciprocated. Needham (who was working at the CLRU on identification of synonyms using word co-occurrence) refers to Firth when he states that, "The properties which are used [to determine semantic similarity] are very commonly of a distributional sort, which we take to mean that they are concerned with such facts as where a word is found in a text, *what company it keeps*, and so on" (Needham, 1967) (emphasis my own, referring to Firth's slogan, "You shall know a word by the company it keeps" (Firth, 1957)), but the reference is uncited, perhaps because Firth's impenetrable prose would be unlikely to clarify discussion (Spärck Jones, 1965).[19]

Part of the problem was that lexical theory was not fully developed for many years. As a result, shared interest in practical data semantics was not readily apparent when collaboration between lexicalists and researchers in IR might have been readily pursued. Sinclair, not Halliday, developed the idea that multi-word units of meaning can be discovered by observation of recurrent word patterns in large text collections (Stubbs, 2009). However, computers in Britain were not powerful enough to manage text collections of the size required for full exploration of his ideas. As a result, although his early experiments took place between 1963 and 1969, his resulting report was not available until 1970 when directions for IR research were already well-established. Furthermore, the report was not widely available. It was only circulated amongst a small group of academics in linguistics. Although "enormously influential" amongst this group (Sinclair, 2004), it was not formally published until 2004.[20] Moreover, extended work, pursued as part of the COBUILD project in lexicography (Sinclair, 1987a; Stubbs, 2009), was postponed until the 1980s when computer-aided analysis of large text collections became possible (Sinclair, 1991).

[18] Rare exceptions include a brief reference to Halliday and Sinclair by Krovetz (1995), and discussion of collocation by Church (2008).

[19] Firth wrote in a manner described by Spärck Jones (1965) at the time as a "philosophical bog".

[20] Known as the OSTI report (UK Government Office for Scientific and Technical Information).

It can be expected that any interaction between lexicalists and the search community outside of the CLRU would be even more limited than it was within the group. The division between lexicalism and IR might therefore best be explained by the attitude of Spärck Jones, who dismissed Firth's phrase-based collocations as irrelevant (at the time) to her interest in a bag-of-words approach to semantics (Stubbs, 2009). She wrote:

> "...it may be difficult to come to any conclusion, but we have to make a decision...[and] in such cases we are dealing with physical rather than linguistic facts" (Spärck Jones, 1965).

This indifference was protracted by the wave of enthusiasm for Chomsky's novel ideas about language. During an interview in 2001, Spärck Jones remembered of the CLRU in the 1960s:

> "I continued to be interested in language, but there was simply no funding of any sort for work on what we then would have called 'computational linguistics' - natural language processing. I tried to maintain an interest in it, and I went to a few meetings... [but] basically, in the late 60s it got very difficult, because a lot of projects were finished...[and] that was the period in language when everybody was dead-keen on syntax; it was all the Chomsky period...[If] you didn't think that Chomsky was the greatest thing since sliced bread, nobody would really take any notice of you...It was like a religion" (Spärck Jones and Abbate, 2001).

On the other side the Atlantic, the isolation of the British linguistics community meant that emerging lexical theory had no impact. Firth is described as "[sharing] some of Britain's insularity, lacking ambition to persuade those elsewhere of his ideas... He was certainly not understood in the U.S." (Honeybone, 2005). Conversely, Chomsky's aggressive personal style, mirrored by many of his young and enthusiastic followers, was to vigorously dismiss and deride opposing theories of language. This had the effect of ensuring that transformational grammar appeared to be the only viable method of language analysis. Nevin (2010) comments that:

> "[Chomsky] has been called an intellectual bully, and has been accused of all sorts of intellectual malfeasance... And if he cannot win, the argument or the terms proposed are dismissed as unimportant, or trivial, or uninteresting. Countless anecdotes have been told, and many have been published."

Chomsky himself, in a personal communication to Randy A. Harris, wrote:

> "I was told that my work would arouse much less antagonism if I didn't always couple my presentation of transformational grammar with a sweeping attack on empiricists and behaviorists and on other linguists. A lot of kind older people who were well disposed towards me told me I should stick to my own work and leave other people alone. But that struck me as an anti-intellectual counsel" (Kilpert, 2003).

Halliday particularly rejected this adversarial approach and refused to engage with it. He maintained that:

> "The better course is surely to make a straightforward statement of one's case... and then simply to let other scholars consider the alternatives offered, and make up their own minds as to which is more helpful" (Halliday and Fawcett, 1987) cited by (Kilpert, 2003).

This oppositional academic climate was compounded by limited interest in IR research from Europe. Salton observed that, "Few Americans read the foreign technical journals, and the assumption is widespread that European work is either inferior in quality or, in any case, lagging behind equivalent American work by many years" (Salton, 1964a). Salton himself was of the opinion that extensive work on word and document associations in America "is not matched by a comparable effort in Europe... in Europe there is a deep distrust of the statistical methodology for the analysis of information" (Salton, 1964a). It is likely that a lack of computing resources in Europe comparable to those in the United States contributed to these observations (Salton, 1964a).

This context for the development of IR research resulted in low awareness of lexicalism and a consequent gap in the continuity of knowledge from language theory to an applied system for language understanding in IR. The gap may be of little practical concern because the field of IR claims a rich, alternative legacy in mathematics. Nevertheless, a number of leading researchers have been inspired to fill it with references to the work of Ludwig Wittgenstein and Zellig Harris. The work of these figures has clear merits, but their theories are arguably less well suited to IR than lexicalism.

Wittgenstein (1953) was a philosopher (1889-1951) of logic, mathematics, mind and language in Britain who is referenced by Salton and cited as the inspiration for query logs by Amit Singhal at Google (Salton and Buckley, 1991; Salton et al., 1993; Levy, 2011). He took the position that language is made explicit through linguistic evidence, advancing the core idea that, "the meaning of a word is its use in the language", and that language is "part of an activity, or of a form of life" (Wittgenstein, 1953).

That is, language is not defined by the semantic references it makes to things in the world, but by the way it is used for communication (Rehman, 2010). As a result, there can be no exhaustive grammar (such as any context-free, phrase structure grammar), and grammatical rules are inextricably intertwined with the social context of language (Frohman, 1990; Halpin, 2009).

Wittgenstein promotes observed word relationships over idealized grammatical structure and his functionalist framework is well-suited to empirical language processing. There is widespread awareness of his ideas, and perhaps most pertinently, Karen Spärck Jones, who contributed greatly to early discussion of language processing for IR, worked for most of her life alongside one of his most prominent disciples.[21] However, it was Firth's students who would go on to make Wittgenstein's view of language specific, and empirically investigate their ideas using large data collections and statistical text processing.[22] This practical contribution is more applicable to IR than Wittgenstein's philosophy, but is rarely recognized in the search community.

Zellig Harris was Chomsky's PhD supervisor and is often credited with proposing a *distributional hypothesis* that is widely applied in IR (Cai and van Rijsbergen, 2009). Harris was an American linguist in the Structuralist tradition who worked to formulate linguistics as the mathematical analysis of language (Strzalkowski and Vauthey, 1992). Unlike Chomsky, he had a strong interest in empirical data and the formulation of "a mathematical system [describing] all the properties and relations necessary and sufficient for the whole of natural language" (Harris, 1968).[23] His statement that, "The meaning of entities, and the meaning of grammatical relations among them, is related to the restriction of combinations of these entities relative to other entities" (Harris, 1968) is often interpreted as a distributional hypothesis. However, Harris never formally defined a distributional hypothesis and taken out of context his intent is often misunderstood. Specifically, his ideas are used inappropriately to motivate techniques

[21] Margaret Masterman, who directed the Cambridge Language Research Unit (CLRU) for many years, was one of only six students of Wittgenstein in 1933-1934 at the time he was developing his theory of language. Her notes on Wittgenstein's lectures contribute to a key published reference on his work (Wittgenstein, 1958). The CLRU was where Karen Spärck Jones completed her PhD dissertation and worked for much of her life.

[22] Firth developed his ideas about collocation around the same time that Wittgenstein advanced his theory of language. Although they were contemporaries, it seems that Firth and Wittgenstein developed their ideas about language independently. Firth was aware of Wittgenstein's work, and its similarities with the philosophy of his mentor and close collaborator, Malinowski (Gellner, 1998), but had little contact with Wittgenstein. However, Wittgenstein's work may have inspired Firth's students via Masterman.

[23] In contrast, Chomsky defined a general mathematical system representing only a grammatical subset of all language data.

that focus on the distributions of *words*, even though Harris was actually focused on a distribution of *contrasts* (Nevin, 2010).

For Harris, the fundamental data in linguistics are contrasts based on "judgments of what is different and what is repetition" (Nevin, 2010). He was particularly interested in morphemes and phonemes (the smallest possible units of sound and semantic meaning), and noted that while auditory experiences are never repeated exactly, categorical perceptions of sounds are repeated, and can be identified by their contrast with neighboring sounds through the application of substitution tests. The differentiating aspect of Harris' work is that he exposed what lay behind the principles of distributional analysis in Structuralist linguistics (Goldsmith, 2005). He was less interested in grammatical categories themselves, and the distributions of categories (such as words and phrases), than the contrasts that helped to identify them (Nevin, 2010). For this reason, it is properly the lexicalists who investigated semantics using distributions of co-occurring words, not Harris.

The fact that both Wittgenstein and Harris are more widely recognised in the IR community than leading lexicalists like Sinclair and Halliday is due to several factors: a differential speed of research development in IR and lexicalism, limited interaction of the corresponding academic communities, and a lack of funding for computational linguistics. In addition, Chomsky effectively repressed the dissemination of lexicalist ideas by ensuring that they were intellectually unfashionable. In the absence of opponents willing to engage in equally vigorous counter-attack, his arguments dominated linguistics and shadowed other research fields as well. The result was a lack of recognition for lexicalism in IR that continues to the present day.

### 3.3.5 Summary

Lexicalists claim that units of meaning are discovered by analysis of real-world data unencumbered with preconceived notions about grammatical relationships. This emphasizes the nature of units for semantic meaning thought to be predominantly phrase-based, syntagmatic and statistical in nature, rather than units of syntax. As such, lexicalism provides a linguistic foundation for many methods of language interpretation that have been independently discovered and empirically validated in IR. It also provides a rationale for what might appear to be unreasonable effectiveness of 'heuristic' methods in IR if we assume that linguistic analysis is more accurate than word proximity for the identification of semantically representative word associations in text.

Lexicalist principles manifest in phrase-based indexing, term selection, proximity measures and IR modeling. Moreover, lexicalist structures (collocations) account for all language data and are efficient to manipulate. Despite this affinity, lexicalism is not directly acknowledged in IR literature. It seems that an unfavorable academic climate during their development, concomitant funding issues, and slow development and dissemination of lexicalist ideas outside of Britain contributed to this state of affairs. It is also possible that because grammatical category labels used in phrase structure grammar had some positive impact on IR effectiveness, it was assumed that grammatical categories are an appropriate direction for further research and alternatives were passed over.

Lexicalism is an appropriate basis for term selection techniques, but nevertheless might not be sufficient to identify informative terms on its own. The lexicalists themselves have argued that:

> "...syntactic structures and lexical items (or strings of lexical items) are co-selected, and that it is impossible to look at one independently of the other... they are ultimately inseparable, and it becomes merely a methodological convenience to regard them as different perspectives from which to view language use" (Francis, 1993).

A combination of features from different theories of language may produce better term selection, and accordingly, more effective queries, if additional noise in the linguistic evidence does not outweigh the benefit of a greater variety of features.

## 3.4 Conclusion

Language processing for IR has been strongly influenced by trends in modern linguistics. Following Chomsky, phrase structure grammars have been used extensively even though techniques that incorporate grammatical categories result in variable retrieval performance. An implicit assumption is that appropriate application of syntactic processing will overcome any detrimental effect of a mismatch between the premises of phrase structure theory and requirements for IR.

Dependency grammars are also widely applied, but techniques have been largely confined to joint application of dependency and phrase structure representations. Dependency relations are typically used as features in machine learning, or to normalize spurious differences in surface phrase structure trees. Syntactic language modeling approaches to IR (Cai et al., 2007b; Gao et al., 2004; Lee et al., 2006; Maisonnasse

et al., 2007) are the major exception in this regard, revealing the independent value of dependency theory for IR.

Finally, lexicalist ideas are assimilated in many IR techniques although this is not widely acknowledged. Techniques that reflect lexical theory by inclusion of syntagmatic sequences and statistical word associations are shown to be highly effective. It appears that lexicalism provides a strong foundation for term selection in IR.

A representative summary of the regularity with which representations of phrase structure theory, dependency theory and lexicalism have been used in term selection is shown in Table 3.2. Selection techniques and IR models that incorporate terms (for example, language models for IR) are categorized according to whether the representations used include word order (e.g. term proximity, ngrams), labels and structure derived from phrase structure or dependency theory (e.g. grammatical categories, governor-dependent relations) and word distributions (e.g. co-occurrence, word frequencies). Word order and word distributions both reflect a lexicalist approach but are listed separately to distinguish heuristic factors. The contributions of term selection techniques, as opposed to the IR models in which they are applied, are the focus for analysis. An approach using an IR model that incorporates features of word distribution is only marked as such if it reports the IR model for the first time.

The review indicates uncertainty about how linguistic theory might facilitate IR. Development of statistical models was the focus from the 1960s to the 1980s, with limited experimentation focused on leveraging the attributes of grammatical categories. Grammatical categories were determined to be noisy and not as reliable as statistical word associations. The mid-1980s to early 2000s saw greater emphasis on methods that combine different linguistic principles. Head-dependent relations were often used to restrict word associations determined by phrase structure, but positive results fell short of expected improvements in IR effectiveness. Alternatively, statistical and syntactic features were combined. Around the early 2000s, less than optimistic papers on the role of language processing for IR reflected low enthusiasm for the application of linguistic principles in IR. Interest revived in the mid 2000s, spurred by the success of syntactic language models for IR. Finally, from the late 2000s onwards, syntactic features, word co-occurrence and word frequencies used in discriminative IR techniques for term selection and weighting achieved significant improvements in IR effectiveness. However, this trend appears to be reaching a zenith, with minimal difference in the ability of learning algorithms to distinguish between informative and uninformative word combinations using a similar set of features.

| | Publication | Ngram / Proximity | Gramm. Category | Head-Dependent Relation | Word Distrib. |
|---|---|---|---|---|---|
| 1958-1979 | Baxendale (1958) | | | x | |
| | Salton (1964b) | | | x | x |
| | van Rijsbergen (1979a) | | | | x |
| 1980-1989 | Yu et al. (1983) | | | | x |
| | Dillon and Gray (1983) | | x | | |
| | Spärck Jones and Tait (1984) | | x | x | |
| | Wong et al. (1985) | | | | x |
| | Fagan (1987) | x | x | | x |
| | Smeaton and van Rijsbergen (1988) | | x | x | |
| | Metzler and Haas (1989) | | x | x | |
| 1990-1999 | Schwartz (1990) | | x | x | |
| | Lewis and Croft (1990) | | x | x | x |
| | Lewis (1992) | | x | x | x |
| | Grefenstette (1992) | | x | | x |
| | Strzalkowski and Carballo (1993) | | x | x | x |
| | Krovetz (1995) | x | | | |
| | Losee Jr. (1994) | | | | x |
| | Smeaton et al. (1995) | | x | x | |
| | Song and Croft (1999) | x | | | |
| 2000-2009 | Nallapati and Allan (2002) | | | | x |
| | Srikanth and Srihari (2002) | x | x | | |
| | Zukerman and Raskutti (2002) | | x | | x |
| | Bruza and Song (2002) | x | | | x |
| | Risvik et al. (2003) | | | | x |
| | Gao et al. (2004) | | | x | |
| | Gao et al. (2005) | | x | x | |
| | Metzler and Croft (2005) | x | | | |
| | Lee et al. (2006) | | | x | |
| | Cai et al. (2007c) | x | | x | |
| | Kumaran and Allan (2007) | | x | | x |
| | Maisonnasse et al. (2007) | | | x | |
| | Bendersky and Croft (2008) | | x | | x |
| | Lioma and Ounis (2008) | | x | | |
| | Na et al. (2008) | x | | | x |
| | Song et al. (2008a) | x | | | x |
| | Tan and Peng (2008) | x | | | x |
| | Balasubramanian and Allan (2009) | | x | x | |
| | Kumaran and Carvalho (2009) | | | | x |
| | Lease et al. (2009) | | x | | x |
| 2010-2013 | Balasubramanian et al. (2010) | | | | x |
| | Huston and Croft (2010) | | x | | x |
| | Xue et al. (2010) | x | x | x | x |
| | Hagen et al. (2011) | x | | | x |
| | Bendersky et al. (2011) | | | | x |
| | Park et al. (2011) | | x | x | x |
| | Bendersky and Croft (2012) | x | | | x |

Table 3.2: Features of language used for term selection methods in IR.

The question now is how research in IR should move forward with respect to interpretation of linguistic principles and application of language processing to improve term seelction. The popular working assumption that "The best improvements in retrieval performance are not to be found by just discovering the appropriate aspects of syntactic description to utilise, but rather in utilising as many aspects of syntax as possible" (Sheridan and Smeaton, 1992) seems too coarse.

It is intuitive that a combination of different linguistic theories will aid the selection of informative terms for IR. However, an emphasis that detracts from appropriate aspects of syntactic description is unaccountable and leaves little room for a motivated understanding of how term selection techniques can be improved. One aspect that makes linguistic theories appropriate for term selection may be their relationship to semantic interpretation. This relationship is explored further in the next Chapter.

# 4

# Semantic Characterization of Terms

The most common definition of relevance in IR describes a binary relation between queries and documents. A simplifying assumption is that a query presented to a search engine represents the semantics of a request, and is a good expression of the user's underlying information need. This assumption may be true for keyword queries submitted to open domain search engines. For keyword queries, users actively select highly representative query terms. Yet verbose requests often require automated reduction and term selection, typically implemented using word association features to predict effective query terms. The motivating idea for these methods is that word associations identified by phrase structure theory, dependency theory or lexicalism (Chapter 3) map onto the semantic interpretation of requests by users.

The difficulty is that semantics is not necessarily accessible using surface syntax or word order. Words, and word associations, are interpreted by humans, and interpretation is influenced by memory, imagination, emotion, world knowledge, social and physical factors (Lakoff, 1987).[1] By consequence, semantics is not always explicit in text and an information need is not always explicit in a query. This results in a gap between an information need and a query that is not addressed by linguistic processing.

This Chapter explores the assumption that there is a relationship between identifiable word associations and the semantics of a request. Clearly, the relationship between word associations and semantics in general (not specific to IR) is the province of linguistics and computational linguistics, with several well-argued camps (Koenig, 2005).

[1] According to Lakoff (1987), there is often an assumption that language is made up of uninterpreted symbols (such as words), in the sense that we can only have knowledge about the world if the symbols we use accurately reflect the external world. The symbols do not require interpretation. Consequently, we come to expect that machines that do not have interpretation functions related to factors such as imagination and emotion should be able to understand language, based solely on the symbols provided. However, such interpretation may be prohibitively difficult.

However, discussion in this Chapter focuses on the context of IR. I present a theoretical discussion of the alignment between syntax, or word order, and semantics. I also investigate the ability of word association methods to identify user-nominated terms. This assumes that humans are expert at semantic interpretation, and user nominated terms are a gold standard for the semantics of requests. Word association methods are described by the nature of the structure used to identify terms, where structures are classified according to things to be counted (text units) and ways of counting them (statistical measures). This results in a classification of word association methods (and terms) as predominantly syntagmatic, syntactic or statistical:

- **Syntagmatic:** Syntagmatic methods use sequential word relations that play an important role in lexicalism and phrase structure theory. Ngrams are a common example of syntagmatic terms in which words are adjacent.
- **Syntactic:** Syntactic methods capture grammatical information within sentences. Phrase structure and dependency theories underlie the syntactic methods presented in this Chapter.
- **Statistical:** Statistical methods use word co-occurrence patterns. Co-occurrence is usually defined with respect to syntagmatic units of text identified in a sentence, document, collection or window of $n$ words. However, co-occurrence can also be defined syntactically.

In the first part of this Chapter, I define semantic representation and describe the features of syntagmatic, syntactic and statistical classes of terms. I also observe the characteristic properties of each class with respect to semantic representation and consequent limitations on term effectiveness. Note that semantic relations determined using external resources, such as WordNet, are not considered. There was a trend towards the application of external resources in IR, particularly during the early 1990s (Krovetz and Croft, 1992; Smeaton, 1999). However, these resources can be domain specific and are not always available. Instead, the focus is on semantics that can be detected from a query and document collection without knowledge engineering.

The second part of this Chapter evaluates the degree to which word associations represent the semantics of requests. Specific methods of word association are described for each class of terms, and the classification accuracy of methods are measured individually and in combination for user nominated targets. Evaluation suggests that simple application of linguistic theories for the identification of word associations does

not identify semantically representative terms in requests. Semantics has only a weak association with the limited number of word association methods explored. Linguistics and computational linguistics are well placed to comment further on this result.

## 4.1 Definition of semantic representation

A term represents the semantics of a request if it composes words that refer to semantically related elements. For the purpose of IR, a semantic relation fits one of two archetypes that provide valuable context for individual words and thereby improve IR effectiveness. The first type occurs when one word can only be evaluated in the context of another word. Such dependencies are non-compositional, and best treated as indivisible units during retrieval.[2] For example, this can occur with idioms (see Section 3.3) and compounds created by a combination of smaller words (e.g. *'database', 'database'* and *'data base'*). The second type of association occurs when it is desirable, but not necessary, to evaluate two or more words together. These words are assumed to have an implicit semantic relation holding between them such as *x uses y*, *x from y* or *x is y* (Levi, 1978). IR performance benefits from some representation of their dependence in a search system.

At a practical level, there are many types of semantic relations that are potentially relevant. A large number of these map to operations between adjectives and nouns. For example, Levi (1978) argues that complex nominals often exhibit an implicit or explicit connection based on one of nine semantic predicates: {*cause, have, make, use, be, in, for, from, about*}. These connections are fairly clear and are the basis of examples used in this Chapter. Other semantic associations include attribute relations such as those described in a knowledge graph ('*x is a y*') and relations identified by semantic role labeling, e.g. agent-patient or predicate-patient where the predicate may be nominalized. Evert (2005) indicates the diversity of word associations identified using word co-occurrence in the British National Corpus (BNC). He describes many different units with semantic significance for the noun '*bucket*', such as:[3]

- **Proper names:** e.g. *Rhino Bucket*, a hard rock band;
- **Compound nouns:** e.g. '*bucket seat, coal bucket*;

[2]The manner of implementation is variable and therefore intentionally unspecified. For example, words might appear in an ordered or unordered window, or have a specific type of dependency relation, and so on.

[3]The BNC is a 100 million word collection containing samples of written and spoken language designed to represent a wide cross-section of British English from the later part of the 20th century.

- **Lexical collocations:** where *bucket* has lost its original meaning e.g. *weep buckets* (cry a lot);
- **Institutionalized phrases:** e.g. *bucket and spade*
- **Idiom:** e.g. *kick the bucket* (die);
- **Semantic restrictions:** e.g. *carry, tip* are things that can be done to a bucket, and *full, leaky* are possible properties or states of a bucket;
- **Semantic similarities:** e.g. *container, shovel, mop*;
- **Conceptual knowledge:** facts of life that do not have a linguistic interpretation but describe frequent objects and events in the world e.g. *bucket of water*;

Any of these text units might be desirable in IR provided they do not decrease search effectiveness. In IR, there is often a (possibly incorrect) assumption that a text unit which represents the semantics of a request also discriminates between relevant and non-relevant documents. More specifically, a multi-word term identifies documents that are semantically similar to a request because the word association used to identify the term is present in relevant documents with the same relationship, or would be present if vocabulary mismatch was not a problem. In other words, it should not be possible that the words have the same semantic relationship but an inaccessible syntax. Finally, if only one type of language structure is used to identify terms, it is assumed that this structure identifies all relevant semantic relations and gives a complete representation of request semantics. These assumptions will be explored for syntagmatic, syntactic and statistical word associations in the following Sections.

## 4.2 Syntagmatic word associations

Syntagmatic associations are defined by the way that words line up in space or time and are the basis for ngrams and other phrasal units. They can be identified from a single occurrence in text without reference to syntax, semantics or statistics and govern two types of juxtaposition: aggregation, or clustering of related words, and structuring, or organization of related units of text (Thompson, 1977). Juxtaposition can be contiguous or proximate within some unit of text e.g. a text window, sentence, document or other unit.

From a functional point of view, syntagamtic sequences are important in many languages because they are efficient and natural units that reduce cognitive load during communication (Sinclair, 1987a, 1991). Fewer selections are required to construct

sentences using aggregations of words (phrases) than individual words. Predictability of word sequences can also expedite language understanding. The observed regularity of certain syntagmatic sequences eventually led Sinclair (1991) to propose the principle of idiom (see Section 3.3.2), which holds that meaning is based on semi pre-constructed phrases. These phrases constitute single lexical choices, even if they appear to be analysable into segments (Sinclair, 1991).

Given that syntagmatic sequences facilitate language production and understanding, it is no surprise that they frequently identify important word relationships. For example, 87% of syntactic relations between words occur within a window of two words or less (Ferrer i Cancho et al., 2004).

### 4.2.1 The semantics of syntagms

Syntagms identify terms composed of semantically related words in many, but not all, cases. In order to meet assumptions that the juxtaposition of things in space and time is meaningful, the semantic composition of words, and groups of words, is often encoded by sequential positioning as well as syntactic structure (Tsarfaty, 2010). Failure to place related words near to each other causes confusion. Consider the following example:

(4.1) *What does the suitcase of the Chancellor that is red contain?*

(4.2) *What does the red suitcase of the Chancellor contain?*

The question in example 4.2 is much easier to understand than the question in 4.1 due to the surface proximity of '*red*' and '*suitcase*'. This is true even though the syntax of 4.1 is likely to indicate the same meaning as 4.2, given the default expectation that people are not red. Language understanding is facilitated by placement of related words and phrases next to each other. As a result, semantic relations are often expressed using word adjacency, and adjacent words often refer to elements that are semantically related.

This observation is predicted by the Gricean maxim of Manner (Grice, 1989), which holds that a communication, such as a query, should be *orderly*, brief, and avoid obscure expressions and ambiguity. The general idea is that a simple expression, in which related words are placed close together, is only avoided in favour of a more complex paraphrase, in which related words are separate, to communicate a semantics that is not read from the simple expression. This means that example 4.1 should only

be used if the wording of example 4.2 does not convey the correct meaning. For example, if the Chancellor is a red puppet. When it is not possible to place words that refer to semantically related elements next to each other, they are often placed in close proximity.

### 4.2.2 Limitations of syntagms

Not all syntagmatic sequences realize semantic relations between words. This is obvious for straightforward examples such as '*What security measures are in effect or are proposed to go into effect in airports?*' (Robust04 #412). Here, a sequence such as '*proposed effect*' is at best uninformative with respect to the topic and at worst misleading. In addition, syntagms cannot be distinguished by syntactic type, so there is often no way to determine which ones are more likely to describe semantic relations. Uninformative syntagms such as '*proposed effect*' have the same type as informative terms such as '*security measures*'. In some cases this limitation may have a considerable effect on query effectiveness. Consider the following queries:

(4.3) *application for a visa in America*

(4.4) *application for a visa to America*

Here, there is an ambiguity in the meaning of '*visa*' (a credit card or an immigration document). We might assume that query 4.3 is about visa credit card applications because it is unlikely that the user is looking for immigration visas if they have not specified a visa destination (at least, this is a reasonable position). Conversely, in query 4.4 the most probable interpretation of visas *to* America refers to immigration visas. Of course, it is also possible that these examples have an alternative intent or are attempts to communicate the same information need using poor English. As discussed in the Chapter introduction, this highlights the fact that semantics is not necessarily aligned with syntax or grammar. However, the focus here is on a plausible scenario.

Identification of appropriate semantic associations might enable 4.3 to retrieve at least some documents related to American credit cards, while 4.4 retrieves only those documents related to American immigration. At the time of writing, this could be demonstrated by typing the following keyword queries into an open domain search engine:[4]

[4]Queries were tested using the Google search engine in October 2012.

(4.5) *application "visa America"*

(4.6) *"application visa" America*

'Visa America' is the name of a credit card company in the United States, so 4.5 retrieved an item for visa credit cards at number two that did not appear in the top ten ranking for query 4.6. However, syntagmatic relations alone do not make the preferred segmentation clear. They are unable to differentiate between informative word combinations and sequences that are merely coincidental.

Finally, syntagms are unable to detect all semantic relations. They reflect the intuition that co-occurrent objects are related, and the closer they are, the more likely they are to be related. This means that they fail to detect long distance dependencies. They also miss important word relationships when expectations about the orderliness of aggregation and structure of text are not met. This can occur, for example, if a user constructs an awkward query, or if an information need is grammatically too complex to permit all related words to appear in proximity. It can also occur where knowledge of frequent events and familiar objects in the real world is assumed to indicate a relation between words (as discussed in Section 4.4.1).

## 4.3 Syntactic word relations

Syntax is a system of rules that govern the combination of words into well-formed phrases and sentences. Every subset of words in a sentence is syntactically related because a complete syntactic parse is represented by a single parse tree. However, the relation may be close or distant. In practice, a syntactic relation is usually defined to exist for only a subset of related words.

Typically, if a parse is a graph $G$ in which vertices are words, and possibly phrases, then there is a syntactic relation between the words $w_i$ in a term $t$ when:

1. A continuous sequence of edges in $G$ connects the vertices for all $w_i$ in $t$, and these edges connect only words or phrases that are subphrases of $t$. Specifically, the words $w$ are connected by relations of immediate dominance;
2. All of the words in $t$ are children of a vertex that does not govern any words that are not in $t$ (see Section 3.1.1 on governance).

Syntax excludes many accidental word associations and can identify terms containing both short and long range word dependencies. It may therefore help to precisely

identify informative terms. In addition, word combinations are characterized by structural information that can be exploited as a criterion for document matching. The type of word relation, and the type of unit that the related words compose (such as a noun phrase), can be matched in documents.

### 4.3.1 The semantics of syntax

Syntactically related word combinations derive their meanings from the meanings of their component words and the nature of their syntactic relations. As a result, syntax often identifies semantically related words. The structure of a syntactic graph may also facilitate detection of preferred word associations.

For example, consider the queries 4.3 and 4.4 (application for a visa in/to America) that posed a challenge in the previous Section. It is possible to assign various syntactic structures to each example, but the assignments shown in Figure 4.1 demonstrate that it is at least possible to construct representations from which discriminative word combinations are identifiable. Both '*visa America*' and '*application America*' in the desired segmentations are identifiable using the parses shown, ignoring any stopwords.

Despite this advantage, the fact that multiple parses are possible for each example suggests a problem. Syntactically defined terms may not always have the same representation in queries and documents. This, and other limitations of syntax for IR, are explored further in Section 4.3.2. The rest of this Section provides background on the nominal syntactic-semantic alignments for phrase structure and dependency grammars.

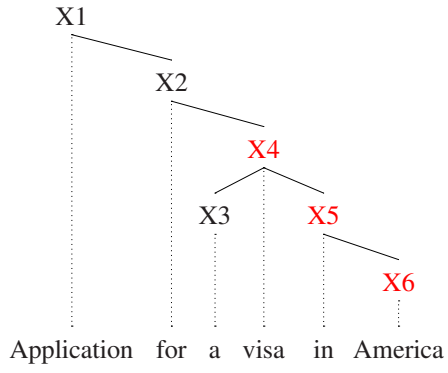
#### 4.3.1.1 Phrase structure theory

The view popularised by Chomsky (1995) is that there is a one-to-one correspondence between syntax and semantics, but this is not necessarily apparent in a surface phrase structure parse. Rather, there are multiple levels of representation, each with its own set of rules, and rules for semantic relations match (or nearly match) syntactic rules for at least one level. Apparent mismatches between semantics and syntax are resolved below the surface.

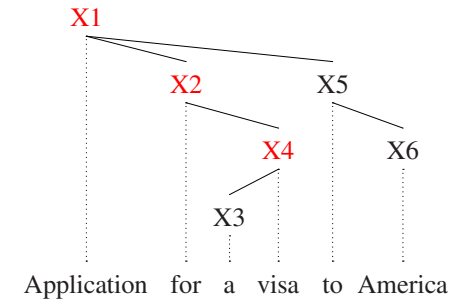
Chomsky (1965) observed that sentence semantics often align with what he called '*deep structure*' (D-structure).[5,6] Deep structure consists of the core logical relations

[5]Chomsky's recent work has abandoned the idea of D-structure, and its counterpart S-structure, in favour of Logical Form (LF) and Phonological Form (PF) (Chomsky, 1995), although the concept of transformation remains central.

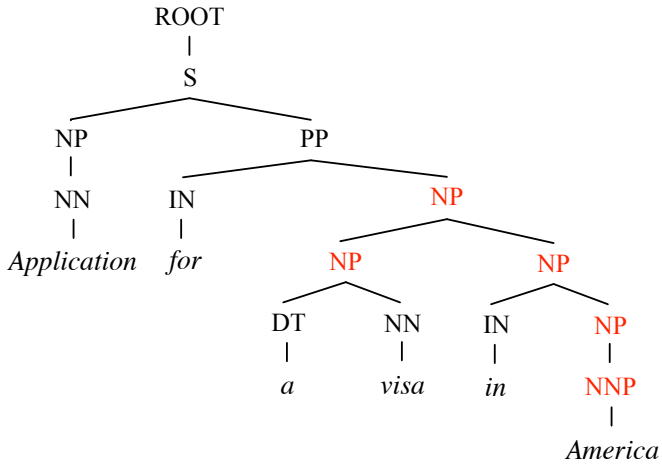
[6]See Chomsky (1995) and subsequent work on the Minimalist Program, plus May (1985). Chomsky gives the example of two sentences "*John is easy to please*" and "*John is eager to please*". These



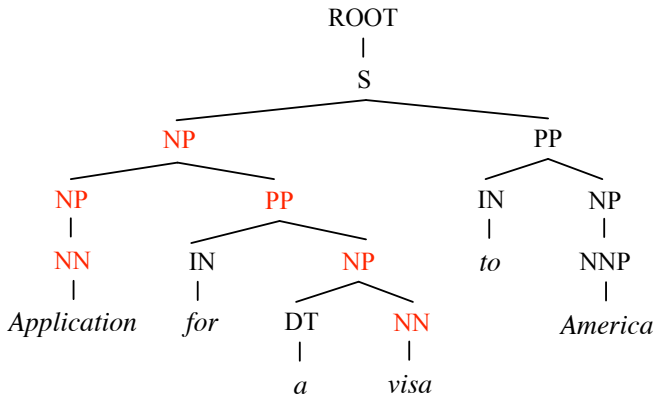

(a) Dependency: *application "visa America"*




(b) Dependency: *"application visa" America*

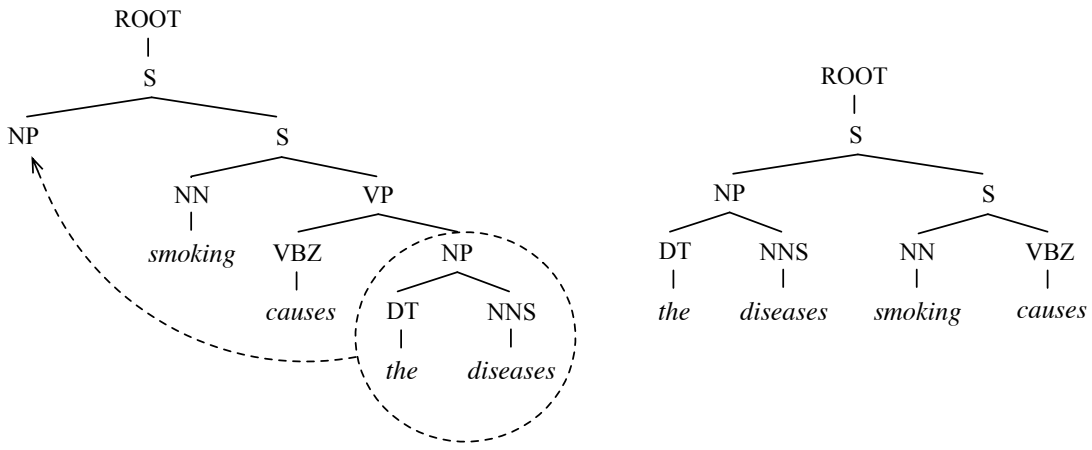



(c) Phrase structure: *application "visa America"*




(d) Phrase structure: "*application visa*" *America*

Figure 4.1: For queries 4.3 and 4.4, a suitable dependency parse (a) and (b), and a suitable phrase structure parse (c) and (d) permit identification of appropriate word relations.




(a) Deep structure and transformation

(b) Surface structure

Figure 4.2: Transformation rules describe derivations from deep structure to surface structure. Both structures are represented as parse trees.

between text elements and transformation rules that apply to these elements. The transformation rules describe derivations from deep structure to surface form, both of which are represented as parse trees. Word order can differ between structures because transformation rules effectively move words and phrases around either within a sentence or its tree representation (Figure 4.2).

An alternative to the Chomskyan view holds that there is an alignment between combinatory *rules* of phrase structure and semantics, and not the syntactic and semantic *structures* that are output by the rules. This theory is grounded in the work of Montague (1974) (cited by Koenig (2005)) who argued that there is no "important theoretical difference" between formal and natural languages (see also Bach (1976)). Montague outlined the rule-to-rule hypothesis that there is a pairing of semantic and syntactic rules such that each expression in language is assigned a syntactic and a semantic category, and combinatory rules apply to categories at both levels of representation (syntax and semantics). For example, the expression '*finds*' takes the syntactic category of a verb phrase missing either a noun phrase to its left or its right, or both. Semantically, '*finds*' is a predicate missing either an agent to its left, an object to its right, or both. This makes it possible to have alternative syntactic and semantic interpretations of sentences that result from non-deterministic interactions of combinatory principles.

sentences have the same surface syntactic structure, but in the deep structure of first sentence, '*John*' is the direct object of the verb '*please*', with the meaning "*It is easy for someone to please John*". In the second sentence '*John*' is the subject of the verb '*please*', with the meaning "*John is eager to please someone*" (Searle, 1972).

Some influential articulations of this approach are Combinatory Categorial Grammar (CCG) (Steedman, 2000), and theories proposed by Klein and Sag (1985) and Copestake et al. (2001). This approach has an advantage in terms of flexibility, and might help to explain the effectiveness of proximity measures for IR. Ordered proximity measures look for a word or phrase within a certain distance to the right or left of another word or phrase and thus reflect the idea of an alignment between syntactic and semantic rules.

#### 4.3.1.2 Dependency theory

Unlike phrase structure grammars, monostratal grammars used by most dependency parsers assume that semantics is aligned within one level of syntactic representation, namely the structure of governor-dependent relations. This is possible because dependency grammars, like deep structure, handle many correspondences between a single semantic meaning and multiple surface representations. In fact, dependency grammars bear a strong resemblance to deep structure.

Both dependency grammars and deep structure use rules that underspecify word order to permit a one-to-many mapping from logical structure to surface representations. This enables them to capture underlying sentence semantics. By consequence, dependency structures are more likely to match different lexicalizations of the same semantic content than surface phrase structures. For example, Figure 4.3 shows multiple phrase structure representations for a query with the semantics '*What diseases does smoking cause*' (WT10G #511). In the Figure, (a) and (b) correspond to the single dependency representation in (c).

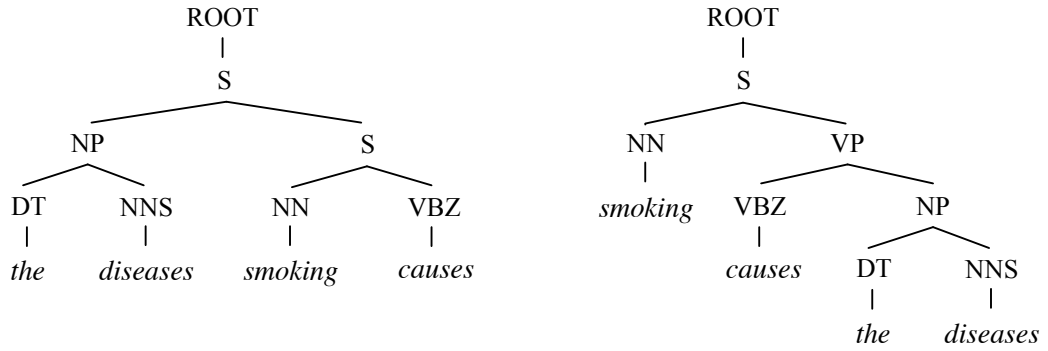
The normalizing effect of dependency grammars is also maintained in cases where alternative lexicalizations do more than shuffle grammatical categories within a sentence. Consider the following two queries in which the semantic unit '*school prayer*' is intact in one query and discontinuous in the other:

(4.7) *Was school prayer banned in the U.S.?*

(4.8) *Was prayer in U.S. schools banned?*

The dependency structures in Figure 4.4 (a) and (b) illustrate that the semantic relation between '*prayer*' and '*schools*' (prayer *in* schools, see Levi (1978)) is accessible for both examples if stopwords are ignored.[7]

[7]Note that this relationship is not necessarily accessible with phrase structure grammar. For example

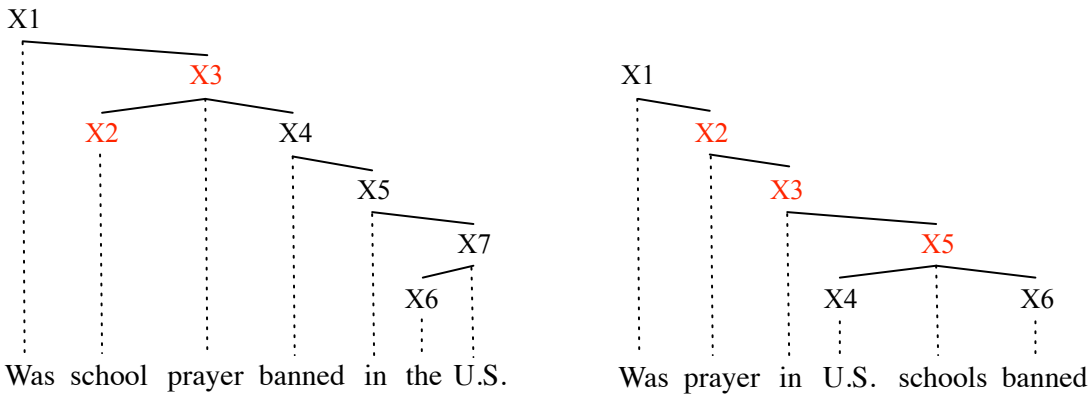



(a) Phrase structure

(b) Phrase structure

(c) Dependency structure

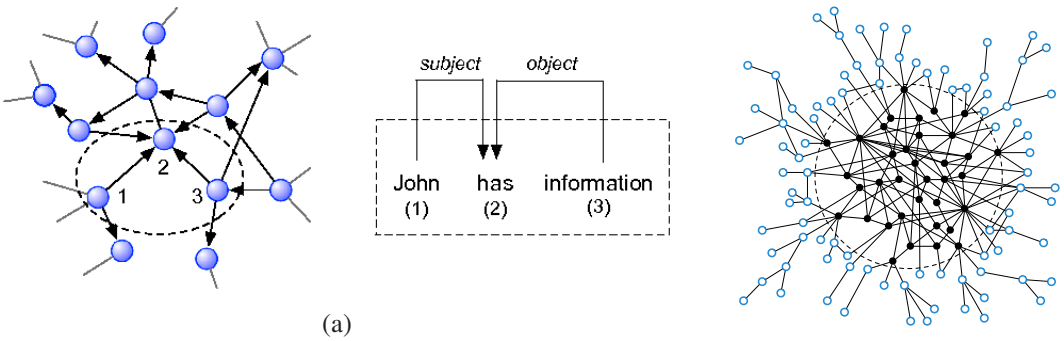
Figure 4.3: Alternative phrase structure representations corresponding to the semantics of the query '*what diseases does smoking cause?*' (a-b), and a single corresponding dependency representation (c).




(a) Dependency parse for example 1.7.

(a) Dependency parse for example 1.8.

Figure 4.4: There is a collapsed dependency relation between '*prayer*' and '*schools*' for both example 4.7 and 4.8.




(a) (b)

Figure 4.5: Structure of a large-scale dependency network: (a) the network is constructed by assigning words in a corpus to nodes in a graph, and drawing directed arcs between nodes that appear in a governor-dependent relationship; (b) high frequency words appear in the center (shown in black) and more content-bearing words show towards the edges (outlined in blue). Figures taken from Ferrer i Cancho et al. (2004)

Naturally, since dependency grammars are quite effective for the normalization of syntax given a single semantic intent, they also identify semantic relations reasonably well. When used to build a large-scale dependency network, the network structure may also help to discriminate words with greater semantic specificity.[8] A dependency network is a graph in which words in a corpus are nodes, and directed arcs link nodes that appear in governor-dependent relationships (Figure 4.5 (a)). In a large-scale, or global, dependency network, the most connected nodes are high frequency, functional words found in the center of the graph, such as those shown in black in Figure 4.5 (b). More content-bearing words tend towards the edges, as shown by nodes outlined in blue. There is a linear relationship between node degree[9] and word frequency (Ferrer i Cancho et al., 2004; Ferrer i Cancho, 2005) such that semantic specificity is predicted to some extent by the position of the corresponding node in the network.

### 4.3.2 Limitations of syntax

Both phrase structure and dependency grammars are purported to align syntax with semantics at some level. However, when only surface syntactic features are consid-

the phrase structure parse for 4.7 contains a node that dominates all and only '*prayer*' and '*schools*'. However, for example 4.8 the relation between prayer and schools cannot be detected because no node dominates all and only these words. They coincide in a grammatical category that includes other words.

[8]The structure of large-scale dependency networks is shared across several languages (Ferrer i Cancho et al., 2004). Chomsky (1967) in his argument for deep structure also observed that deep structure "seem[s] to be very similar from language to language".

[9]The degree of a node is the number of edges, or arcs, that connect it to other nodes in the graph.

ered, as they typically are in IR, a one-to-one correspondence seems to be lacking. If such a correspondence existed, natural language would be like formal languages, such as mathematics, in which every rule or constraint on the combination of expressions aligns with a unique rule or constraint on the combination of semantic meanings. In mathematics, there is a unique way to evaluate every expression used to combine numbers. For example, if we combine 2 and 3 with the syntax of addition or multiplication, we have the expressions $(2+3)$ or $(2\cdot 3)$. There is one unique rule to compute the meaning, or semantics, of each expression such that $(2+3)$ *means* 5 and $(2\cdot 3)$ *means* 6. This homomorphism between syntax and semantics conforms strictly to the Compositionality Principle (Frege's principle): the meaning of an expression is a function of the meaning of its parts and the way the parts are combined (Koenig, 2005).

Conversely, there are strong reasons to believe that the relationship between structure and semantics in natural language is not so straight-forward. It is argued that syntax and semantics are independent generative systems, and the illusion that a homomorphic relationship exists between them is due to default interpretations of syntactic structures (Jackendoff, 1997; Sadock, 1991). These interpretations arise from a receiver's need to infer what a communicator intends based on a particular set of circumstances in which a communication takes place. The communicator can narrow the field of the receiver's possible inferences using the syntax of language. Repeated use of particular ways of formulating language to successfully constrain inference eventually become conventionalized and expected, and thus appear to be grammatical rules with varying degrees of flexibility (LaPolla, 2006). However, a one-to-one correspondence between syntax and semantics is not required since in some circumstances an unclear statement is sufficient for a communicative intent.

A lack of homomorphism between syntax and semantics makes it unlikely that any IR system relying on syntax will detect all semantic relations described by a query. It also makes it unlikely that the system will identify all semantically relevant documents for a query, even if there is no vocabulary mismatch. There are three core problems that limit syntactic-semantic alignment for IR:

1. A single semantics can be interpreted from multiple syntactic interpretations, even within a single syntactic theory.
2. A single syntactic representation can have multiple semantic interpretations.
3. Some semantic associations are not readily identified by any syntactic structure. This can be a problem for both grammatical and ungrammatical texts.

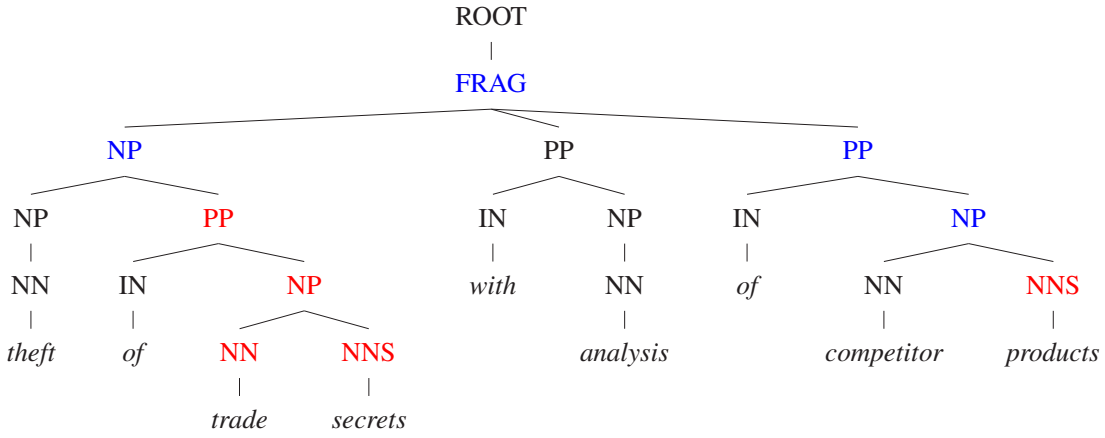
The first problem is more of a challenge for some theories than others. Dependency grammars tend to normalize syntactic representations (although imperfectly, as will be shown) whereas phrase structure parsing frequently results in multiple surface syntactic representations for a single semantic meaning. Discussion on this point is therefore deferred to the theory-specific Sections below.

The second problem is quite generic, and results in ambiguity for both information needs and document content. For example, '*I like her cooking*' in Section 3.1.5 had many possible interpretations, including '*I like the way that she cooks*' and '*I like the fact that she is being cooked*'. This is a problem of syntactic ambiguity. In addition, POLYSEMOUS words can create semantic ambiguity that is not alleviated by syntax. For example, '*trade secrets*' can refer to secrets about import and export operations, secrets held by a specific industry or company, and the exchange of confidential information of any sort. The syntactic relation between '*trade*' and '*secrets*' is easily identified but not particularly informative.

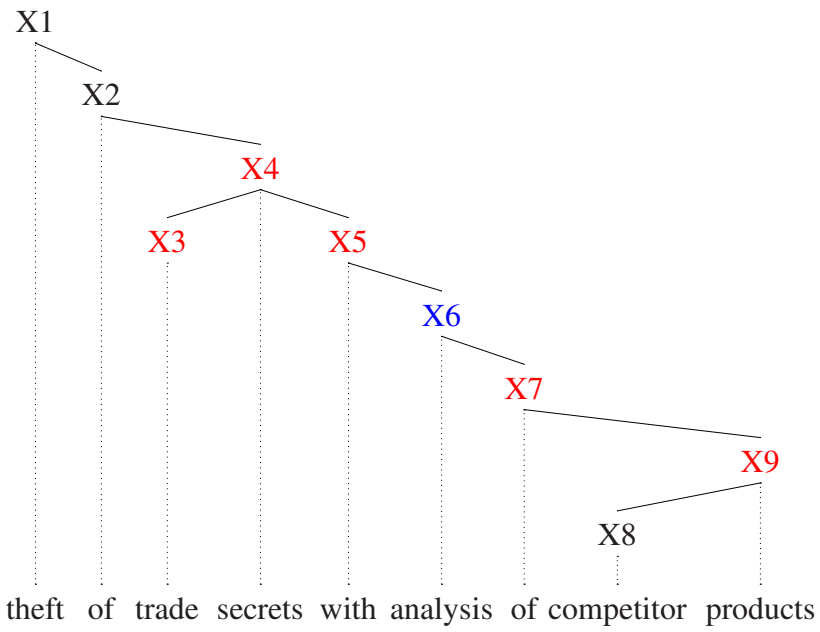
Context can alleviate ambiguity with polysemous words. For example, the term '*trade secrets products*' from the query '*theft of trade secrets with analysis of competitor products*' (adapted from Robust04 #311) can lend the semantic interpretation '*trade secrets about products*'. However, '*trade secrets products*' cannot be identified using syntax, as Figure 4.6 shows. There is no connection between '*trade secrets*' and '*products*' that does not pass through a vertex dominating other words in either a phrase structure parse or a dependency parse. Syntactic rules alone cannot unravel the semantics of every sentence.

The third problem for any theory of language in IR is that some semantic associations are not readily identified by any syntactic structure as just illustrated by '*secrets about products*'. Another example is the query '*why bacteria seem to be beating antibiotics*'. The words '*bacteria*' and '*antibiotics*' refer to semantically related elements because antibiotics are used for fighting bacteria (*x for y*, see (Levi, 1978)). The term '*bacteria antibiotics*' is also likely to be an effective search term for the information need. However, '*bacteria*' and '*antibiotics*' are not related in the most probable phrase structure parse or dependency parse, as Figure 4.7 shows. For IR, there is also the problem of texts that are not suitable for syntactic analysis. Open domain queries are often ungrammatical, as are many informal documents such as blogs and forum posts.

This collapse of an alignment between syntax and semantics is symptomatic of both LEXICAL RELATIONS and textual economy. Lexical relations are associations that are interpreted using the semantics of words and encyclopaedic knowledge of the world

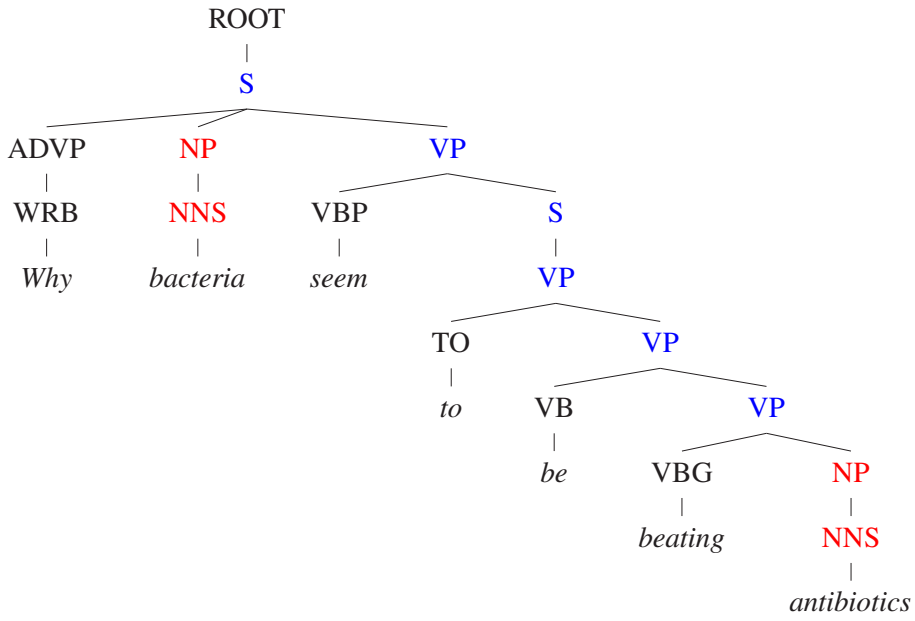



(a) Phrase structure parse

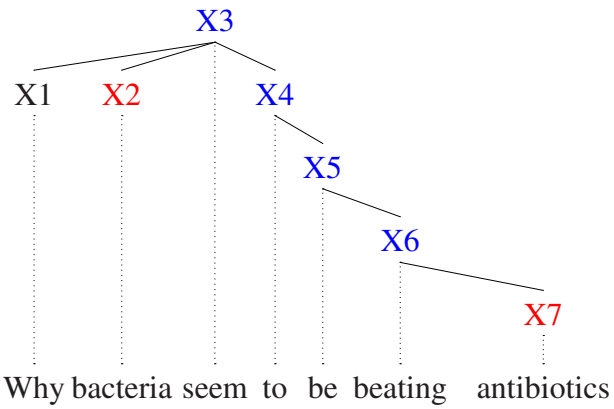



(b) Dependency parse

Figure 4.6: A phrase structure and dependency parse of the query: "*theft of trade secrets with analysis of competitor product*". Neither structure detects a relation between '*trade secrets*' and '*products*'




(a) The most probable phrase structure parse




(b) The most probable dependency parse

Figure 4.7: Neither a phrase structure parse nor a dependency parse of the query '*why bacteria seem to be beating antibiotics*' identifies an association between '*antibiotics*' and '*bacteria*'

around us, rather than the semantics of words and syntactic relations. For example, the component words of '*electrical clock*' and '*musical clock*' are lexically related because it requires world knowledge to understand that an electrical clock is powered by electricity, and a musical clock plays music ( (Levi, 1978), as cited by Giegerich (2005)). Although the words are frequently syntactically related or contiguous, there is no need for them to be so. Notice that for the query '*why bacteria seem to be beating antibiotics*' it requires world knowledge to understand that antibiotics kill bacteria, and a syntactic or contiguous association between '*bacteria*' and '*antibiotics*' is not required for the association to be inferred.

Lexical relations are discussed further in Section 4.4.1 in a presentation of the semantics of statistical word associations. In the meantime, it is worth noting that they are frequently relied upon for interpretation of *textual economy* (Stone and Webber, 1998). Textual economy condenses information in language by overloading words or clauses whose primary function is to serve some other communicative goal. In order to recover word associations, a receiver must infer links between the text units using real-world knowledge. Textual economy has particular relevance for IR because users often condense information in queries.

To estimate the frequency with which textual economy (and lexical relations) occur in verbose queries, a sample of 100 randomly selected Robust04 and GOV2 topics were examined. Two assumptions facilitated analysis. First, all possible 2-word combinations in a title query were assumed to be informative. Second, a title query was assumed to capture a complete, succinct information need. So, given a topic with the title '*blood alcohol fatalities*' and description, '*What role does blood-alcohol level play in automobile accident fatalities?*', the terms {*blood alcohol, alcohol fatalities, blood fatalities*} were assumed to capture the information need. Call these terms a pseudo informative set. Analysis counted how many pseudo informative terms could be identified using the syntax of a description topic. To overcome differences in lexicalization, words in title queries and descriptions were normalized. For example '*pharmacists*' was normalized to '*pharmacist*' and '*tooth*' was normalized to '*dentistry*'.

If there is no textual economy in natural language descriptions then most 2-word combinations in a title query should be identifiable from a syntactic parse of the corresponding description topic. However, analysis revealed that 22% of queries contained at least one pseudo-informative association that could not be detected using either phrase structure or dependency relations. An additional 11% contained at least one association that could only be detected using dependency relations. Missing associa-

tions were semantic relations, not spurious links. For example, processing applied to the description query, '*Do police departments use profiling to stop motorists*' (#432), failed to detect any of the word associations in the corresponding title query '*profiling motorists police*', namely {*profiling motorists, profiling police, motorists police*}. This suggests that there are a substantial number of cases in which semantic associations cannot be identified using syntax. A full review of the missing associations in queries for which neither phrase structure nor dependency relations were adequate is provided in Appendix B.

To better understand the limitations of phrase structure and dependency theories with respect to semantic interpretation, these theories are considered separately below.

#### 4.3.2.1 Phrase structure theory

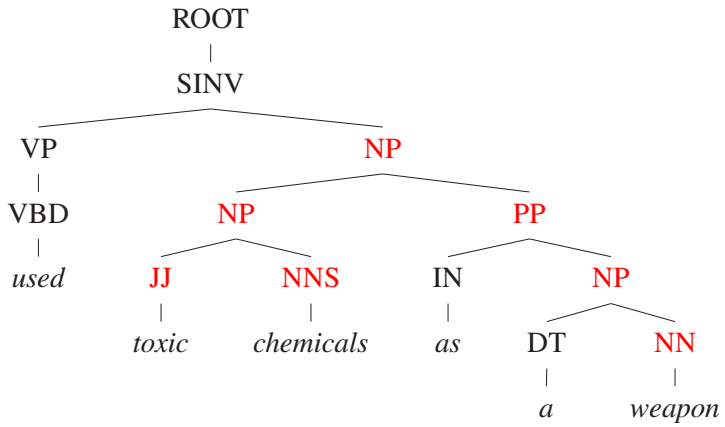
- **Deep structure:** The multi-stratal nature of phrase structure theory is problematic for IR techniques that assume semantic relationships are explicitly encoded in surface structures. This is because words referring to semantically related elements are not reliably detected by syntactic realisations when deep structure and surface structure differ. Consider the simple case in which a query contains *wh-fronting*. Wh-fronting occurs when the canonical form of a sentence, as typically observed in a document, is re-ordered to create a question:

  (4.9) Canonical: *used toxic chemicals as a weapon*

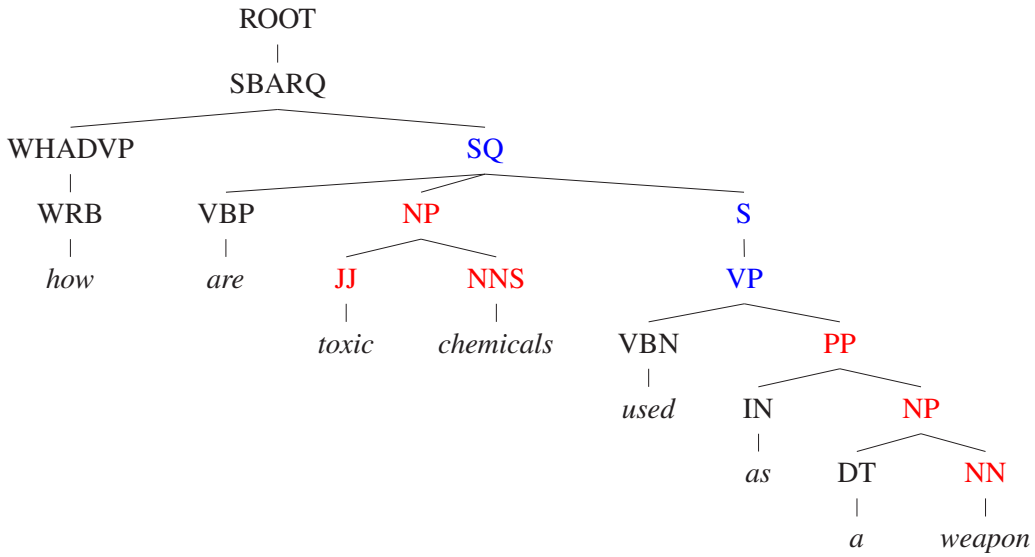
  (4.10) Wh-fronting: *how are toxic chemicals used as a weapon?*

  Ignoring the interrogative construction '*how are*', examples 4.9 and 4.10 share the same deep structure and semantic content. Transformation rules account for the movement of '*toxic chemicals*' to the front of the sentence. However, as shown in Figure 4.8, the most probable surface phrase structures for 4.9 and 4.10 are different. If stopwords are excluded, the relation between '*toxic chemical*' and '*weapon*' can be identified from the surface syntax of the canonical sentence and not the interrogative one ('*used*' intervenes in the interrogative form). This means that the term '*chemical weapon*', which is highly representative of the query semantics, cannot be identified from the surface interrogative form. Further, surface phrase structures are limited in their ability to identify the same semantic content in queries and documents.

- **Grammatical silos:** For the example in Figure 4.8, the grammatical categories of the related words are consistent: '*toxic chemicals*' is always an adjective-




(a) Canonical sentence (example 4.9)




(b) Interrogative sentence (example 4.10)

Figure 4.8: The canonical and interrogative forms of a sentence can have different surface phrase structure representations but the same core semantic content.

noun combination and '*weapon*' is always a noun. A major challenge for phrase structure grammars in IR is that grammatical categories for related words can cut across relevant relationships. Consider variations on the relation between '*strong*' and '*argue*' (Halliday, 1966):

(4.11) He argued strongly.

(4.12) There was strength to her arguments.

(4.13) His argument was strengthened by evidence.

(4.14) It was a strong argument.

The statements in (4.11)-(4.14) contain information that would be relevant to a query about '*strong arguments*'. However, there is no regularity in the grammatical categories associated with inflections of '*strong*' and '*argue*'. The grammatical categories in each case are (4.11) verb-adverb; (4.12) noun-noun; (4.13) noun-verb; (4.14) adjective-noun. This variety makes it even less likely that queries and documents will match on a syntactic basis. In fact, it is the main reason stemming so often improves IR performance. Examples like this led Halliday to claim that, "It is not merely irrelevant the particular grammatical relations they [words or linguistic units] enter into - it may also be irrelevant whether they enter into a grammatical relation at all." (Halliday, 1966).

- **Nonconfigurationality:** Configurationality describes a pattern of correspondence between formal syntactic constituents and the logical structure of sentences. Logical roles (sometimes called *grammatical functions*) are abstract roles based on the semantics of a sentence (Hudson, 2012). In configurational languages such as English, arrangements of constituents are indicative of logical relations. For example, the logical relation of *direct object* is indicated by a verb preceding a noun phrase within a larger verb phrase.

  However, this correspondence is not true for all languages. In particular, by definition, nonconfigurational languages, such as Arabic, lack a correspondence between constituent structure and logical relations. This is one basis for disagreement in the linguistics community about the degree to which we can presume that phrase structure has any alignment with logical, or semantic, content (Hale, 1983). The proposed ideals of semantic alignment for IR are therefore even less plausible for phrase structure grammars in non-configurational languages than they are in English.

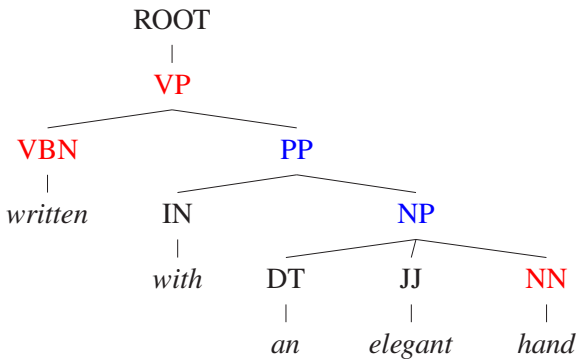



Figure 4.9: A phrase structure parse for example 4.16.

- **Word-phrase association:** Phrase structure contains phrasal nodes, so an individual word may be associated with phrasal nodes, rather than nodes that represent other individual words. As a result, phrase structure grammars are unable to capture dependencies between words when a word is associated with, for example, a noun phrase, rather than a word *in* a noun phrase. Consider the compound '*handwritten*'. There is a semantic relation between the words '*hand*' and '*written*' (written using a hand) and these words can appear near to each other or quite separate in text.

  (4.15) written by hand

  (4.16) written with an elegant hand

  Figure 4.9 shows that in 4.16, '*hand*' and '*written*' cannot be associated by surface phrase structure because the verb '*written*' is structurally linked with a node representing the prepositional phrase '*with an elegant hand*' and not with a node representing the word '*hand*' alone. The problem persists irrespective of the location for prepositional phrase attachment or improved parsing accuracy.

#### 4.3.2.2 Dependency theory

- **Semantic representation:** It has been argued (Smeaton et al., 1995) that dependency structure normalizes differences in surface word order because one dependency parse can represent multiple surface structures. This argument is supported by several examples in Section 4.3.1.2. However, while dependency grammars mitigate the problem of multiple syntactic representations for a single semantic meaning, there can be cases in which this still occurs in practice. For example, when there is uncertainty regarding prepositional phrase attachment, two dependency trees can be interpreted as having the same semantics. This is

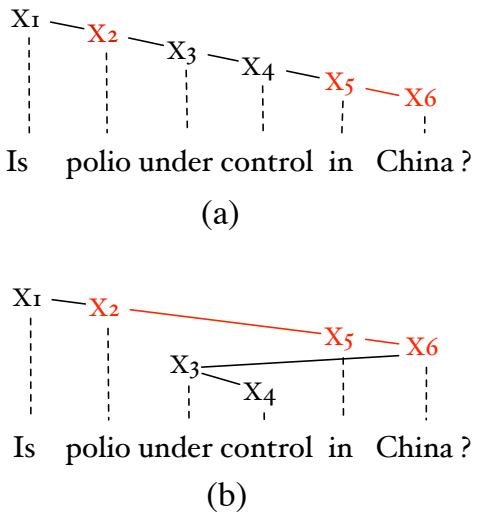



Figure 4.10: One word order can correspond to more than one dependency parse.

demonstrated in Figure 4.10, and discussed further in Section 6.2.2. As a result, a dependency representation may not capture all semantic relations. This one-to-many syntactic correspondence is less frequent for dependency parsing than for phrase structure parsing, but still limits the utility of dependency parsing for IR.

Overall, theoretical evidence pushes the conclusion that syntactic language processing may not be privileged for the identification of semantic relationships in IR. Dependency structure is more closely aligned with semantics than surface phrase structure but also does not perfectly capture the meaning of text.

## 4.4 Statistical word associations

Statistical word associations capture a word's ability to combine with other words (see Section 3.3.2). All statistical association methods measure the strength of association, or 'glue', between words and can be used to either classify or rank word combinations based on how likely it is that their co-occurrence is an instance of collocation. Here, *collocation* refers to word pattens that are in some way significant, and *co-occurrence* refers to patterns in which words share textual context, irrespective of their relationship.

From a data-driven perspective, statistical word associations quantify observable properties of language using word frequencies and word co-occurrence. Words are not chosen at random during communication, so the presence or absence of one word provides information about the probability of observing other words (Losee Jr., 1994).

| | $word_1$ | $\neg word_2$ | |
|---|---|---|---|
| $word_1$ | $O_{xy}$ | $O_{x\neg y}$ | $O_{xz}$ |
| $\neg word_2$ | $O_{\neg xy}$ | $O_{\neg x\neg y}$ | $O_{\neg xz}$ |
| | $O_{zy}$ | $O_{z\neg y}$ | $O_{zz}$ |

Figure 4.11: A contingency table for word co-occurrence, used to calculate statistical word association measures.

The significance of word co-occurrence can therefore be computed by comparing the expected frequency of word co-occurrence $E$ with the observed frequency of co-occurrence $O$. These frequencies correspond to two hypotheses:

- **Hypothesis $H_1$:** The words $x$ and $y$ in a term are independent and occur together only by chance (null hypothesis);
- **Hypothesis $H_2$:** The words $x$ and $y$ in a term are dependent.

These hypotheses can be understood with reference to a contingency table containing four cells (Table 4.11). The cells show the marginals: the observed frequency of co-occurrence of words $x$ and $y$ ($O_{xy}$); the frequencies of $x$ and $y$ co-occurring with a word other than $y$ and $x$ respectively ($O_{x\neg y}$ and $O_{\neg xy}$); and the frequency of co-occurrent words neither of which are $x$ and $y$ ($O_{\neg x\neg y}$). The expected values, e.g. $E_{xz}$, of internal cells are calculated from the marginals. In the case of a term with two words, the expected value is the product of the relevant marginals divided by the sample size. For example:

$$E_{xy} = \frac{O_{zy} \cdot O_{xz}}{O_{zz}}$$

With reference to the observed and expected frequencies of word co-occurrence, association methods can be classified into two types: *measures of effect size* and *measures of significance*.

Measures of effect size take into account the statistical association between $x$ and $y$ as determined by how much the observed frequency of co-occurrence exceeds expected frequency (Evert, 2005). If the expected probability of co-occurrence $E_i$ (derived from the null hypothesis) for two words is very small, then even one or two observed co-occurrences produce an estimate of strong association. For this reason, measures of effect size are biased towards combinations including very low frequency words, and should always be used with a filter to exclude infrequent words from consideration. Pointwise mutual information (PMI) is perhaps the most well-known measure in this

class (see Section 4.5.7.1). Typically, words with a frequency count of less than 3-5 should be ignored (Evert, 2005), but the threshold may depend on the size of the corpus used to identify co-occurrences.

Measures of significance are mathematically more rigorous, and take asymmetry of word frequencies and sampling variation into account. This enables them to distinguish between collocations that are intuitively more or less important. For example, imagine that two collocations, '*the Iliad*' and '*must also*' (Evert, 2005) appear in a collection 10 times with the same expected frequency $E$. Their score using a measure of effect size is the same. However, '*Iliad*' is an infrequent word. If it is preceded by '*the*' in every instance, the collocation '*the Iliad*' could not have occurred more often. So, the observed collocation frequency $O_{xy}$ is constrained by the frequency of '*Iliad*'. Conversely, the frequencies of '*must*' and '*also*' are much greater than 10, and '*must also*' could easily have occurred more than 10 times. For this reason, it can be argued that '*the Iliad*' is a more important collocation than '*must also*'.

Measures of significance, such as the log likelihood ratio (see Section 4.5.7.2), take this kind of sampling variation into account using the likelihood of $H_1$. However, if the observed frequency $O_{xy}$ is very large, then even a small difference between $O$ and $E$ (a small change in a large data sample) can be statistically significant. This makes measures of significance biased towards combinations of words that co-occur with high frequency.

Statistical word association methods can also be classified based on their definition of word co-occurrence, and its application within a node-centric or unit view (Evert, 2005).

- In a *node-centric* view, the significance of a statistical relation between words depends on the frequency, or probability, with which a central word co-occurs with other words in a term. This may be calculated relative to the frequency, or probability, that the central word co-occurs with any word not in the term (Gledhill, 2000).
- A *unit view* holds words to be separate units. The association between words can be determined by the frequency of their occurrence together, or calculated from the degree of similarity between their textual environments in a large corpus.

For either view, the specific definition of co-occurrence can have a substantial impact on the rank order of word combinations. There are four main definitions of co-occurrence:

1. A common definition uses a somewhat arbitrarily determined *window of text*. The window is usually fairly small, on the order of 2 to 5 words (Evert, 2005; Stubbs, 1995). This agrees with early work by Sinclair who defined collocations to occur between words "within a short space of each other in a text" (Sinclair, 1991). In practice, the definition is often interpreted as a separation of up to four words (Krishnamurthy, 2005; Sinclair, 1987b). However, much larger windows, up to several hundred words, are sometimes used;

2. Co-occurrence can be defined within a *textual unit of variable length*, such as a sentence or document. An approach based on textual units captures strong associations, such as compound nouns, as well as weaker ones that are often separated by some distance, such as synonyms and other semantically related words. It also helps to overcome the problem of prepositional phrases causing related words to be separated by more than a small window of text;

3. Co-occurrence can be defined with respect to a *dependency structure*. The structure can capture distant word relations and exclude accidental co-occurrences that intrude when larger spans of text are considered. In addition, word co-occurrence can be measured separately for different dependency types. However, some linguistic phenomena do not typically manifest as syntactic relations and this approach is limited in the type of word relations it can detect;

4. Words considered to be co-occurrent may be entirely separate in text, linked by reference to an *external resource* such as a thesaurus (Medelyan, 2007).

Overall, what distinguishes collocation from mere word co-occurrence is not the definition of co-occurrence, but the point at which statistics are deemed to indicate that word co-occurrence is not accidental (Gledhill, 2000).

### 4.4.1 The semantics of statistical associations

One of the main problems with statistical associations, as pointed out in Section 3.3.2, is that there is no theoretical basis for a clear distinction between collocations and accidental co-occurrences. Yet despite this imprecision, statistical methods have an advantage over other classes of word associations in IR because they do not make rigid assumptions about language structure. This enables them to detect a wide variety of linguistic phenomena, as well as alternative lexicalizations of specific semantic relationships.

In particular, statistical methods can detect lexical relations. Lexical relations are not governed by syntactic compositionality or word sequence (Section 4.3.2) and always appear in recurrent relationships in text (Giegerich, 2006). By consequence, they have a unique affinity with statistical association that is not found for many more prototypical dependencies. Moreover, lexical relations may be critical for IR. For example, they often occur between words that appear as noun phrases and compounds in alternative lexicalizations of semantic content. Salient occurrences of lexical relations are:

- **Adjective-noun combinations:** Normally, adjective-noun combinations such as '*Chinese protesters*' form noun phrases with an *ascriptive* relation where one word refines or adds to the meaning of another (the pattern 'something *is* something'). As such they are relatively easy to detect by various methods of word association. However, there are cases in which adjective-noun combinations have a *non-ascriptive* function (the pattern 'something *is associated with* something'). In these cases, there is no requirement that the words are either syntactically related or contiguous in text. Consider '*Chinese protesters*" as a candidate term for the topic '*What has been the outcome for the pro-independence protesters in Tibet who were arrested by Chinese authorities?*' (Robust04 #612). '*Chinese protesters*' is a frequent term in articles about protests against the treatment of Tibet, yet '*Chinese*' does not always describe the protesters. It can describe the authorities that the protesters oppose (and with whom they are associated). The relation between '*Chinese*' and '*protesters*' in this case is lexical and must be interpreted using world knowledge. It is not accessible using syntax or ngrams. A second example can help to make the pattern clear. In the phrase '*dental decay*', '*dental*' does not directly describe a property of the decay. It describes a property of something associated with the decay, namely a tooth that is decaying (Giegerich, 2006).
- **Noun-noun combinations:** According to Levi (1978), some COMPLEX NOMINALS result in lexical relations because they delete or incorporate a predicate in a COMPOUND NOUN before it is represented in surface text.[10] Deletion of the se-

[10]Conversely, some complex nominals are easy to detect with dependency relations and ngrams. It is generally easy to detect cases in which a nominalized verb acts as a head noun, and a modifier placed before the noun is derived from either the underlying subject or direct object of the nominalized verb (Levi, 1978). For example, given the query '*efforts by world governments to seek reduction of foreign debt*' (adapted from GOV2 #705), the implied verb '*reduce*' can act as a head noun ('*reduction*'). Complex nominals are derived for '*reduction*' from the argument structure of the implied verb, in this case, '*governments reduce debt*'. They are '*government reduction*' (underlying subject) and '*debt reduction*' (underlying direct object).

mantic predicates {*cause, have, make, use, be, in, for, from, about*} account for most complex nominals. For example, consider '*Saudi laws*' as a candidate term for the topic, '*Provide any description of laws or restrictions affecting Saudi Arabian women's rights*' (GOV2 #790). The predicate '*for*' has been deleted from '*Saudi laws*' (laws for Saudis, or laws for Saudi Arabia). Crucially, in predicate deletion the predicate does not leave a TRACE so there is no guarantee of a syntactic word relation. People typically infer the requisite relationships, even if the words are quite separate in text. In this case, '*saudi*' and '*law*' are neither adjacent nor syntactically related. Rather, it requires world knowledge to understand that '*saudi*' refers to the country Saudi Arabia for which there is a code of law.

- **Compounds:** Single words created by a combination of smaller words (e.g. *data-base, database*) have a semantic relationship that is unchanged when the words are distant in text. This relation is lexical, often involving a deleted predicate (as described for complex nominals). Compounds that appear as contiguous phrases in text have been studied in IR (Krovetz, 1995) but the incidence and effect of non-contiguous compounds has not been studied. In order to capture these relations, statistical measures may be required. For example, the relation between '*hand*' and '*written*' in the phrase '*written with an elegant hand*' (example 4.16) corresponds to the compound '*handwritten*' with the semantic relation 'written using a hand'.

One of the core strengths of statistical word association methods is their ability to detect lexical relations in addition to other linguistic phenomena. I contend that these relations may be pivotal to the success of techniques that account for the recurrence of word proximity in text, as well as word independence models for IR. Where dependence models fail to represent lexical word relations, their influence can be approximated by by considering words individually.

### 4.4.2 Limitations of statistical associations

- **Mathematical fitness:** Statistical word associations can be computed for very large quantities of text robustly and quickly. However, word co-occurrence data often fail to match the assumptions of statistical models (Evert, 2005). The null hypothesis (words are independent and occur together only by chance) is unrealistic because words are not combined at random. Words are used for communication and are therefore subject to conventionalized linguistic patterns that

improve language interpretability. They are also subject to semantic constraints associated with events in the real world to which they refer. When language data is collected from a large corpus, even a small deviation from the null hypothesis (a small change in a large data sample) can result in inflated association scores. This may be problematic for rare words that always co-occur with each other (e.g. *dèjá vu* in English). Moreover, the null hypothesis is even less realistic for terms with more than two words because it results in very small expected frequencies.

- **Descriptive ability:** Statistical word association measures have greater sensitivity to some types of linguistic phenomena (Pecina and Schlesinger, 2006; Tan et al., 2002). For example, mutual information highlights low frequency and high attraction collocations, such as proper nouns, but is less successful for the detection of strong, frequent associations such as some compound nouns (Evert, 2005). The type of phenomena identified depends both on the measure used and the definition of co-occurrence e.g. a window size, text unit, syntactic or semantic relations.

  A number of comparative evaluations of statistical word association measures have been carried out, but provide no conclusive evidence that one measure is consistently better than others (Evert, 2008; Tan et al., 2002). The 'best' association measure for a task depends on the target collocations, the language and properties of the data collection, and heuristic choices such as the definition of co-occurrence. Even statistical word association measures that are sub-optimal from a mathematical point of view can be highly effective for tasks that match particular term extraction profiles.

  It seems likely that ad hoc IR requires the detection of a broad range of word association types. For this reason, a combination of word association measures may be effective. For example, Pecina and Schlesinger (2006) list 82 statistical measures, and demonstrate that a neural network using the collocation scores for 17 of them improves the detection of five collocation types (idioms, technical terms, support verb constructions, proper nouns and stock phrases) by 21%.

- **Spurious association:** Statistical associations are frequently based on syntagmatic word relations in arbitrary units of text e.g. a sentence, document, or 10 word window. Syntagmatic relations do not precisely identify semantic, or informative, terms and this can affect the precision of statistical word associa-

tion methods for the detection of informative terms. Smaller window sizes of 2-5 words reduce the chance of spurious word co-occurrence but increase the chance that associated words are excluded accidentally. For example, smaller windows miss the relationship between '*hand*' and '*written*' in the sentence, "*to imitate the elegant way documents were written in the past, start writing by hand regularly*". Longer windows are sometimes necessary, even for languages like English without free word order.

## 4.5 Methods of word association

This Section describes seven word association methods divided according to the classes just presented. The relationships between these methods and semantics are evaluated in Section 4.6. The methods are:

- **Syntagmatic:** bigrams (Seq2), trigrams (Seq3) and nterms (Nterm);
- **Syntactic:** noun phrases (NP), governor-dependent pairs (GDep), catenae (Cat) and bounded phrases (BPhr);
- **Statistical:** terms identified by pointwise mutual information (MI) and the log likelihood ratio (LogL).

### 4.5.1 Ngrams

Ngrams are sequences of adjacent words in text. In this dissertation, I use sequential bigrams (Seq2) and trigrams (Seq3) identified following removal of stopwords using the INQUERY stoplist (Allan et al., 2000). Consider the query $q$, which will be used as a running example in this Section: '*What types of cases were heard by the World Court (International Court of Justice)?*'. After removal of the stopwords {*what, of, were, by, the*}, the sequential bigrams are easily identified, as shown in Figure 4.12.

Sequential trigrams are not typically used for IR, but are a suitable baseline for comparison with three-word terms identified by other means. Two example sequential trigrams in the stoplisted query $q$ are '*types cases heard*' and '*cases heard world*'.

### 4.5.2 Nterms

Nterms are a generalization of NGRAMS that do not require component words to be contiguous (ngrams are sequences of $n$ contiguous words). They are defined using

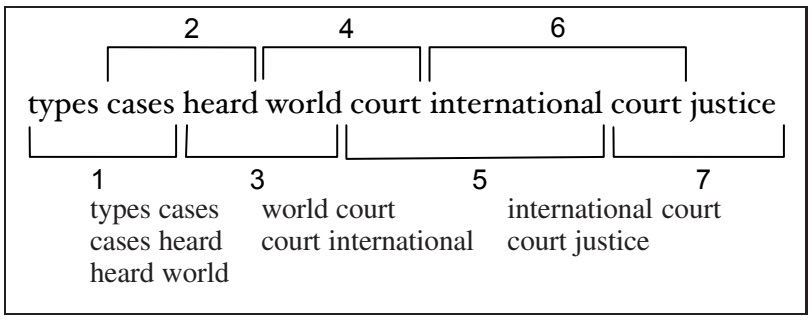



Figure 4.12: Sequential bigrams are all pairs of adjacent terms in a stoplisted query.

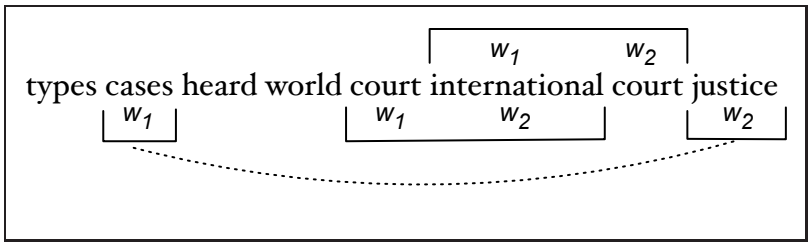



Figure 4.13: The words in nterms retain their word order from surface text: '*cases justice*' is a possible nterm but not '*justice cases*'. For the example query $q$ there are 77 nterms. For example, the nterms with two words that also contain '*types*' are {*types cases, types heard, types world, types court, types international, types justice*}.

$\wp(x)$, the power set of all words in a text $x$. $\wp(x)$ is the set of all possible subsets of words in $x$, including the empty set and the $x$ itself.

In this dissertation, nterms are subsets of $\wp(x)$ with a size of $1 \leq n \leq 3$. $x$ is a query excluding stopwords, and the words in each nterm retain their surface word order. This facilitates matching against documents when nterms are applied in IR models that impose constraints on word order. As shown in Figure 4.13, for query $q$ this means a possible nterm is '*cases justice*' but not '*justice cases*' because $w_1$ (*cases*) only ever appears before $w_2$ (*justice*). On the other hand, both '*international court*' and '*court international*' are nterms because '*court*' appears in the query twice, once before, and once after, '*international*'.

### 4.5.3 Noun phrases

A noun phrase is a sequence of words that can be substituted for a noun in text and the resulting sentence will be grammatically correct even if it does not make sense semantically (see Section 3.1.1). Noun phrases can be identified by parsing, or by CHUNKING a sentence into basic phrase types. In the case of parsing, noun phrases are normally defined on a phrase structure tree as shown in Figure 4.14. They can also be defined on a dependency tree.

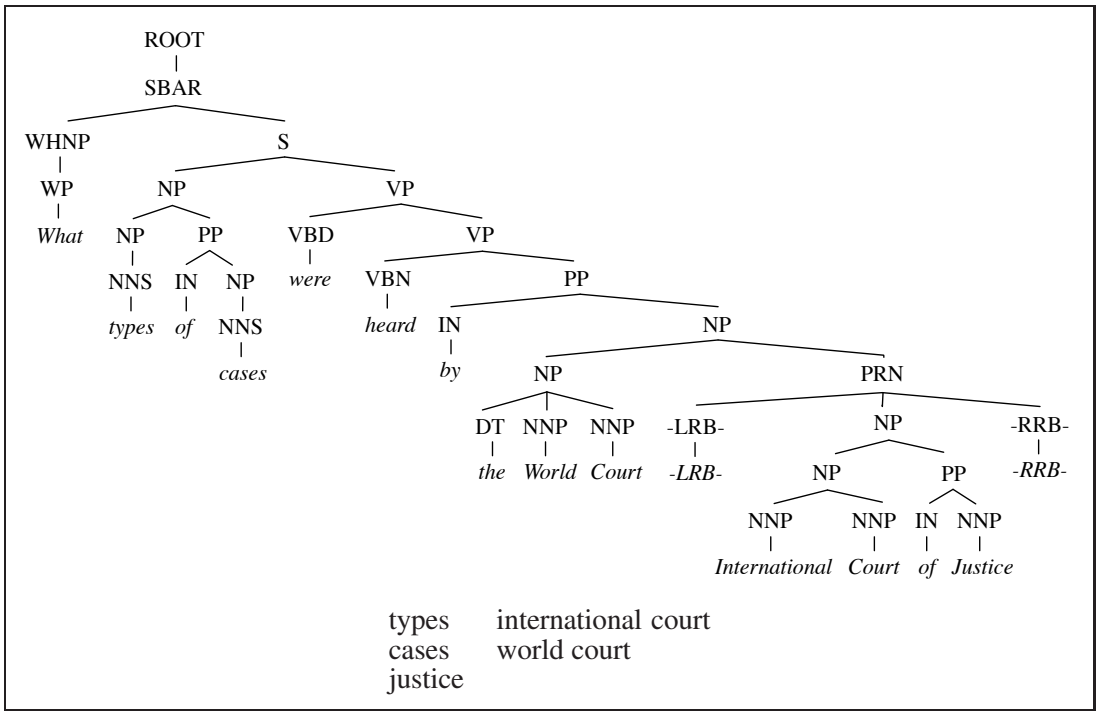
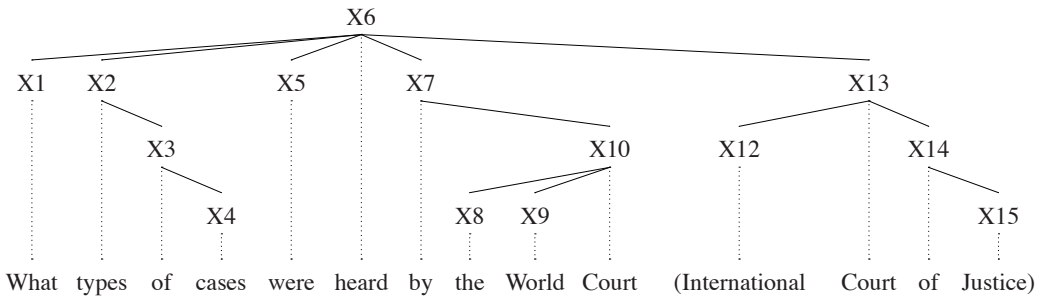



Figure 4.14: A phrase structure parse for the identification of noun phrases (NPs) in query $q$, and the noun phrases identified from this parse.




(a) Stanford format dependency parse

| | |
|---|---|
| types cases | heard types |
| world court | heard court |
| court court | court international |
| court justice | |

(b) Head dependent pairs

Figure 4.15: Illustration of (a) a Stanford dependency parse, and (b) the extracted collapsed governor-dependent pairs.

In this thesis, noun phrases are identified using phrase structure parsing output by the MontyLingua natural language processing toolkit (Liu, 2004). Only the smallest possible noun phrases are used, such that there are no phrases that are combinations of two or more noun phrases e.g. '*the world court (international court of justice)*' in Figure 4.14. Stopwords are removed before phrases are applied in an IR model.

### 4.5.4 Governor-dependent pairs

Head dependent pairs are defined on a dependency parse and consist of two words in a governor-dependent relation. Experiments in this Section and Chapter 5 use governor-dependent pairs generated by the Stanford pCFG (probabilistic context free grammar) parser. This parser makes suitable choices about headedness for applications that are sensitive to semantics and provides a collapsed dependency format (de Marneffe et al., 2006). The collapsed format removes vertices representing stopwords in order to narrow the distance between semantically related nodes in a dependency graph. For example '*plants* $\overset{nmod}{\longrightarrow}$ *in* $\overset{pmod}{\longrightarrow}$ *water*' becomes '*plants* $\overset{prep\_in}{\longrightarrow}$ *water*' (see Figure 4.15).

### 4.5.5 Catenae

A catena (plural 'catenae') is a word, or sequence of words, that are continuous with respect to a walk on a dependency graph (Osborne and Groß, 2012). Some examples of catenae are shown in Figure 4.16 for the sentence "*This tree illustrates the chain unit*". The dependency parse of this sentence generates 22 catenae in total: (using $i$ for $Xi$) 1, 2, 3, 4, 5, 6, 12, 23, 36, 46, 56, 123, 236, 346, 356, 456, 1236, 2346, 2365, 12346, 12356, 123456. Visually, a line can be drawn that connects the nodes in a catena without passing through any nodes that are not in the catena.

Words in catenae can be discontinuous in surface text (Figure 4.16 (a)) or continuous (Figure 4.16 (b)). They can also be constituents (b) or not constituents (a). Moreover, words in catenae can be continuous even if they are not constituents. The sentence '*This tree illustrates catenae*' in Figure 4.17 shows an example of this peculiarity. The sequence '*tree illustrates catenae*' is an ngram and a catena but it is not a constituent.

These Figures show that catenae identify a type of word association that is distinct from constituents, ngrams and any simple intersection of these criteria. Structures similar to catenae presented elsewhere make reference to *partial trees* and *dependency constituents*, where a dependency constituent is a constituent in phrase structure gram-

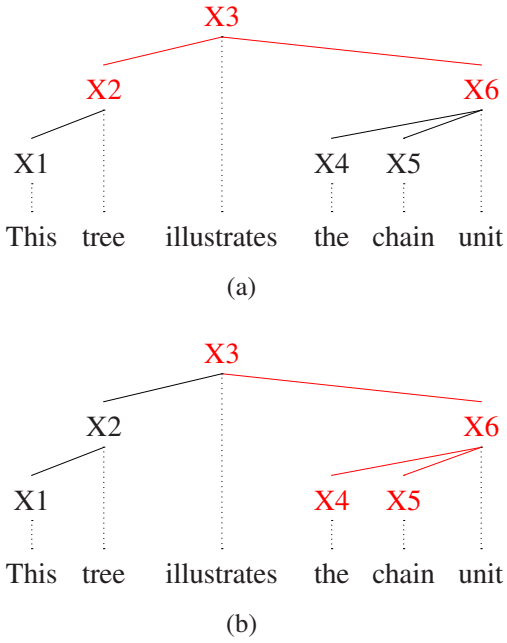
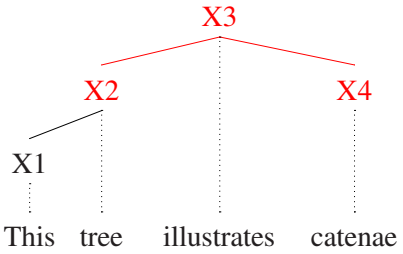



Figure 4.16: Examples of catenae paths shown in red. A line can be drawn that connects the nodes in a catena without passing through any nodes that are not in the catena.

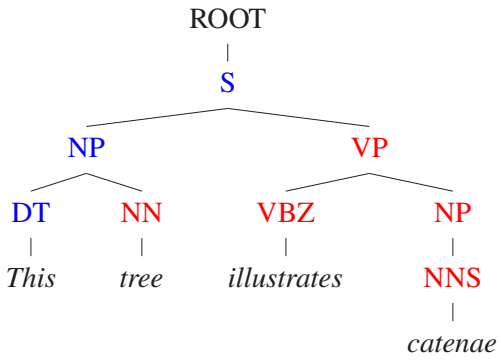
mar that is defined on a dependency parse ( Kunze (1975) and Pickering and Barry (1993) cited by Osborne (2005)). However, the formal definition of catenae does not refer to either subtrees or constituents, but dependency structure (O'Grady, 1998).[11]

Catenae are an economical and intuitive representation of paths on a dependency tree. Dependency paths and catenae differ in that a path is ordered and includes both word tokens and the relations between them, whereas a catena is a set of word types that may be ordered or partially ordered. This is illustrated in Figure 4.18 using examples of catenae for the topic '*Is polio under control in China?*' (adapted from Robust04 #302). Notice that the dependency paths includes relations between nodes that are absent in catenae. In this dissertation, catenae are post-processed to remove stop words on the INQUERY stoplist (Allan et al., 2000) and 18 TREC description stop words such as '*describe*'. This results in catenae such as the ones shown in Figure 4.18. For example, '*control in China*' becomes '*control China*'.

[11]The formal definition for catenae, referred to as "chains", is given as "Words *A*...*B*...*C*... (order irrelevant) form a chain iff *A* immediately dominates *B* and *C*, or if *A* immediately dominates *B* and *B* immediately dominates *C*" (O'Grady, 1998). Note that chains in this case do not refer to lexical chains. Lexical chains are independent of grammatical structure and use resources such as WordNet, whereas catenae are defined on dependency graph and use grammatical structure. For clarity, the terminology of '*catenae*' is preferred to '*chains*'.

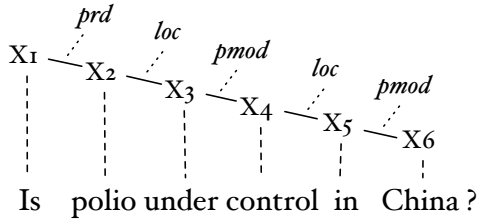



(a) The text '*tree illustrates catenae*' is a catena and an ngram.




(b) The text '*tree illustrates catenae*' is not a constituent.

Figure 4.17: Not all catenae that are also ngrams are constituents. In this sentence, '*tree illustrates catenae*' is an ngram and a catena, but not a constituent.




| **Catenae (stoplisted)** | **Dependency paths** |
|---|---|
| polio | |
| polio control | polio —loc→ under —pmod→ control |
| control | |
| control China | control —loc→ in —pmod→ China |
| China | |
| polio control China | polio —loc→ under —pmod→ control —loc→ in —pmod→ China |

Figure 4.18: Catenae are an intuitive representation of dependency paths.

Source: '*what types of cases were heard by the world court (international court of justice)?*'
Marked: // types of cases // // // // world court // international court of justice //

| | | |
|---|---|---|
| types | court | international court |
| cases | world court | court justice |
| types cases | international | international court justice |
| world | justice | |

Figure 4.19: Derivation of bounded phrases using frontier markers, and the resulting set of terms, including subgroups.

### 4.5.6 Bounded phrases

Bounded phrases are syntactically-derived words and word combinations used in the 1990s to extract the main topics or keyphrases from text (Turney, 1999).[12] Bounded phrases are similar to noun phrases, but are heuristically delimited by a set of frontier markers. These markers can be identified by any means, but typically include tokens like punctuation, prepositions, conjunctions and certain parts of speech. Frontier markers aim to demarcate certain word sequences including compound nouns, noun-adjective combinations, and noun phrases. For this reason, they may exclude prepositions such as '*of*' in order to detect phrases such as '*United States of America*'.

Bounded phrases may also be split into subgroups to create an expanded list of phrases. Subgroups are all terms consisting of one or more consecutive words in an extracted phrase. For example, the possible subgroups of '*international criminal court*' are {*international criminal, criminal court, international, criminal* and *court*}.

Frontier markers used to delimit bounded phrases in this dissertation include punctuation, prepositions other than '*of*', and all parts of speech except nouns and adjectives. Phrase subgroups are also included. Figure 4.19 shows an example of marker placement for the example query with // symbols and the resulting phrases. Bounded phrases can be effective for the selection of keyphrases, but the heuristics involved in their placement mean that certain long distance word associations are difficult to detect, particularly those affected by prepositional phrase attachment.

[12] Keyphrase extraction was developed for automated indexing in IR, and while it is sometimes applied to IR tasks (Nallapati et al., 2004), it has been largely explored as a means of text summarization (Bourigault, 1992; Frantzi, 1997; Justeson and Katz, 1995; Turney, 1999).

### 4.5.7 Statistical methods

There are many statistical word association measures that differ with respect to their mathematical rigour and the convenience with which they can be applied. Only a few are applied with regularity. The best-known are perhaps the information-theoretic notion of mutual information (MI) (Church and Hanks, 1990), the t-score measure (Church et al., 1991), and the log-likelihood ratio (Dunning, 1993). MI and the log likelihood ratio are discussed in more detail below. Other measures are either not well suited to IR, or inferior to the log likelihood ratio as approximations to Fisher's exact test (Evert, 2005).[13]

These measures are described with reference to the contingency table with four cells shown in Table 4.11. The cells show the marginals, for example the observed frequency of co-occurrence of words $x$ and $y$ ($O_{xy}$) and the frequencies of $x$ and $y$ co-occurring with a word other than $y$ and $x$ respectively ($O_{x\neg y}$ and $O_{\neg xy}$). The expected values, e.g. $E_{xz}$, of internal cells are calculated from the marginals. (see Section 4.4). These measures include:

- **The Dice coefficient:** measures the proportion of independent occurrences of words $x$ and $y$ that are also co-occurrences of $x$ and $y$ in combination. The Dice coefficient has a long history in automatic language analysis (Croft et al., 2010) and may be suitable for IR because it tends to highlight rigid multiword expressions (Evert, 2005) such as proper nouns. However, it is not very comprehensive in the linguistic phenomena that it identifies.
- **T-score:** a common measure of word association used to identify high frequency collocations that most distinguish between two words with similar meanings (Stubbs, 1995). The t-score assumes a normal distribution over the frequency of terms in a vocabulary. This assumes very few rare terms but in fact about 20-30% of tokens in a moderate-sized sample of English newswire are 'rare' with a frequency of less than one in 50,000 (Dunning, 1993). There is an even greater incidence of rare word co-occurrences. As a result, the t-score is not well suited to IR because the most frequent collocations are not necessarily the most informative, especially if they contain very high frequency words.
- **Z-score:** is similar to the t-score, but is characterized by a strong bias towards collocations containing low frequency words (Evert, 2005). It compares the observed and expected frequencies of word co-occurrence and evaluates these with

[13] Fisher's exact test is a rigorous mathematical test of word association.

respect to the standard deviation (computed using $E$, the expected frequency of word co-occurrence) instead of the sample standard deviation (computed using $O$, the observed frequency of word co-occurrence). Because it also assumes a normal distribution over term probabilities, the approximations used to calculate the z-score are inaccurate if any expected frequencies $E$ are small.

- **Chi-squared statistic**: is a generalization of the z-score that adds the squared z-scores for all cells in a contingency table. Like z-scores, the chi-squared statistic has a low-frequency bias. The log-likelihood ratio is more appropriate as a measure of collocation than the $\chi^2$ test because it more closely approximates the $\chi^2$ distribution for very infrequent word combinations (Manning and Schütze, 1999).[14]

#### 4.5.7.1 Mutual information

There are many variants of mutual information, all of which are measures of *effect size* (see Section 4.4). Pointwise mutual information (PMI) is perhaps the most well-known and intuitive. PMI is a measure of how much information an occurrence of $x$ provides about occurrences of $y$, and vice versa, for a particular definition of co-occurrence (e.g. text window, text unit, syntactic relation). It compares $O_{xy}$, the observed co-occurrence of words $x$ and $y$, and $E_{xy}$, the expected co-occurrence of $x$ and $y$ under the null hypothesis and is defined by:

$$PMI = log_2 \left( \frac{p(x,y)}{p(x)p(y)} \right)$$

where $p(x), p(y)$ and $p(x,y)$ are maximum likelihood estimates normalized by $N = O_{zz}$, the size of the corpus. If $x$ and $y$ are associated, then the joint probability $p(x,y)$ will be much larger than $p(x)p(y)$, the probability of observing $x$ and $y$ together by chance. Consequently the PMI between $x$ and $y$ will be much greater than one. If $x$ and $y$ are independent and occur together only by chance, then $p(x,y) \approx p(x)p(y)$, and $PMI = 0$.

PMI favours low frequency and high attraction collocations. For example, when applied to contiguous bigrams, it tends to identify proper names and noun compounds with high accuracy (Evert, 2005). Unfortunately, like the t-score and z-score, PMI assumes a normal distribution over term frequencies. This means that the estimates based on frequency counts substantially overestimate shared information for rare col-

[14] The chi-squared distribution is a better fit to language data than the normal distribution because it has a longer tail on the right than the left.

| Term | PMI | $C_{w1}$ | $C_{w2}$ | $C_{w1w2}$ |
|---|---|---|---|---|
| court justice | 7.89 | 91456 | 27266 | 2338 |
| cases court | 6.07 | 130837 | 91456 | 3177 |
| heard court | 5.69 | 20550 | 91456 | 383 |
| cases heard | 5.29 | 130837 | 20550 | 414 |
| cases justice | 4.43 | 130837 | 27266 | 304 |
| international justice | 4.24 | 174312 | 27266 | 354 |
| world international | 3.97 | 177723 | 174312 | 1911 |
| heard justice | 3.66 | 20550 | 27266 | 28 |
| international court | 3.43 | 174312 | 91456 | 678 |
| heard world | 2.69 | 20550 | 177723 | 93 |

(a) PMI table

| Term | MI | $C_{w1}$ | $C_{w2}$ | $C_{w1w2}$ |
|---|---|---|---|---|
| cases court | 13371.11 | 130837 | 91456 | 3177 |
| court justice | 12789.78 | 91456 | 27266 | 2338 |
| world international | 5253.63 | 177723 | 174312 | 1911 |
| court international | 1611.80 | 91456 | 174312 | 678 |
| cases heard | 1516.85 | 130837 | 20550 | 414 |
| heard court | 1510.61 | 20550 | 91456 | 383 |
| international justice | 1039.93 | 174312 | 27266 | 354 |
| cases justice | 933.97 | 130837 | 27266 | 304 |
| types cases | 371.83 | 107232 | 130837 | 248 |
| cases international | 313.49 | 130837 | 174312 | 278 |

(b) MI table

Figure 4.20: The top ten terms for query, "*What types of cases were heard by the World Court (International Court of Justice)?*", ranked by their (a) PMI score, and (b) MI score determined from word counts in the Robust04 collection. Component frequencies used in calculations are also shown.

locations. PMI does not consider the amount of evidence provided by co-occurrence data, so it assigns a high score when $O$ exceeds $E$ by a large amount, even if only one co-occurrence is observed and $E < 1$ (Evert, 2005). This bias is particularly problematic when calculations are based on data from large corpora. Corpus size has a substantial influence on maximum likelihood term probability estimates (Krishnamurthy, 2005; Stubbs, 1995).

Several modifications of PMI have been proposed to compensate for this low frequency bias. Mutual information (MI) weights the expected value of PMI over all possible instances of $x$ and $y$ by $p(x,y)$. This favours collocations with high-frequency words, instead of low-frequency words. It is defined as:

$$MI_{restricted} = p(x,y)PMI$$

Tables 4.20 (a) and (b) show the top ten terms for query $q$, "*What types of cases*

*were heard by the World Court (International Court of Justice)?*", as determined by PMI and MI respectively using counts from the Robust04 collection for all nterms with two words (see Section 4.5.2). Terms containing words with fewer than four occurrences in the retrieval collection were eliminated from consideration. It is clear that the ranking over terms can differ substantially for alternative mutual information measures.

A problem for both PMI and MI is that they are defined between two words and there is no generally accepted multivariate extension (Williams and Beer, 2010), although several measures are used in practice for language processing (Van de Cruys, 2011). For example, multivariate mutual information measures such as *interaction information* (McGill, 1954) can take positive or negative values that are difficult to interpret.

#### 4.5.7.2 Log likelihood ratio

The log likelihood ratio is a well-established measure for the identification and ranking of collocations that closely approximates Fisher's exact test (Evert, 2005). For a combination of two words $x$ and $y$, the log likelihood ratio ($log\lambda$) represents the deviation between the number of observed instances in which $y$ follows $x$ and the number of expected instances if $x$ and $y$ are independent. It is based on a ratio of the maximum value of the likelihood of co-occurrence for $x$ and $y$, for the subspace of all observations that are consistent with the hypothesis that $x$ and $y$ are dependent ($H_2$), to the maximum value of the likelihood for all observations (the hypothesis that $x$ and $y$ are independent, $H_1$):

$$log\lambda = log\frac{L(H_2)}{L(H_1)}$$

The log likelihood ratio for two words is computed using all observations $o_i \in O$ recorded in a contingency table as follows:

$$log\lambda = 2 \cdot \sum_{o_i \in O} o_i \cdot PMI_{o_i}$$

$$log\lambda = 2 \cdot (O_{xy} \cdot log\frac{O_{xy}}{E_{xy}} + O_{x\neg y} \cdot log\frac{O_{x\neg y}}{E_{x\neg y}} + O_{\neg xy} \cdot log\frac{O_{\neg xy}}{E_{\neg xy}} + O_{\neg x\neg y} \cdot log\frac{O_{\neg x\neg y}}{E_{\neg x\neg y}})$$

This measure has an advantage over MI because it is suitable for comparison of frequent and rare collocations, and works well for large or small text samples. It can

| Term | log $\lambda$ | $C_{w1}$ | $C_{w2}$ | $C_{w3}$ | $C_{w1w2}$ | $C_{w1w3}$ | $C_{w2w3}$ | $C_{w1w2w3}$ |
|---|---|---|---|---|---|---|---|---|
| cases court justice | 43232 | 130837 | 91456 | 27266 | 3177 | 304 | 2338 | 12 |
| cases heard court | 25028 | 130837 | 20550 | 91456 | 414 | 3177 | 383 | 27 |
| international court justice | 24374 | 174312 | 91456 | 27266 | 678 | 354 | 2338 | 249 |
| heard court justice | 23539 | 20550 | 91456 | 27266 | 383 | 28 | 2338 | 5 |
| cases international court | 22912 | 130837 | 174312 | 91456 | 278 | 3177 | 678 | 11 |
| world court justice | 21426 | 177723 | 91456 | 27266 | 168 | 83 | 2338 | 2 |
| types court justice | 21199 | 107232 | 91456 | 27266 | 35 | 22 | 2338 | 0 |
| court justice | 21187 | 91456 | 27266 | | 2338 | | | |
| types cases court | 21038 | 107232 | 130837 | 91456 | 248 | 35 | 3177 | 0 |
| cases world court | 20989 | 130837 | 177723 | 91456 | 257 | 3177 | 168 | 2 |

Figure 4.21: The top ten terms for query, "*What types of cases were heard by the World Court (International Court of Justice)?*", ranked by their log likelihood ratio score with counts determined from the Robust04 collection. Component frequencies used in the calculation are also shown.

also be applied equally well to the multinomial and the binomial cases. However, as with all multivariate measures of significant association, the multivariate log likelihood ratio is not well understood due to complex interaction effects. The number of methods by which it can be calculated increases as the number of words increases (McInnes, 2004).[15] The method used to calculate the log likelihood ratio for three words in this dissertation has the same base formula as for two words, namely (McInnes, 2004):

$$log\lambda = 2 \cdot \sum_{o_i \in O} o_i \cdot PMI_{o_i}$$

except the denominator for the calculation of $E_{wxy}$ in the appropriate expansion is $O^2_{zzz}$, corresponding to an increase in the degrees of freedom:

$$E_{wxy} = \frac{O_{wzz} \cdot O_{zxz} \cdot O_{zzy}}{O^2_{zzz}}$$

Table 4.21 shows the top ten nterms with two or three words (see Section 4.5.2) for query $q$ as ranked by the log likelihood ratio with counts determined from the Robust04 collection.

## 4.6 Evaluation of semantic representation

Word association methods described in the previous Section may be used in IR to identify query terms. These terms aim to describe the semantics of queries better than

[15]The expected values for three words $w_1$, $w_2$ and $w_3$ can be based on four models: 1) word independence; 2) the probability that $w_1$ and $w_2$ are dependent and independent of $w_3$; 3) the probability that $w_2$ and $w_3$ are dependent and independent of $w_1$; and 4) the probability that $w_1$ and $w_3$ are dependent and independent of $w_2$ (McInnes, 2004).

individual words and thereby improve retrieval effectiveness. This Section evaluates the degree to which terms identified by different word association methods represent the semantics of requests. Specifically, the precision, recall and F1 score[16] with which automatically detected terms match user-nominated terms is taken as a measure of semantic representation.

### 4.6.1 Gold standard terms

Users seem to accurately interpret language semantics (Raghavan and Allan, 2007) and their judgements have been applied to improve IR effectiveness in related work (Allan and Raghavan, 2002; Kumaran and Allan, 2006, 2008). However, systematic identification of a gold standard for request semantics is difficult. It is not always practical to prompt users either to grade the plausibility of term candidates, or identify semantically relevant terms from a set of candidates (Evert and Krenn, 2001; Lapata et al., 1999). The TREC description queries used in this evaluation are an example: 500 queries produced over 50,000 candidate terms with 1-3 words.

An alternative approach to the acquisition of gold standard terms for semantic representation adapts human-authored resources for the identification of important word associations. Unfortunately, these resources are not flexible. Manually compiled knowledge resources, such as thesauri (Bordag, 2007) and WordNet (Schone and Jurafsky, 2001), focus on particular word relationships and may only be applicable to a restricted set of word pairs. Some resources include multi word units (MWUs) (Schone and Jurafsky, 2001) but typically only record words in formal relations, such as synonyms and hypernyms. It seems unlikely that informative query terms are restricted to particular word association types and word pairs.

For these reasons, user *nominated* terms are chosen as a gold standard. Three native speakers were prompted with the description topics for the Robust04, GOV2 and WT10G retrieval collections (Section 2.4). For each topic, subjects were asked to nominate all terms with up to four words that they thought would represent the meaning of a query. Terms were required to contain only words from the original topic, excluding stopwords. It is assumed that the there is some semantic relation between the words in a term. Full task instructions are reported in Appendix C.

Subjects differed greatly in the average number of terms identified per topic: 13.4, 4.9 and 2.0 respectively. The word order nominated by different users was inconsis-

[16] F1 score is the harmonic mean of precision and recall.

tent for more than 10% of terms, but analysis suggested that these variations were not significant. In sample test-retest conditions with 50 description topics, users nominated alternative word orders for the same terms compared to their initial judgments. Word orders were therefore normalized to their surface order in the description topic, and a final list of gold standard terms was generated to include all those normalized terms nominated by at least two subjects. Terms nominated by only one subject were discarded.

A manual review indicated that the nominated terms were of reasonable quality even though in general the reliability of human judgments can be compromised by confounding factors including disinterest, distraction, judgment of grammaticality instead of content, the influence of context, and individual variation. In part, this is because queries are provided by individual users, so agreement between users may not be pertinent to the intended semantics for a specific user. Perhaps by consequence, a limitation to the judgments of three subjects is fairly common for IR (Allan et al., 2013; Dori-Hacohen and Allan, 2013; Kong and Allan, 2013).

Given these considerations, the methodology here is sufficient to investigate a relationship between word association and semantics for IR. However, term quality could be further validated by asking each subject to rate every nominated term. There is evidence that even if agreement on the quality of nominated keyphrases is low, there is significant, and sometimes strong, agreement on keyphrase ranking (Jones and Paynter, 2001). Gold standard term selection could be based on term rating or rank.

### 4.6.2 Methodology

The ability of terms described by word association methods to identify user nominated terms is measured by precision, recall and F1 score. Word association methods tested are those described in Section 4.5 plus unigrams (Uni). Word association methods are evaluated independently and in all possible combinations of 1 to 9 methods. Method combinations use the set of all terms identified by the individual methods being combined, and apply no term weighting.

### 4.6.3 Results

Results for individual methods are reported in Table 4.1. Overall, it is apparent that word association methods have different strengths. Noun phrases and trigrams are the most precise methods for selection of user nominated terms, followed by bigrams. The

precision of trigrams and bigrams suggests the reliability of syntagmatic relations for communication of semantic relatedness. The performance of trigrams is somewhat surprising, but especially for Robust04 may reflect a tendency of users to select terms that summarize a query.

With respect to noun phrases, precision might be attributed to exclusion of many undesirable terms from the set of all possible word combinations. Noun phrases define all and only the word combinations that fill a certain role in language (Section 3.1.1). It may also reflect a strong alignment of syntactic and semantic compositionality. However, this reasoning is speculative. A phrase that is part of a grammatically correct sentence can overlap with a word combination that is determined by requirements and restrictions on word combinability irrespective of grammatical considerations. Syntactic relations, such as noun phrases, can be lexical even if syntactic relations *in general* are not lexical, or do not account for *all* lexical relations.

Catenae, as well as terms identified by the log likelihood ratio, are less precise but demonstrate better recall. Catenae consistently achieve significantly better recall than all other methods tested (excluding nterms that exhaustively describe all possible combinations of 1-3 words). This may be because dependency theory directly captures semantic information (Section 3.2). Catenae also form the largest set of terms identified by an individual word association method and this can contribute to recall. Log likelihood ratio terms also achieve strong recall given that they describe a much smaller number of terms. This is reflected in their F1 score. For two of three collections, log likelihood ratio terms achieve significantly better F1 scores than all other methods tested. This might be expected based on the ability of statistical word association methods to detect lexical word relations, as discussed in Section 4.4.1.

Combinations of word association methods were also explored. Results for the best combinations with respect to precision, recall and F1 are reported in Table 4.2. As expected, combinations of many methods deliver the best recall but still fall substantially short of optimal performance. A reason for this is illustrated in Table 4.3, which shows that there is substantial overlap in the stoplisted terms identified by word association methods for a sample query. More importantly, the F1 scores for method combination are not noticeably better than they are for classification using individual methods. This is due to a decrease in precision when methods are combined. In practice, many IR techniques combine evidence from several word association methods (e.g. noun phrases, governor-dependent relations, ngrams, and so on), but such combinations do not appear to substantially improve semantic representation of queries.

| Method | Rob04 | | | GOV2 | | | WT10G | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 |
| Nterm | 0.0987 | 0.9600 | 0.1652 | 0.1832 | 0.9341 | 0.2810 | 0.2481 | 0.9760 | 0.3608 |
| Uni | 0.0295 | 0.0404 | 0.0324 | 0.0758 | 0.0908 | 0.0797 | 0.1518 | 0.1549 | 0.1490 |
| Seq2 | 0.1654‡ | 0.2404 | 0.1793 | 0.2954‡ | 0.2818 | 0.2700 | 0.3681† | 0.2792 | 0.3033 |
| Seq3 | **0.1870**‡ | 0.2083 | 0.1785 | 0.3183‡ | 0.2103 | 0.2340 | 0.3309† | 0.2069 | 0.2355 |
| NP | 0.1571‡ | 0.1381 | 0.1335 | **0.3510**‡ | 0.2328 | 0.2605 | **0.3733**‡ | 0.2140 | 0.2573 |
| BPhr | 0.1121 | 0.2784 | 0.1494 | 0.2148 | 0.3712 | 0.2570 | 0.3046 | 0.4006 | 0.3137 |
| GDep | 0.1492‡ | 0.2140 | 0.1616 | 0.2785‡ | 0.2690 | 0.2565 | 0.3329† | 0.2488 | 0.2706 |
| Cat | 0.1033 | **0.4329** | 0.1589 | 0.1874 | **0.5252** | 0.2630 | 0.2452 | **0.5574** | 0.3262 |
| MI | 0.1206 | 0.1770 | 0.1311 | 0.2450 | 0.2698 | 0.2336 | 0.2949 | 0.2922 | 0.2741 |
| LogL | 0.1674‡ | 0.3311 | **0.2124**‡ | 0.2981‡ | 0.4450 | **0.3336**‡ | 0.3382† | 0.4035 | **0.3376** |

Table 4.1: Precision (P), recall (R) and F1 score calculated over all queries for three TREC collections, with user nominated terms as binary targets (presence or absence). † shows significant ($p < .05$) and ‡ highly significant ($p < .01$) improvement compared to nterms as determined by a t-test.

| | Best | Method combination | P | R | F1 |
|---|---|---|---|---|---|
| Rob04 | Precision | Seq3 | **0.1870** | 0.2083 | 0.1785 |
| | Recall | Seq2 Seq3 NP BPhr Cat MI LogL | 0.1058 | **0.6622** | 0.1740 |
| | F1 | Seq2 Seq3 | 0.1801 | 0.4327 | **0.2345** |
| GOV2 | Precision | NP | **0.3510** | 0.2328 | 0.2605 |
| | Recall | Seq2 Seq3 NP BPhr Cat MI LogL | 0.1887 | **0.7592** | 0.2856 |
| | F1 | Seq2 NP LogL | 0.2667 | 0.6234 | **0.3533** |
| WT10G | Precision | Seq2 Seq3 | **0.3823** | 0.4608 | 0.3837 |
| | Recall | Uni Seq2 Seq3 Cat MI LogL | 0.2552 | **0.8060** | 0.3648 |
| | F1 | Seq3 NP LogL | 0.3206 | 0.6500 | **0.3983** |

Table 4.2: Precision (P), recall (R) and F1 score calculated over all queries for three TREC collections, with user nominated terms as binary targets (presence or absence).

| All methods | Sequential bigram | Sequential ngram | Noun phrase | Governor-dependent | Bounded phrase | Mutual information | Log likelihood |
|---|---|---|---|---|---|---|---|
| cases | | | cases | | cases | | |
| heard | | | | | heard | | |
| int'l | | | | | int'l | | |
| justice | | | | | justice | | |
| type | | | type | | type | | |
| court | | | | | court | | |
| world | | | | | world | | |
| cases court | | | | | | cases court | |
| cases heard | cases heard | cases heard | | | | cases heard | |
| | | | | court court | | | |
| court int'l | court int'l | court int'l | | court int'l | | court int'l | |
| court justice | court justice | court justice | | court justice | court justice | court justice | |
| heard type | | | | heard type | | | |
| heard court | | | | heard court | | heard court | |
| heard world | heard world | heard world | | | | | |
| int'l court | int'l court | int'l court | int'l court | | int'l court | | |
| int'l justice | | | | | | int'l justice | |
| type cases | type cases | type cases | type cases | type cases | | | |
| world court | world court | world court | world court | world court | world court | | |
| world int'l | | | | | | world int'l | |
| cases court justice | | | | | | | cases court justice |
| cases heard court | | | | | | | cases heard court |
| cases heard world | | cases heard world | | | | | |
| cases int'l court | | | | | | | cases int'l court |
| court int'l court | | court int'l court | | | | | |
| heard court justice | | | | | | | heard court justice |
| heard world court | | heard world court | | | | | |
| int'l court justice | | int'l court justice | int'l court justice | | int'l court justice | | int'l court justice |
| type cases heard | | type cases heard | | | | | |
| type court justice | | | | | | | type court justice |
| world court int'l | | world court int'l | | | | | |
| world court justice | | | | | | | world court justice |

Table 4.3: Terms selected by different word association methods for the query, '*What types of cases were heard by the World Court (International Court of Justice)?*' (Robust04 #376)

Nevertheless, combinations of ngrams achieve high F1 scores, as do noun phrases and log likelihood terms as individual method sets. The subsequent impact of method combination on search effectiveness is explored further in the next Chapter.

## 4.7 Conclusion

It is easy to comprehend the appeal of an IR system that 'understands' the semantics of text and can use this knowledge to select terms. From a theoretical standpoint, a system that uses statistical relations most flexibly identifies a broad range of semantic word relations. Statistical methods are particularly suitable for the identification of lexical relations that are not consistently identified by syntax or word sequences but always appear in recurrent relationships in text (Giegerich, 2006). An evaluation of the ability of word association methods to identify user nominated terms also shows that terms identified by the log likelihood ratio consistently achieve the highest F1 score of all individual methods tested. They also contribute to the best performing combinations of methods.

Noun phrases are the most precise method for identification of user-nominated terms for two of three collections. However, they have low recall. On the one hand this may be due to differences in the number of terms described by various methods (more user nominated terms are likely to be identified by a large set of term candidates). However, it is worth observing that from a theoretical standpoint, phrase structure theory has two drawbacks. First, grammatical categories may be irrelevant to word relations when they are used as determinate features, as they are here. Second, phrase structure grammars assume that semantic interpretation occurs below the surface. This makes it difficult for word associations that work with surface phrase structure representations to reliably identify certain semantic content.

Dependency theory is arguably better suited to the detection of semantic relations than surface phrase structure theory because it better accounts for variation in surface word order given the same semantic intent. Along with lexicalism, dependency theory is *functionalist*, driven by the idea that language form is motivated by underlying semantic function (Searle, 1972). However, while catenae achieve strong recall of user nominated terms, governor-dependent relations demonstrate unremarkable performance. The flexibility of catenae to describe terms of varying length appears to be a factor in their relative ability to represent the semantics of requests. Noun phrases and log likelihood terms display similar flexibility.

Syntagmatic word order is a reasonable heuristic for the detection of semantic relations. A substantial number of syntagmatic word combinations are coincidental, but a large proportion describe a semantic relation. In addition, syntagmatic relations frequently underpin statistical methods and contribute to their success. For these reasons, we might expect a large number of syntagmatic units to be semantically meaningful terms for IR. Experimental results confirm that they are reasonably precise and contribute to a strong F1 score in method combinations.

Overall, both theoretical and empirical evidence suggest there is no strong relationship between word association methods and the semantic terms that are optimally extracted from queries. The highest F1 measure for the detection of user-nominated terms by any individual method or combination of methods did not exceed 0.4. It appears that each class of word association methods (syntagmatic, syntactic, statistical) imperfectly interprets the semantics of language. However, empirical evidence is required to identify whether these shortcomings are important in the discrimination of relevant documents.

In the next Chapter, I address three questions concerning the ability of terms to discriminate between relevant and non-relevant documents in a retrieval collection: How do terms selected by individual word association methods compare with each other? What contributions do word association methods make to the identification of informative terms when they are combined in a term selection model? Finally, what is the effect of term combination on retrieval effectiveness, and how is this affected by the application of multiple word association methods?

# 5

# Term Discrimination

The selection of informative query terms depends on two criteria, as proposed in Section 1.1.1: the degree to which a structure used to identify terms captures request semantics, and the ability of terms to discriminate relevant documents.

Empirical evaluation in this Chapter explores the hypothesis that word association methods identify discriminative terms. Evaluation proceeds in three stages. In the first stage, word association methods are evaluated with respect to their ability to identify terms that improve IR effectiveness compared to baseline queries. This is measured with standard precision, recall and F1 scores. Methods are tested individually and in combination. Term combinations are also considered in order to determine the degree to which discrimination is a function of individual terms and term sets. Some terms are only discriminative as part a set (Barker and Cornacchia, 2000; Jones and Paynter, 2003).

The second stage explores the relationship between word association and discrimination in more detail, applying a linear regression framework that is more sensitive to the scale of changes in IR effectiveness. Once again, both individual methods and method combinations are considered. Since suboptimal alignment was observed between methods of word association and gold standard semantic terms in Chapter 4, this analysis also explores an upper bound on the expected association between semantic representation and discrimination of relevant documents.

Finally, interaction effects with term and method combinations are visualised to explore the balance between the information provided by multiple word association methods and the overhead of language processing.

For all evaluations, terms are extracted from description topics for Robust04, WT10G and GOV2, and results are computed over all topics. Word association methods ex-

plored are unigrams (Uni), and methods described in Section 4.5: bigrams (Seq2), trigrams (Seq3), nterms (Nterm), noun phrases (NPs), governor-dependent pairs (GDep), catenae (Cat), bounded phrases (BPhr), mutual information terms (MI) and log likelihood ratio terms (LogL).

## 5.1 Evaluation of term discrimination

This Section reports two straightforward evaluations of the sensitivity and specificity of word association methods for the description of terms that discriminate relevant and non-relevant documents. The first evaluation reports the precision, recall and F1 scores for classification of terms that improve IR effectiveness. The second evaluation provides statistics on the improvement in NDCG observed for individual terms and term combinations. Two constraints limit combinations to a reasonable number: 1) only combinations of 2 or 3 terms are considered, and 2) combined terms must be identified by a single word association method. For each word association method, the percentage of terms that are discriminative in a collection, and the average, maximum and total improvement they deliver across all queries, are reported.

### 5.1.1 Methodology

Performance metrics measure the ability of terms to improve IR effectiveness compared to query likelihood when they are interpolated with the query likelihood query. NDCG is chosen as the metric of IR effectiveness from a set of acceptable alternatives {MAP, P@10, P@100, R-Prec, NDCG, NDCG15} (see Section 2.5). This selection is based on a series of logistic regressions in which term IR metric scores are used to predict user nominated terms. The intuition is that IR aims to match the semantics of queries and documents, so discriminative terms should reflect query semantics. This intuition is not proven to be true, and in fact in Section 5.2.3.1 it will be shown that there is a very limited relationship between semantics and discrimination. However, for the purpose of choosing between widely applied IR metrics the assumption is acceptable. NDCG was chosen because it had the strongest relationship with semantic terms by a narrow margin (see Appendix G).

A limiting factor when using any IR metric for evaluation of term quality is reliance on a pre-selected IR model and query reformulation template. The effectiveness of a term cannot be established unambiguously because its ability to discriminate relevant

| | |
|---|---|
| A | #weight(<br>0.85 #combine(outcome pro independence protesters tibet arrested chinese authorities)<br>0.15 #1(protesters tibet)) |
| B | #weight(<br>0.85 #combine(outcome pro independence protesters tibet arrested chinese authorities)<br>0.15 #uw8(protesters tibet)) |

Figure 5.1: A query using (A) an ordered window operator (#1), and (B) an unordered window operator (#uw8). In all other respects the queries are the same.

| Term | MAP | |
|---|---|---|
| | Model A | Model B |
| protesters tibet | 0.6142 | 0.4818 |
| pro independence | 0.4886 | 0.4858 |

Table 5.1: Evaluation of term selection using IR metrics is dependent on the IR model.

documents can vary with different choices of IR model and query reformulation. For example, consider the query "*What has been the outcome for the pro-independence protesters in Tibet who were arrested by Chinese authorities?*" (Robust04 #612). Figure 5.1 shows two possible reformulations of this query in Indri query language using the term $t_1$='*protesters Tibet*'. Model A applies an ordered window operator to $t_1$, and model B applies an unordered window operator.

Consider that we want to compare the discriminative ability of $t_1$ with a second term, $t_2$='*pro independence*'. If a query using $t_1$ retrieves more relevant documents than a query using $t_2$, then it may seem trivial that $t_1$ is more discriminative than $t_2$. However, depending on which query reformulation strategy is applied, $t_1$ may be more or *less* discriminative than $t_2$. Table 5.1 shows that $t_1$ is more discriminative than $t_2$ according to query MAP (see Section 2.5) when using model A and less discriminative than $t_2$ when using model B.

This example illustrates that when an IR metric is used to identify discriminative terms, it can be difficult to know whether gains in IR performance are attributable to terms themselves, term selection methods (e.g. features of word association), term combinations or the ranking behaviour of an IR system.

A comprehensive evaluation of alternative IR models is beyond the scope of this thesis. Instead, an IR model and query reformulation framework is chosen that is consistent with experimental and highly competitive models in Chapters 6 and 7. The reformulation employs a robust, highly effective linear feature model based on the sequential dependence (SD) model (see Section 2.2.4). This reformulation inserts a single term, or a combination of several terms, in the ordered and unordered windows

(cliques 2 and 3) of the SD model in place of bigrams. In Indri query language, for the query '*new york city*' and the single term '*new york*', this takes the form:

**Baseline:** #combine(new york city)

**Reformulation:**
#weight(
0.85 #combine(new york city)
0.1 #1(new york)
0.05 #uw8(new york))

For a combination of two terms *x*='*new york*' and *y*='*york city*', and the same query, the reformulation takes the form:

**Reformulation:**
#weight(
0.85 #combine(new york city)
0.1 #combine(#1(new york) #1(york city))
0.05 #combine(#uw8(new york) #uw8(york city)))

The first element in each reformulation is a query likelihood query. Terms are evaluated by comparing the effectiveness of reformulated queries to the baseline query likelihood. All queries are implemented using version 4.12 of the Indri search engine.[1]

### 5.1.2 Results

Table 5.2 shows the ability of word association methods to identify discriminative terms. Excluding nterms, catenae achieve the highest recall and F1 scores on all collections while the most precise method varies for different collections. Nterms achieve a higher F1 than catenae but this is due to perfect recall; nterms identify around 40 to 150 terms per query compared to an average of 7 terms per query for word association methods.

Table 5.3 shows that small improvements in F1 are possible with method combination compared to individual methods. The methodology used in Section 4.6.2 is repeated for consistency. Namely, method combination uses the set of all terms identified by the individual methods being combined, and applies no term weighting. The most desirable combination is remarkably consistent, and essentially composes all of the methods except noun phrases and bounded phrases (governor-dependent pairs are a subset of catenae). This result is considered further in Section 5.3.

Despite improvements in F1 with method combination, the highest F1 is still less than 0.4, similar to the F1 scores in Chapter 4 for prediction of user-nominated terms.

[1] `http://www.lemurproject.org/`

| Method | Rob04 | | | GOV2 | | | WT10G | | |
|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 |
| Nterm | 0.3087 | 1.0000 | 0.4393 | 0.3054 | 1.0000 | 0.4407 | 0.3246 | 1.0000 | 0.4657 |
| Uni | 0.2313 | 0.1626 | 0.1637 | 0.2911 | 0.2461 | 0.2430 | **0.3100** | 0.3220 | 0.2754 |
| Seq2 | 0.2584 | 0.1474 | 0.1647 | 0.3125 | 0.1785 | 0.2124 | 0.2837 | 0.1715 | 0.1952 |
| Seq3 | **0.2974** | 0.1061 | 0.1398 | 0.3237 | 0.1121 | 0.1579 | 0.2894 | 0.1049 | 0.1414 |
| NP | 0.2550 | 0.1003 | 0.1235 | 0.3388 | 0.1470 | 0.1831 | 0.3886 | 0.2293 | 0.2427 |
| BPhr | 0.2822 | 0.2573 | 0.2248 | **0.3521** | 0.3710 | 0.3199 | 0.3690 | 0.4064 | 0.3374 |
| GDep | 0.2367 | 0.1364 | 0.1515 | 0.2908 | 0.1596 | 0.1938 | 0.2370 | 0.1547 | 0.1715 |
| Cat | 0.2582 | **0.3962** | **0.2697** | 0.3010 | **0.5276** | **0.3549** | 0.2933 | **0.5739** | **0.3548** |
| MI | 0.2207 | 0.1483 | 0.1479 | 0.3061 | 0.1922 | 0.2124 | 0.2523 | 0.1989 | 0.1977 |
| LogL | 0.2718 | 0.2202 | 0.1983 | 0.3266 | 0.2905 | 0.2758 | 0.2872 | 0.2538 | 0.2339 |

Table 5.2: Precision (P), recall (R) and F1 score for the classification of terms that improve NDCG, calculated over all queries for three TREC collections. Catenae consistently deliver the highest F1 of all methods tested except nterms.

| | Best | Method combination | P | R | F1 |
|---|---|---|---|---|---|
| Rob04 | Precision | Seq3 | **0.2974** | 0.1061 | 0.1398 |
| | Recall | Uni Seq2 Seq3 Cat MI LogL | 0.2558 | **0.5890** | 0.3181 |
| | F1 | Uni Seq2 Seq3 Cat MI LogL | 0.2558 | 0.5890 | **0.3181** |
| GOV2 | Precision | BPhr | **0.3521** | 0.3710 | 0.3199 |
| | Recall | Uni Seq2 Seq3 Cat MI LogL | 0.2929 | **0.7153** | 0.3895 |
| | F1 | Uni Seq2 Seq3 Cat LogL | 0.3014 | 0.6981 | **0.3939** |
| WT10G | Precision | NP | **0.3886** | 0.2293 | 0.2427 |
| | Recall | Uni Seq2 Seq3 Cat MI LogL | 0.2963 | **0.7547** | 0.3963 |
| | F1 | Uni Seq2 Seq3 Cat MI LogL | 0.2963 | 0.7547 | **0.3963** |

Table 5.3: Precision (P), recall (R) and F1 score calculated over all queries for three TREC collections, with terms that improve NDCG as binary targets (presence or absence).

In contrast to results in Chapter 4, the highest F1 score was achieved with nterms for all three collections (rather than a specific word association method, see Table 4.1). Based on these results, it might appear that a successful approach to document retrieval simply assumes dependence between all words in a request and ignores linguistic information. Word association methods are not noticeably better for the detection of discriminative terms than random combination of words in a sentence. However, straightforward application of a full word dependence assumption is no more effective in practice than a simple approach using bigrams (Metzler and Croft, 2005). To gain further insight into how this is possible, the scale of improvements must be considered.

Table 5.4 shows that nterms have the lowest average NDCG improvements per term of all the individual methods explored.[2] In contrast, noun phrases demonstrate the strongest performance for both average and maximum NDCG improvement for individual terms and term combinations. Noun phrases and catenae are both further differentiated by their ability to identify terms that work well collectively.[3] I speculate that noun phrases are distinguished in this way because they represent a fairly exclusive set of word combinations that fill a certain role in language. In contrast, other methods reflect more general principles of word association and identify a wider variety of word combinations. The dominance of noun phrases in this analysis indicates that selectivity can play a pivotal role in document discrimination.

Table 5.4 also illustrates that for all word association methods, around 70-80% of terms decrease effectiveness. There is little difference between methods in this respect, although bigrams, noun phrases and bounded phrases, and perhaps governor-dependent pairs, achieve slighter greater average improvements for each individual term.

For both individual words and word associations, substantially more terms improve NDCG scores on average when used in combinations than when used alone. This is because discriminative terms modulate the effect of terms that do not discriminate relevant documents and vice versa. It may also be due to the influence of TRIANGULATION effects. I define triangulation to be a condition that occurs when two terms share a common word and are used in a query together (see Section 8.2). For example, consider the topic '*Identify any efforts, proposed or undertaken, by world governments to seek reduction of Iraq's foreign debt*' (GOV2 #705). For this topic, the terms '*foreign debt*'

[2] Seq3 terms have a lower average NDCG improvement for GOV2.

[3] Based on average percent improvement in NDCG per term. Total gain in this Table must be interpreted with care as it is strongly influenced by the number of terms, or term combinations, in a set.

| | | Robust04 | | | | GOV2 | | | | WT10G | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Method | Terms Combined | % Terms Imprv | % Imprv NDCG Avg | Max | Total | % Terms Imprv | % Imprv NDCG Avg | Max | Total | % Terms Imprv | % Imprv NDCG Avg | Max | Total |
| Nterm | 1 | 29 | 3.7 | **85** | 38948 | 24 | 7.9 | **100** | 15144 | 29 | 8.5 | **100** | 9504 |
| Uni | 1 | 21 | 12.2 | 82 | 4626 | 26 | 10.7 | **100** | 2363 | 28 | 12.0 | 77 | 1713 |
| | 2 | 21 | 3.9 | 31 | 1508 | 27 | 3.2 | 40 | 719 | 31 | 4.3 | 38 | 687 |
| | 3 | 23 | 1.9 | 16 | 794 | 29 | 1.5 | 21 | 373 | 36 | 2.6 | 23 | 461 |
| Seq2 | 1 | 21 | 11.2 | **85** | 3842 | 24 | 12.1 | 79 | 2047 | 23 | 14.9 | **100** | 1448 |
| | 2 | 34 | 4.5 | 41 | 2534 | 34 | 4.7 | 42 | 1123 | **52** | 5.5 | 40 | 860 |
| | 3 | 45 | 2.9 | 25 | 2043 | 48 | 2.4 | 19 | 745 | 50 | 4.8 | 33 | 910 |
| Seq3 | 1 | **28** | 4.9 | 72 | 1925 | 28 | 7.2 | 85 | 1115 | 26 | 13.4 | **100** | 1087 |
| | 2 | **49** | 2.5 | 33 | 1711 | 47 | 3.0 | 23 | 741 | 50 | 6.0 | 50 | 897 |
| | 3 | **53** | 1.8 | 21 | 1223 | 51 | 2.3 | 13 | 560 | 51 | 3.7 | 22 | 500 |
| NPs | 1 | 25 | **12.3** | **85** | 1452 | 30 | **12.3** | 85 | 1452 | **32** | **15.5** | 84 | 1286 |
| | 2 | 34 | **8.6** | 60 | 4244 | **49** | 4.6 | **66** | 757 | 44 | **10.5** | **88** | 1578 |
| | 3 | 33 | **9.5** | **52** | 7930 | **66** | 4.9 | **48** | 897 | 40 | **10.7** | **84** | 1518 |
| BPhr | 1 | 24 | 12.1 | **85** | 7155 | **31** | 11.3 | **100** | 4004 | 31 | 14.2 | **100** | 2796 |
| | 2 | 28 | 4.6 | 41 | 3190 | 41 | 4.0 | 39 | 1861 | 40 | 5.7 | 41 | 1041 |
| | 3 | 36 | 4.4 | 32 | 10607 | 49 | 3.7 | 35 | 4149 | 40 | 5.4 | 37 | 2497 |
| GDep | 1 | 19 | 10.6 | **85** | 3498 | 21 | 12.0 | 79 | 1825 | 18 | 13.5 | **100** | 1082 |
| | 2 | 31 | 4.3 | 41 | 2233 | 33 | 4.9 | 39 | 1150 | 40 | 5.5 | 40 | 671 |
| | 3 | 42 | 2.6 | 23 | 1793 | 44 | 2.6 | 17 | 751 | 44 | 4.1 | 33 | 707 |
| Cat | 1 | 24 | 8.4 | 32 | 10512 | 25 | 9.6 | 100 | 5387 | 26 | 11.1 | **100** | 3835 |
| | 2 | 27 | 8.2 | **86** | 24992 | 34 | **6.6** | 82 | 9066 | 34 | 8.2 | 79 | 6512 |
| | 3 | 34 | 6.7 | 50 | 57823 | 40 | **5.9** | 78 | 15367 | 37 | 6.9 | **84** | 10196 |
| MI | 1 | 19 | 10.3 | 71 | 2927 | 21 | 11.3 | 79 | 1916 | 23 | 13.7 | **100** | 1634 |
| | 2 | 31 | 4.0 | 33 | 1836 | 32 | 4.3 | 40 | 1063 | 41 | 5.8 | 50 | 1182 |
| | 3 | 20 | 2.5 | 20 | 1568 | 46 | 2.7 | 27 | 933 | 46 | 4.6 | 33 | 1032 |
| LogL | 1 | 26 | 6.3 | 72 | 2745 | 29 | 8.3 | 85 | 2266 | 26 | 11.6 | **100** | 1836 |
| | 2 | 44 | 3.0 | 34 | 2245 | 46 | 3.5 | 42 | 1508 | 47 | 5.5 | 50 | 1509 |
| | 3 | 50 | 2.0 | 22 | 1696 | 54 | 2.5 | 28 | 1245 | **52** | 3.7 | 33 | 1136 |

Table 5.4: Improvement in NDCG for queries containing single terms (1), pairs of terms (2), and three-term combinations (3) compared to QL baseline. Best result for each metric given a combination (1, 2 or 3) shown in bold for each collection. Metrics: percentage of terms that improve NDCG (% Terms Imprv); average % improvement in NDCG (Avg); % improvement for the most effective term or term combination (Max); the sum of % improvements for all terms that improve NDCG (Total).

and '*reduction debt*' used alone may retrieve many irrelevant documents related to a local (non-Iraqi) national budget. Likewise, '*iraq foreign*' may retrieve documents about war in Iraq. However, these terms are more likely to retrieve documents about Iraqi debt, the focus of the information need, when used together to constrain document content.

Triangulation can result in robust term combinations, but the terms must be well chosen to avoid an overall reduction in search performance. Even the most effective term combination is usually not as effective as the single most discriminative term.[4] This highlights the need for careful term selection strategies to optimize IR performance. Queries that contain fewer, well chosen terms are likely to perform better than queries that postulate a distribution over many terms unless those terms are carefully selected. This finding validates the need for improved term selection and query reduction strategies.

Overall, it appears that word association methods may have some strengths for the prediction of discriminative terms, but since random combinations of words in nterms also perform adequately it is not clear that linguistic information is useful in IR. The possibility that word association methods can help to identify the semantics of requests is explored further in the next Section.

## 5.2 Evaluation of semantics and discrimination

This Section investigates the scale of changes in IR effectiveness when using user-nominated (semantic) terms and word association methods. Broadly, there are two possibilities. If language semantics is grounded in a model of the world, and has objective existence independent of speakers and language interpretation, then it is plausible that semantics and discrimination for IR effectiveness could be correlated. Semantics might not need to be considered separately in order to determine document relevance. A user generates a request with full knowledge of the model, including the information necessary to disambiguate meaning, and the semantics of a request is the popular interpretation of the request. This interpretation can be determined by reference to a large query log with click-through data, or word frequency statistics in a retrieval collection. This approach to semantics is often assumed by industrial search providers.

However, while an assumption of objective semantics is practical, it is not necessarily encountered in reality. The semantics of a request properly reflects the information

[4] Exceptions in this exploration were catenae for Robust04 and noun phrases for WT10G

need perceived by the user making the request (Klein and Rovatsos, 2011; Mizzaro, 1998), possibly with incomplete knowledge of alternative interpretations. As such, terms that users nominate to represent queries provide an ambiguous signal modulated by the influence of user conceptual models of meaning. In addition, native speaker intuitions about the ability of terms to distinguish relevant documents do not necessarily match actual term effectiveness. Users can fail to notice term ambiguity, and may struggle to discern terms that only discriminate between relevant and non-relevant documents when used as part of a set. Terms containing words that are not discriminative on their own, but realize effective queries when used together, can also be problematic.

When a request reflects the conceptual world of a user, there may be no way to align an idiosyncratic request semantics with objective data. Documents that are relevant according to a user are not necessarily relevant according to patterns of word use. Semantics and discrimination are not clearly correlated, and it is difficult to determine the extent to which semantic factors are important in IR.

Section 5.1 binned terms according to whether they improve or decrease NDCG, but this approach is insensitive to small changes, especially around boundaries between two bins. Ideally, such influences would be reduced as much as possible. The experiments in this Section investigate the relationship between semantics and discrimination with a focus on the scale of improvement, and provide evidence that semantics and discrimination are not necessarily related.

### 5.2.1 Methodology

Linear regression is used to identify word association methods and user nominated terms (predictors) that best explain, or predict, discriminative terms (targets) for unseen data, both individually and in the presence of interactions with other terms. These contributions are assessed by 'goodness of fit' measures that quantify the strength of association across all data (description topics). The relative contributions that word association methods make to predictions are also determined. Three regressions are used to predict the scale of change in NDCG compared to baseline when terms are added to a baseline query. These regressions are:

1. Regression of user nominated terms against percent change NDCG. This identifies an upper bound on the relationship between terms that represent the semantics of requests and terms that discriminate in a collection;

2. Regression of terms identified by individual word association methods against percent change NDCG;
3. Regression of all possible combinations of word association methods against percent change NDCG. This may be important if word association methods work collectively to identify discriminative terms.

Regressions use terms identified from description topics of the Robust04, GOV2 and WT10G collections and target percent change in NDCG. Data points represent single terms, and scores for statistical word association methods are linearly transformed to bring their values closer to the range of target values.[5] Two characteristics are reported: validated $R^2$, which measures the explanatory power or proportion of variance explained by a regression model, and *LMG* (Lindeman et al., 1980), which is a decomposition of variance similar to ANOVA.

$R^2$ is typically used to explain the agreement between targets and predictions,[6] but does not account for a tendency towards over-fitting when a model requires more information than predictors provide. Over-fitting causes regression to indicate a better fit, or less prediction error, than is really present. To avoid over-fitting, this analysis uses bootstrap validated $R^2$ (Efron, 1983). Validated $R^2$ fits a regression model to a training set composed of $n$ samples taken from the data with replacement. The test set contains all data. This delivers a slight downward correction for the estimated $R^2$.

*LMG* (named after Lindeman, Merenda and Gold who first reported the metric) provides a sum of squares decomposition of the variance accounted for by each word association method. It is similar to standard ANOVA, but ANOVA uses a *sequential* sum of squares. This means that the order of regressors can have a strong impact on the assessment of their relative importance. LMG uses an unweighted average over possible orderings of regressors (Lindeman et al., 1980) so it is more appropriate for predictors that have no natural order, such as word association methods. Furthermore, LMG accounts for uncertainty about the relationships between correlated regressors by allowing them to benefit from each others 'shares' of prediction importance. This provides an acceptable interpretation for relationships such as $X1 \rightarrow X2 \rightarrow Y$, where $X1$ and $X2$ are regressors, $Y$ is the target, and $\rightarrow$ indicates an influence or effect.

[5] Statistical word association scores are divided by a constant: 500,000 for Robust04 and WT10G, and 5,000,000 for GOV2.

[6] $R^2$ is the coefficient of determination, defined as the squared correlation between predictions and targets.

| Regression | Max CN |
|---|---|
| Robust04 | 62.0 |
| GOV2 | 20.1 |
| WT10G | 21.2 |

Table 5.5: The condition numbers for all regressions against terms that improve NDCG are within acceptable bounds except those using more than two regressors to predict terms that improve Robust04 NDCG. This indicates that multicollinearity may be a concern, but is not a clear problem in all but the identified cases.

### 5.2.2 Assumptions

A concern for term data is the potential for multicollinearity, as suggested by Table 4.3.[7] Multicollinearity occurs when two or more predictor variables in a single model are highly correlated, meaning that one variable can be determined with non-trivial accuracy from a linear combination of the others. On the one hand, moderate multicollinearity is fairly common and may not matter if two conditions hold. First, each sub-type of the data population occurs with similar frequency, and second, there is no requirement to handle outliers well (or no outliers, as for request term data). On the other hand, if multicollinearity is severe, then the matrix inversion used in regression is ill-conditioned and can be highly sensitive to slight variations in data. By consequence, the estimated regression coefficients are numerically unstable and the results of regression may be invalid.

For query term data, there are no outliers and all unigrams, bigrams and trigrams are included so that each sub-type of the data population is reasonably evenly represented. In addition, the *condition number* is used to quantify the presence of multicollinearity. The condition number is a standard measure of ill-conditioning in a matrix (Belsey et al., 1980), computed by finding the maximum eigenvalue of the matrix, dividing by the minimum eigenvalue, and taking the square root. As a general rule, a condition number greater than 15 is cause for concern, and a number greater than 30 indicates a problem with multicollinearity (Belsey et al., 1980).

Table 5.5 shows the maximum condition number computed for all regressions using two or more word association methods as predictor variables. In general, condition numbers increase as more word association methods are used in a single model

[7]The three major assumptions of regression are: 1) the link function used for prediction is a linear combination of the predictive variables; 2) for categorial target data, the choice of membership in one category is not related to the choice of membership in another category (independence of target variables, or Independence of Irrelevant Alternatives, IIA); and 3) the observed data points for regressors are independent such that the model does not include any predictive variables that duplicate information provided by other variables (absence of multicollinearity).

(highest values for 9 methods in combination). Most condition numbers are within acceptable bounds except predictions for Robust04 using three or more word association methods.[8] Results are reported for all regressions, but care should be taken when interpreting the results of Robust04 regressions. These results are likely to be numerically unstable when more than two regressors are used.

### 5.2.3 Results

For all collections, the minimum and maximum values for validated $R^2$ are reported for regressions with an acceptable condition number ($< 30$). The method, or combination of methods, that delivers the maximum validated $R^2$ is assumed to be better for selection of discriminative terms than the other methods or combinations tested. Minimal association is found in all experiments. This suggests that there is very little relationship between both word association and discrimination, and semantic interpretation and discrimination, when discrimination is measured by percent change in NDCG.

#### 5.2.3.1 Semantics and discrimination

Word association methods are not strongly related to the semantics of requests, so the first regression explores the ability of user nominated terms to predict terms that improve NDCG. Results are reported in Table 5.6. Within this evaluation framework, there is very little relationship between user nominated terms, assumed to represent query semantics, and the discriminative ability of the same terms. Validated $R^2$ accounts for less than 1% of variance.[9] However, it appears that semantic understanding may be more important for Robust04. This collection contains particularly complex and difficult description topics.

#### 5.2.3.2 Association methods and discrimination

Table 5.7 shows the validated $R^2$ for regressions in which terms identified by individual word association methods predict percent change in NDCG. The first observation is that validated $R^2$ for all word association methods is very low. Individual methods account for less than 0.1% of the variance in prediction of discriminative terms. More-

[8] Percent improvement in MAP was explored as an alternative target, but resulted in a similar increase in condition number for Robust04.

[9] The $R^2$ for GOV2 is negative (-0.0001). Since $R^2$ compares the fit of a model with the fit of a horizontal straight line (the null hypothesis), a negative $R^2$ indicates that the model fits worse than a horizontal line. The model does not follow any trend in the data.

| **Regression** | $R^2$ | **CN** |
|---|---|---|
| Robust04 | 0.0215 | 15.4 |
| GOV2 | -0.0001 | 10.3 |
| WT10G | 0.0040 | 10.1 |

Table 5.6: Performance for prediction of percent change in NDCG using user nominated terms. Association is low for all collections, but relatively more important for Robust04. Robust04 contains particularly complex and difficult description topics.

| Collection | Selection by association method → % change NDCG: validated $R^2$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Uni | Seq2 | Seq3 | NPs | GDep | Cat | BPhr | MI | LogL |
| Robust04 | 0.0002 | 0.0011 | -0.0003 | 0.0020 | 0.0004 | -0.0003 | 0.0007 | -0.0014 | -0.0001 |
| GOV2 | 0.0000 | -0.0006 | -0.0007 | 0.0010 | -0.0009 | 0.0024 | -0.0001 | -0.0005 | -0.0005 |
| WT10G | 0.0023 | -0.0018 | -0.0019 | -0.0008 | 0.0000 | -0.0005 | -0.0007 | -0.0009 | -0.0003 |

Table 5.7: Validated $R^2$ for regression against terms that improve NDCG. Associations are very minimal and individual words are the only predictor that has no negative association.

| Collection | Selection by association method → $\Delta$ NDCG $> 0.02$: validated $R^2$ | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | Uni | Seq2 | Seq3 | NPs | GDep | Cat | BPhr | MI | LogL |
| Robust04 | -0.0067 | 0.0253 | -0.0157 | 0.0287 | 0.0134 | -0.0072 | 0.0184 | 0.0467 | -0.0127 |
| GOV2 | -0.0130 | -0.0109 | -0.0164 | -0.0132 | -0.0010 | -0.0228 | -0.0151 | -0.0333 | -0.0232 |
| WT10G | -0.0243 | -0.0191 | 0.0031 | -0.0134 | -0.0168 | -0.0124 | -0.0023 | -0.1747 | -2.5418 |

Table 5.8: Validated $R^2$ for regression against terms that achieve an absolute difference in NDCG $> 0.02$ are often negative, indicating that the regression model does not follow any trend in the data. Results appear to be affected by query difficulty.

| Regression | Min $R^2$ | Max $R^2$ |
|---|---|---|
| Robust04 | -0.0017 | 0.0048 |
| GOV2 | -0.0031 | 0.0026 |
| WT10G | -0.0094 | 0.0023 |

Table 5.9: No combination of word association methods is closely associated with terms that improve NDCG.

over, they tend to identify terms that are less effective than unigrams. This suggests that there is a minimal, or even an overall negative, relationship between individual methods of word association and term discrimination. No word association method is substantially better for document retrieval by this analysis.

An alternative analysis might consider absolute change in NDCG greater than some nominal value, such as 0.02, based on the observation that users cannot detect very small changes in IR effectiveness. Table 5.8 shows the results of this analysis. For GOV2 and WT10G the validated $R^2$ is negative, indicating that the regression model does not follow any trend in the data. Conversely, for Robust04 there is considerable improvement for several methods, particularly Seq2, NPs, and MI. I speculate that this may be due to the difficulty and complexity of Robust04 queries. These queries produce smaller absolute changes in NDCG than queries for other collections, and are more likely to benefit from language processing.

#### 5.2.3.3 Combined association methods and discrimination

Validated $R^2$ for all possible combinations of word association methods are shown in Table 5.9. Once again, validated $R^2$ is very low for all regressions. It accounts for less than 0.006% of model variance. This suggests that a particular combination of word association methods is not the most important factor that determines whether a term is discriminative. Other factors that are not accounted for in the model, such as term frequency in a retrieval collection, are more influential.

Given this observation, Table 5.10 shows the combination of word association methods that produced the highest validated $R^2$ with a condition number less than 30. According to this analysis, the combination of a few word association methods produces the best results but the improvements are marginal. This re-affirms empirical results in IR that while simple retrieval models are difficult to beat (Huston, 2013), the combination of several word associations can be profitable (see discussion, Section 3.4). Results vary for different collections so an optimal approach to the incorporation of word dependencies in IR is not apparent.

| Regression | Best combination | $R^2$ | CN |
|---|---|---|---|
| Robust04 | Uni Seq2 NP | 0.0033 | 25.0 |
| GOV2 | Seq2 NP Cat | 0.0026 | 15.5 |
| WT10G | Uni LogL | 0.0023 | 9.4 |

Table 5.10: The best performance for prediction of percent change in NDCG with a condition number (CN) $< 30$ occurs with a minimal number of features that vary for different collections.

Relative importance for the prediction of NDCG percent change compared to baseline was also computed for all pairwise combinations of methods (see Appendix D, and the example in Figure 5.2). Although the differences are marginal, results suggest that within the regression models, noun phrases dominate predictions for Robust04, catenae dominate predictions for GOV2 and unigrams dominate for WT10G. Several factors may contribute to these results.

Briefly, catenae may be influential for GOV2 because they recall many terms that align with request semantics (see Section 4.6.3). They may be less influential for Robust04 due to the increased potential for ambiguous prepositional phrase attachment given more complex topics. This limitation is described further in Section 6.2.2.

Unigrams may be important in WT10G due to the frequency of lexical relations between words as well as the discriminative ability of words individually. Lexical relations are not necessarily captured by syntactic structure or sequential word order (see Section 4.4.1). Nevertheless it is not clear why WT10G is distinguished in this way. It may be that many noun phrases in WT10G are better described by a series of unigrams. For example '*incandescent light bulb*' can be split into three desirable unigrams {*incandescent, light, bulb*}'. However, a manual inspection of WT10G topic descriptions did not support this theory. An alternative explanation is that the effectiveness of word association methods is affected by the sufficiency of relevance judgments. When a collection is built, a number of IR models are used to retrieve candidate documents. These candidate documents are judged for relevance. WT10G is a fairly dated collection, so it may be that the IR models used to retrieve candidate documents placed more emphasis on query likelihood (unigrams). By consequence, a model that uses word dependence might retrieve relevant documents that have not been judged. In this case, multi-word terms could appear to be less effective than unigrams. However, this is speculative.

Noun phrases appear to be the most important predictor for Robust04. Evidence for a pivotal role of noun phrases is found with pairwise combinations of predictors

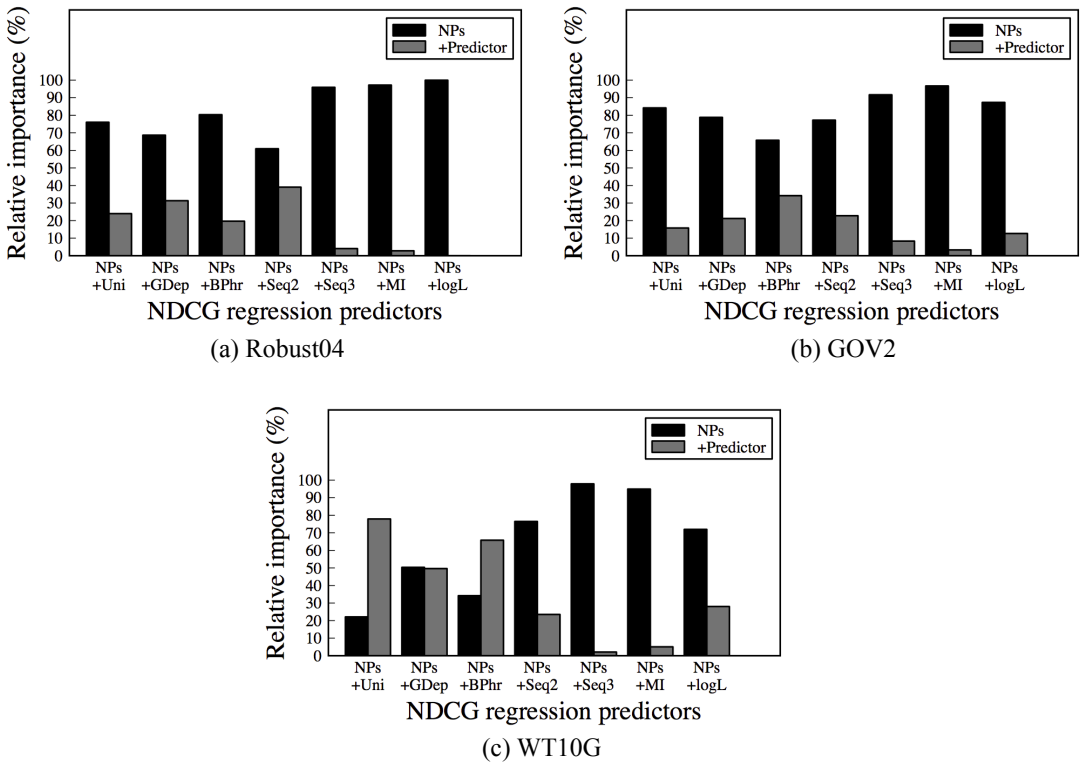



Figure 5.2: Relative importance of word association methods for improvement in NDCG for models using two predictors. Noun phrases have a similar pattern of behaviour for Robust04 and GOV2 regressions. The pattern varies for WT10G.

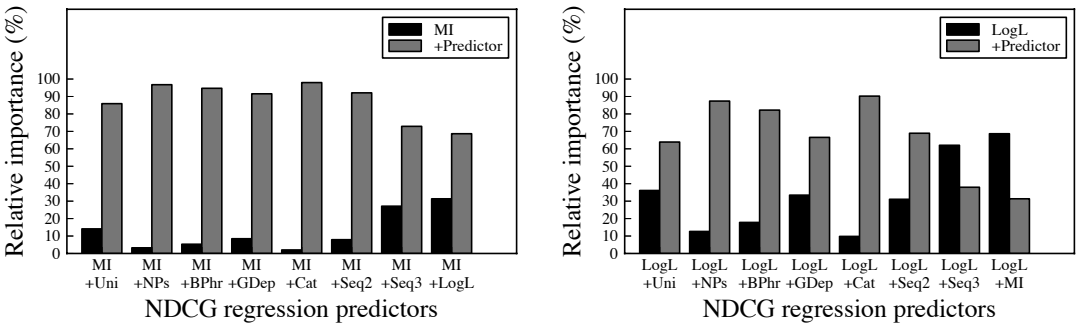



Figure 5.3: Terms identified using the log likelihood ratio (LogL) are generally more useful than terms identified by mutual information (MI). Plots shown for GOV2.

for both Robust04 and GOV2 (Figure 5.2), and results of other empirical research that demonstrates their effectiveness (Bendersky and Croft, 2008).

The following additional observations are made for Robust04 and GOV2. They are based on the fact that LMG shares credit for correlated regressors, so low relative importance reflects a limited number of discriminative terms identified by a word association method.

- The contribution of governor-dependent pairs (GDep) is not clear. GDep may contribute to the prediction of terms that improve NDCG as much as bigrams. These associations are considered further in Chapter 6.
- The contribution of bigrams is also inconclusive and resembles the behaviour of GDep pairs. Word co-occurrence is considered further in Chapter 7.
- Seq3 terms are relatively poor predictors of discriminative terms. Fewer documents meet the tighter constraints they impose on document content. In addition, Seq3 terms can be poorly composed for very long queries with many words that are not closely related to an information need. Users tend to compress long topics by selecting words from the full length of a topic and compiling them into a single term, rather than selecting three words in sequence.
- Terms identified using the log likelihood ratio (LogL) are generally more useful than terms identified by mutual information (MI) (Figure 5.3). Neither method is particularly influential for prediction of terms that improve NDCG in these experiments although they are reasonable for binary classification of terms that improve or decrease IR effectiveness. This may be due to the presence of query words that are not closely related to the information need, and do not enter into relationships that accurately reflect the information need. The relative importance

of LogL and MI terms concurs with theoretical expectations. Mathematically, LogL is better than MI for comparison of frequent and rare word combinations, and works well for large or small text samples. In contrast, MI is biased towards proper nouns (*low frequency*, high attraction collocations) rather than compound nouns that may be more likely to appear in queries (strong, *frequent* word associations).

## 5.3 Practical considerations

There appears to be a minimal relationship between word association, semantics, and the discriminative ability of individual terms, but it is difficult to determine from previous analyses where there may be a profitable balance between the information provided by word associations and the overhead of language processing.

Using combinations of word association methods for both individual terms and combinations of 2 or 3 terms, this balance was explored by plotting the overlap in terms that deliver an increase in percent change NDCG for particular combination types. An example is shown in Figure 5.4. The first subplot for GOV2 shows that noun phrases and bounded phrases are the only methods that detect terms with one word (apart from unigrams themselves). Further, it shows that when unigrams are combined in pairs, and their effect on NDCG is calculated as the average of their effects in each pair, bounded phrases identify almost 90% of unigrams that improve NDCG. Similar plots for Robust04 and WT10G, and for individual terms and combinations of 2 and 3 terms, are provided in Appendix E.

All combinations are composed of a term $x$ and 1 or 2 other terms selected using the same word association method that was used to select $x$. This approach avoids computation of exponentially many term and method combinations. Plots show data only for terms that improve NDCG. The effect on NDCG is computed as the average of the percent changes in NDCG attributed to $x$ for the query reformulations in which it is used. Catenae are excluded on the basis that they can be intensive to compute and do not contribute to a desirable balance of discriminative ability and efficiency of term identification.

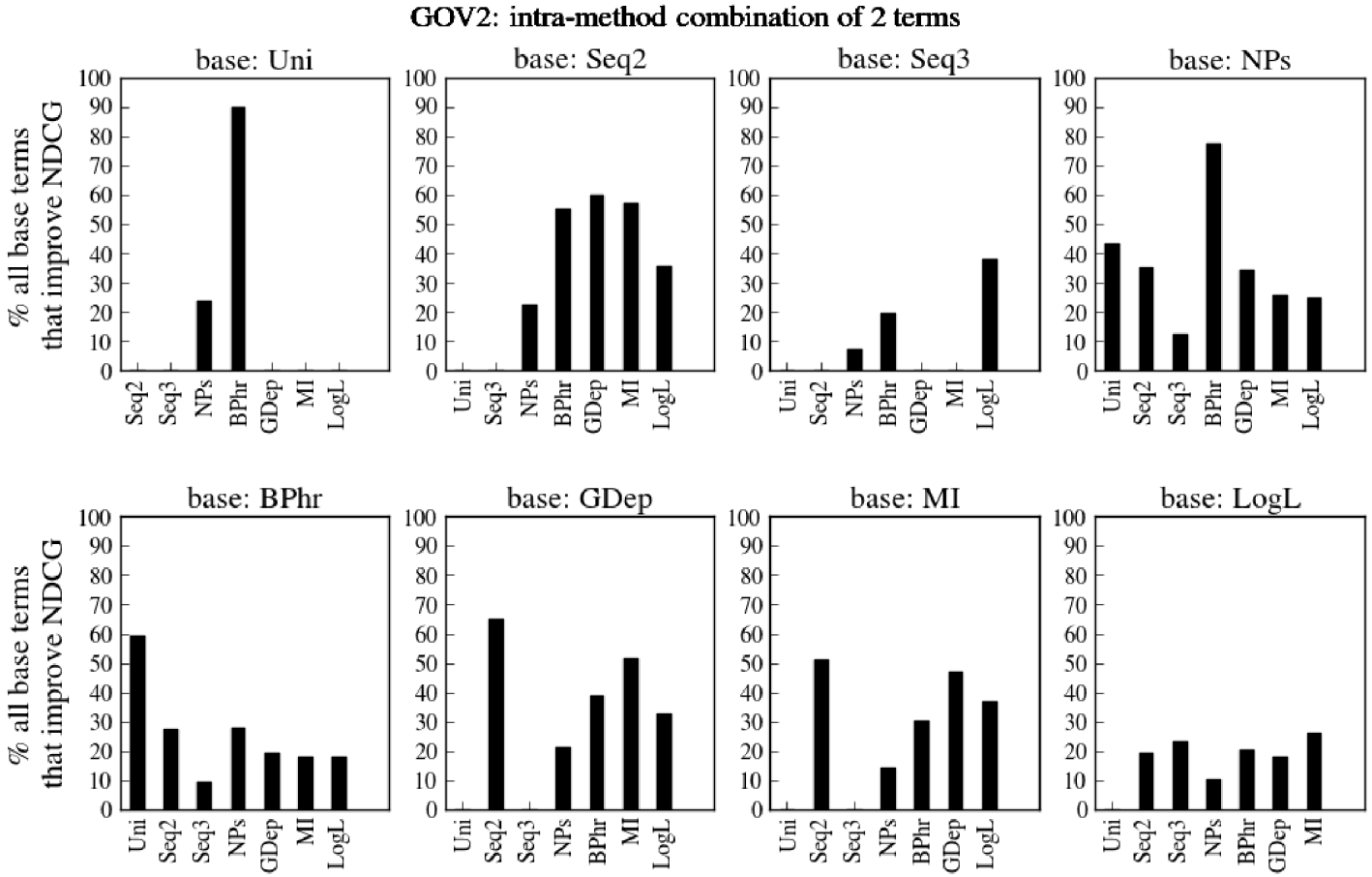



Figure 5.4: Results for combinations of two terms identified by each base method for the GOV2 collection. Data is shown for terms that contributed to an improvement in NDCG overall. Plots show the percentage of these terms identified by other methods. A substantial proportion of terms can be identified by alternative means.

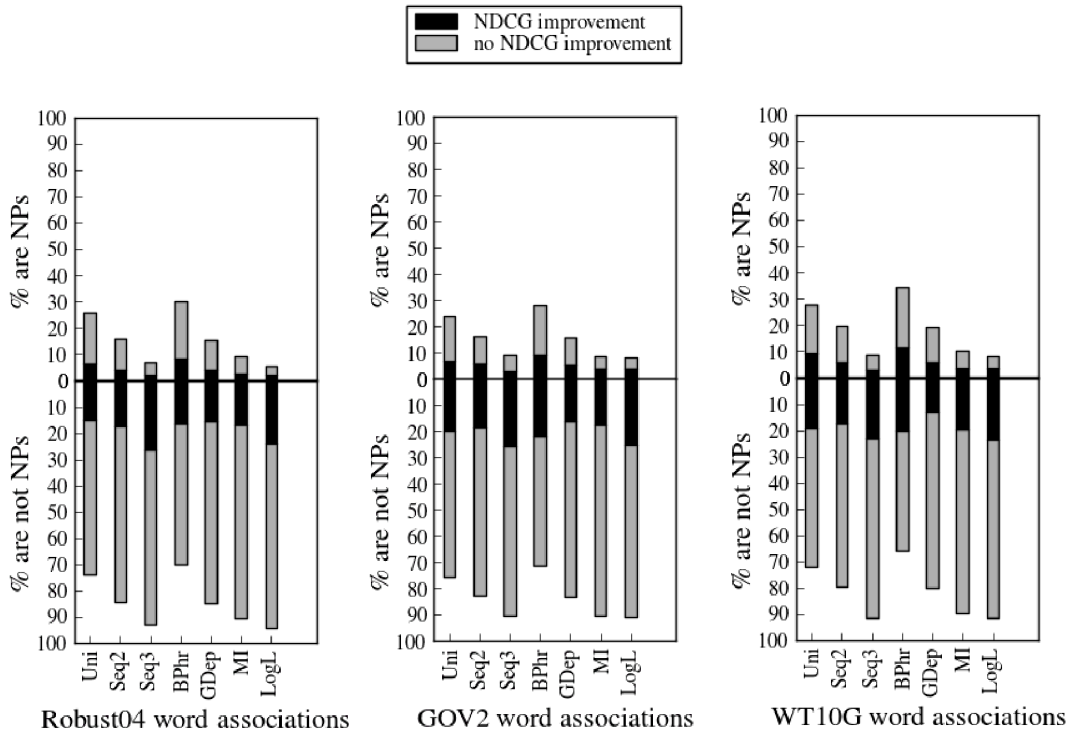



Figure 5.5: Noun phrases are one of the single most important word association methods for selection of discriminative terms, but may not identify *all* discriminative terms.

The visualisations enable observation of trends that might manifest for combinations of word association methods as they are applied in IR techniques. Data shows that terms of one association method that improve NDCG scores (on average) can sometimes be detected by the most important alternative word association methods. For example, across three collections, between 92% and 98% of noun phrases that improve NDCG can be detected using unigrams, bigrams and trigrams that require no linguistic processing. Moreover, as shown in Figure 5.5, the majority of unigrams, bigrams and trigrams that improve NDCG are not noun phrases. This means that while noun phrases identify more discriminative terms than non-discriminative terms, and might be the single most important method of term selection, non-linguistic methods are capable of detecting most of the discriminative terms identified by noun phrases as well as some discriminative terms that are not captured by noun phrases.

Based on this evidence, the advantage given to noun phrases for IR appears to be due to their linguistic discrimination. However, discrimination between terms can be achieved by other means. Simultaneous application of heuristic (syntagmatic) associations with statistical weighting has much potential to constrain terms, and may perform better than noun phrases. Combinations of word association methods can also

approximate text units other than noun phrases. These observations contribute to an explanation of the effectiveness of IR models that use no linguistic knowledge.

## 5.4 Conclusion

If only one method of word association is used for query term selection and the terms are unweighted or unfiltered, then noun phrases (NPs) are generally the best choice. They describe some of the most discriminative terms and work well together to describe an information need. Their discriminatory properties appear to outweigh the limitations of phrase structure grammars noted in Chapters 3 and 4 with respect to semantic characterization. However, noun phrases lose much of their advantage when combined with other features of text. Their contribution to query effectiveness can be largely approximated by unigrams and bigrams. So, while noun phrases are discriminative, they do not necessarily hold a privileged position for term selection when applied in IR techniques that combine multiple text features.

If one method of word association is used with a filtering mechanism or term weighting, then catenae appear to be the best choice. Catenae describe more discriminative terms than any other method, and are reasonably stable in term combinations. Linguistic theory also indicates that they are more likely to align with request semantics at the level of surface text.

Terms selected by the log likelihood ratio (LogL) and bigrams (Seq2) may make a robust contribution to query effectiveness as observed in visualisations of term and method combinations. By consequence, these association methods should contribute positively to techniques that combine features of many linguistic representations. However, bigrams are less able than noun phrases and LogL terms to identify discriminative word combinations. Noun phrases are also adjacent word sequences and may be preferred in this respect.

Overall, the experiments in this Chapter suggest that word associations have a marginal relationship with the discriminative ability of terms, but this relationship might still be leveraged for small gains in IR performance by feature-driven probabilistic techniques. The way in which association methods and terms are combined appears to be more important than any limitations inherent in their individual motivating principles and semantic alignment. In addition, some terms are more discriminative when combined with other terms. Since term combinations can reduce query effectiveness, they are best applied with a careful selection mechanism.

In the next Chapter, I describe a novel approach that conceives of query reformulation as a two stage process that selects a few terms from a candidate pool. The approach focuses on identification of discriminative terms with strong semantic alignment, where the semantic alignment emerges from the use of catenae and linguistic properties of language structure. Discriminative ability is ensured by the application of typical IR features such as word and term frequencies. The approach explores the possibility of a fully automated, semantically motivated IR technique that does not reference external semantic resources. It contrasts with work in the following Chapter that uses simple word adjacency and term frequency statistics with random word combinations (nterms).

# 6

# Semantically Motivated Term Selection

Regression and simple term overlap experiments suggest that a term that represents the semantics of a request does not necessarily discriminate well between relevant and non-relevant documents. However, it is worth considering the effectiveness of query terms selected by a fully automated, semantically motivated IR technique in practice. This provides further comparison for techniques that are not linguistically informed.

A novel approach to term selection in this Chapter aims to identify discriminative query terms with a strong semantic alignment. In contrast to previous work (see Chapter3), this includes features of what is arguably a semantic phenomenon (Winkler, 2005) and not an approximation to one (annotated language structures). Such features may be preferable to features of linguistic representation because word association methods are not strongly related to semantic interpretation (Chapter 4).

The approach leverages catenae and properties of language structure to approximate request semantics, while the discriminative ability of term candidates is determined in part by typical IR features such as word and term frequencies. A catena (Latin for 'chain') is a word, or a sequence of words, that are continuous with respect to a walk on a dependency graph (see Section 4.5.5). They are a simplified representation of dependency paths that have been applied to ad hoc IR in the past, but the generation of term candidates with catenae is novel (see Section 6.1).

Catenae support a semantically motivated approach to IR in several ways. First, of all the word association methods tested in Chapter 4, excluding exhaustive enumeration of all combinations of 1-3 words, they have the highest recall of user nominated terms. User nominated terms are assumed to align with the semantics of a request. Recently, it has also been contended that catenae present a constraint on the semantic phenomena of ELLIPSIS (Osborne et al., 2012). This offers a tractable way to identify

a subset of semantically salient terms, and makes catenae a flexible representation of language semantics suitable for IR. Moreover, catenae appear better able to distinguish discriminative terms than other word association methods (excluding nterms, see Section 5.1.2).

Unfortunately catenae can also describe many uninformative word associations so they must be filtered before they are used in a query reformulation. A supervised machine learning technique proposed in this Chapter extends previous work on machine learning for term weighting and classification (Bendersky and Croft, 2008) by inclusion of novel semantically motivated features that are specific to paths. It also uses standard linguistic and discriminative features, such as grammatical categories and term frequencies. The comprehensive feature set builds on the idea that word association may be combined with standard discriminative features in IR to leverage small improvements in query term selection (Xue et al., 2010).

The rest of this Chapter proceeds as follows. The first Section describes the application of dependency paths for term selection. Section 2 describes the application of catenae as semantic units. This includes a hypothesis about the utility of induced features of ellipsis, an evaluation of its validity, and a discussion of the limitations of dependency paths as a method for detection of candidate antecedents. A supervised classifier for term selection is then described and applied in classification experiments. Finally, I report experimental results using terms identified with a combination of paths and the proposed supervised selection technique.

## 6.1 Dependency paths for term selection

The idea that sentence meaning can be flexibly captured by dependency parse tree fragments motivates an increasing number of techniques for automated language processing tasks such as paraphrasing, summarization, entailment detection, machine translation and the evaluation of word, phrase and sentence similarity. Dependency paths (or simply, 'paths') are used to determine the similarity, entailment or alignment between short texts. The basic framework compares the dependency structures for two texts (such as a query and a document sentence) using techniques such as tree edit distance (Heilman and Smith, 2010; Punyakanok et al., 2004), relation probability (Gao et al., 2004) and parse tree alignment (Park et al., 2011; Wang et al., 2007). A generic approach uses a matching function to compare a dependency path between any two stemmed terms $x$ and $y$ in a sentence $A$ with any dependency path between $x$ and $y$ in a

sentence *B*. The match score for *A* and *B* is computed over all dependency paths in *A*.

A well-known example of this approach uses dependency paths to detect paraphrases (Lin and Pantel, 2001). The authors derive the assumption that if two paths tend to link the same words, then the meanings of the paths are similar. Their unsupervised algorithm for Discovering Inference Rules from Text (DIRT) identifies equivalencies such as '*X is author of Y*' ≈ '*X wrote Y*'. Typed dependency relations are strung together to extract dependency paths between 'slot-fillers' that are constrained to be nouns. For example, *John found a solution to the problem* is represented as N:subj:V ⟵ *find* ⟶ V:obj:N ⟶ *solution* ⟶ N:to:N. The method determines the strength of associations between slots and fillers, so that by taking account of the relative importance of each slot-word feature, the similarity of two paths can be calculated from their common features. These similarities are stored and used to augment queries.

In IR, much work with dependency paths focuses on question answering (QA) where textual inference requires attention to linguistic detail. Paths are used to improve question representation, answer selection and answer ranking compared to methods that use a bag-of-words approach and ngram features (Surdeanu et al., 2011). For example, Echihabi and Marcu (2003) align all paths in questions with trees for heuristically pruned answers; Cui et al. (2005) score answers using a variation of the IBM translation model 1; Wang et al. (2007) use quasi-synchronous translation to map all parent-child paths in a question to any path in an answer; and Moschitti (2008) explores syntactic and semantic kernels for QA classification.

In ad hoc IR, most models that go beyond a word independence assumption use word co-occurrence and proximity (Metzler and Croft, 2005; van Rijsbergen, 1977; Song and Croft, 1999; Srikanth and Srihari, 2002). Few path-based methods have been explored, largely because parsing large document collections is computationally prohibitive. Methods that do parse documents typically use governor-dependent relations to normalize spurious differences in the surface text of requests and documents. Syntactic language models for IR (Cai et al., 2007b; Gao et al., 2004; Lee et al., 2006; Maisonnasse et al., 2007) fall in this category. The quasi-synchronous translation model for IR (Park et al., 2011) does not limit paths, but still requires parsing a document collection. This model leverages the insight that semantically related words have a variety of direct and indirect relations.

In contrast, the application of catenae throws away the glue that binds words together and eliminates the need for parsing a document collection. Catenae are thus an economical and intuitive representation of dependency paths (see Figure 4.18). They

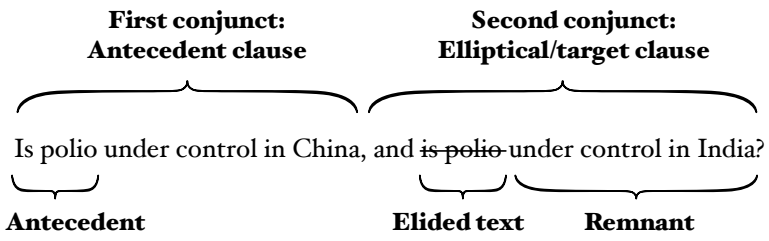



Figure 6.1: Ellipsis in a coordinated construct.

reduce paths to word sequences that can be interpreted using efficient and flexible proximity matching during search. For this reason they are compatible with a variety of existing IR models. However, they do not precisely identify discriminative terms (5.1.2). In the next Section, I make the novel proposal that features of ellipsis can contribute to the identification of catenae that are both discriminative and represent the semantics of a request.

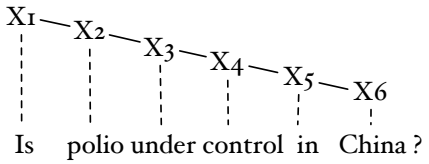
## 6.2 Catenae as semantic units

Catenae are dependency-based syntactic units with unique semantic properties. Specifically, it is claimed that catenae identify words that can be omitted in elliptical constructions (Osborne et al., 2012).[1] They thus flexibly represent salient semantic information in text.

Figure 6.1 shows terminology for the phenomenon of ellipsis. The omitted words are called *ellided* text, and I call words that could be omitted, but are not, *elliptical candidates*. Ellipsis relies on the logical structure of a coordinated construction in which two or more elements, such as sentences, are joined by a conjunctive word or phrase such as '*and*' or '*more than*'. A coordinated structure is required because the omitted words are 'filled in' by assuming a parallel relation $p$ between the first and second conjunct. In ellipsis, $p$ is omitted and its arguments are retained in text. In order for ellipsis to be successful and grammatically correct, $p$ must be salient shared knowledge at the time of communication (Prince, 1986; Steedman, 1990). If $p$ is salient then the omitted text can be inferred. If $p$ is not salient then the omission merely results in ungrammatical, or incoherent, sentences.

This framework is practically illustrated in Figure 6.2 for the topic, '*Is polio under*

[1] Catenae are claimed to account for gapping, and other types of ellipsis, as well as various discontinuities including fronting, scrambling, extraposition, verb complexes and idioms (Osborne and Groß, 2012).




**Ellided sentences**

Is polio under control in China, and...

a) in India ?
b) is cancer under observation ?
c) * cancer observation ?
d) * under India ?

**Ellipsis candidates marked in italics: they are catenae**

a) Is polio under control in China, and ( *is polio under control )* in India ?
b) Is polio under control in China, and is cancer under observation ( *in China* ) ?
c) * Is polio under control in China, and ( *is* ) cancer *( under )* observation ( *in China* ) ?
d) * Is polio under control in China, and ( *is polio )* under *( control in* ) India ) ?

Figure 6.2: For ellipsis to be successful, ellided words must be catenae. Ellipsis candidates are catenae.

*control in China?*'. A parallel proposition is created by appending the topic to itself to create a *coordinated topic*. For a natural language topic *x*, a coordinated topic takes the form "*x*, and *x*". In this Chapter, one or more words in coordinated topics may be selectively replaced in examples to make them easier to read e.g. '*Is polio under control in China, and is polio/cancer under control/observation in China/India*?'. This highlights the existence of a parallel relation between conjuncts. In each case, the antecedent is in the first conjunct and parallels the ellided words. In Figure 6.2, the coordinated topics marked by * are incoherent, and it is evident that the omitted words do not form a salient semantic unit. They also do not form catenae. In contrast, the omitted words in successful ellipsis do form catenae, and they represent informative word combinations with respect to the topic. The observation that ellided text often forms a catena leads to an *ellipsis hypothesis* that is the basis of experimentation:

> **Ellipsis hypothesis:** For natural language topics, induced features of ellipsis help to identify query terms that improve search precision and recall.

Clearly this hypothesis should be tested within an IR framework, but it is also helpful to consider whether the potential for ellipsis is a tractable surrogate for the semantic interpretation of requests. Semantic accessibility is the motivating principle behind

the hypothesis, and some preliminary experiments help to establish the validity of this idea.

The next Section reports three results that evaluate respectively: the ability of candidate antecedents to identify user nominated terms; the assertion that catenae detect candidate antecedents (Osborne, 2005); and the ability of candidate antecedents to identify terms that improve NDCG. A more comprehensive evaluation in the following Sections explores an implementation of induced features of ellipsis in an IR technique.

### 6.2.1 Evaluation of ellipsis

In order to evaluate whether the potential for ellipsis is a tractable surrogate for the semantic content of requests, it is necessary to identify text units that may be successfully ellided from requests and manually judge these for acceptability. This is achieved for a limited sample of 30 TREC description topics.

A limited sample is used for two reasons. First, the projection of a coordinated topic ('*q*, and *q*') from some description topics does not make grammatical sense and so cannot be judged for semantic accuracy by an English speaker. Any topic that contains ellipsis, coreference or coordination is problematic. For example, consider the topic, '*What is the current role of the Civil Air Patrol and what training do participants receive?*'. This contains an ellipsis of '*of the Civil Air Patrol*', as in '*What is the current role of the Civil Air Patrol and what training do participants ~~of the Civil Air Patrol~~ receive?*'. A partial analysis of this topic with respect to ellipsis is presented in Table 6.1. Some ellipses are possible and interpretable if the topic is broken up into components. For example, the antecedents '*current role*' and '*training receive*' are identifiable as shown. However, these antecedents are not available when the topic is treated as an indivisible text because the resulting sentences are implausible. A native speaker is very unlikely to ask, '*What is the current role of the Civil Air Patrol and what training do participants receive and the American Marine Corps and what training do participants receive?*'

Similar challenges arise for topics that contain coreference and other forms of coordination, such as '*What harm do cruise ships do to sea life, and what is the extent of the damage ~~cruise ships do to sea life~~?*' (where '*damage*' refers to '*harm*'), and '*Describe the collared peccary and its geographic range*' (where '*its*' refers to '*the collared peccary*').

In addition, enumeration of *every* possible ellided sentence is prohibitively difficult,

| Subtopic | Coordinated Subtopic with Ellipsis | Antecedent Term |
|---|---|---|
| *What is the current role of the Civil Air Patrol?* | What is the current role of the Civil Air Patrol, and ~~what is the current role of~~ the American Marine Corps? | current role |
| *What training do participants of the Civil Air Patrol receive?* | What training do participants in the Civil Air Patrol receive, and ~~what training do~~ participants in the American Marine Corps ~~receive~~? | training receive |

Table 6.1: Possible stoplisted antecedents for coordinated topics with ellipsis, based on sub-topics of '*What is the current role of the Civil Air Patrol and what training do participants receive?*'

even for a simple topic. If there are $n$ words in a topic, then there are $n!$ possible word combinations that are candidate antecedents. For a single topic, this can easily generate more than 350,000 ellided sentences for review. Clearly, it is impractical to consider all these possibilities for a large number of topics.

Due to these challenges, I select 30 description topics that do not contain ellipsis, coordinated structures or co-reference and use these for a limited exploration of relationship between catenae, possible antecedents and discriminative query terms. The set is biased towards shorter topics with simple syntax but is sufficient to indicate a general trend in results.

The procedure for detection of word combinations whose deletion would result in successful ellipses is as follows. Given a topic such as '*What methods are used to control type II diabetes?*', a coordinated topic is constructed by appending the topic to itself e.g. '*What methods are used to control type II diabetes and what methods are used to control type II diabetes?*'. A possible antecedent is a text unit, such as '*what methods are*' that can be successfully elided. Detection of all the possible antecedents in a coordinated topic is simplified by reducing the number that must be considered. Specifically, it is assumed that some word combinations cannot be successfully ellided. The number of possible antecedents is reduced by dividing the original topic into plausible text units according to simple rules.[2]

[2]The rules for dividing a topic into text segments are: (1) If a noun has a determiner, then a possible antecedent including the noun must include the determiner, and a possible antecedent including the determiner must include the noun; (2) If a main verb has an auxiliary, then a possible antecedent including the auxiliary must include the main verb, and a possible antecedent including the main verb must include the auxiliary; and (3) A possible antecedent including one word in a noun group (a compound noun or noun-adjective combination) must include the other words in the noun group unless rephrasing using '*of*' is possible with no change in semantics. In the latter case, the noun group should be split where '*of*' would be inserted. For example, '*type II diabetes*' can be re-phrased as '*diabetes of type II*' with no change in meaning, so it splits into '*type II*' and '*diabetes*'.

The smaller parts into which a topic is split are base antecedents. The base antecedents for the example topic are: {*what, methods, are, used, to control, type II, diabetes*}. Combinations of base antecedents are also possible antecedents, subject to the constraint that they include no more than three content-bearing words (non-stopwords) e.g. '*what methods*' is a possible antecedent. For the example topic, there are 127 possible antecedents when combinations are considered.

For each possible antecedent, a coordinated topic is created using the original query, and the possible antecedent is deleted from the second conjunct. For example, in order to decide whether the text unit '*what methods are used to*' can be ellided (is an antecedent), an annotator judges the acceptability of the resulting sentence, '*What methods are used to control type II diabetes and control/diagnose type II diabetes?*'. For the example topic, this process identifies 25 successful ellipses that collapse to 6 successful antecedents following removal of stopwords: {*methods, methods type II, diabetes, type II diabetes, type II, methods diabetes*}.

Evaluation of 30 description topics identified 468 successful ellided sentences. Corresponding antecedents were then compared with user nominated terms. Precision, recall and F1 score for matching terms were 0.25, 0.46 and 0.30 respectively, indicating that potential for ellipsis is not an ideal surrogate for the semantic interpretation of requests. However, heuristic constraints on the generation of candidate ellided sentences may not fully specify antecedents that result in successful ellipsis. In addition, although these measures are not directly comparable to measures for all queries presented in Section 4.6.3, they suggest that antecedents may be at least as good as the best word association method for identification of user-nominated terms.

The precision, recall and F1 score for identification of antecedents by catenae were also explored. These were 0.45, 0.70 and 0.52 respectively. The data suggest that catenae are associated with antecedence, but are not a comprehensive constraint on ellipsis as contended by Osborne and Groß (2012). It appears that ellipsis as a semantic phenomenon is not amenable to purely syntactic interpretation. The limitations of dependency parsing described in Chapter 4 may also be a factor in this result. Further reasons why catenae may fail to detect all possible antecedents, and more generally, desirable terms for IR, are discussed in the next Section.

Finally, the ability of antecedents to identify terms that improve NDCG was explored. The precision, recall and F1 score for this classification were 0.73, 0.47 and 0.54 respectively. This suggests that induced features of ellipsis may help to identify query terms that improve search precision and recall.

### 6.2.2 Limitations of paths and catenae

There are four problems that can limit the effectiveness of any technique that uses catenae or dependency paths for IR:

1. **Parser error:** As with any natural language processing, dependency parsing is prone to error. The pseudo-projective dependency parser used in this Chapter has a syntactic accuracy of 82% - 90%. The Stanford PCFG parser against which it is compared achieves 80% accuracy on typed dependencies. In both cases, there is significant error that will affect the ability of catenae to predict successful antecedents.

2. **Syntactic ambiguity:** A single dependency parse may only partially represent the ambiguous semantics of a description topic. A simplifying assumption that the most probable parse is accurate and sufficient for extraction of relevant catenae is not always true. In particular, there may not be one 'correct' attachment for prepositional phrases. This is suggested by successful ellipses in which ellided words only form catenae if multiple competing parses are accepted. For example, the ellided words '*is polio in china*' are relevant to a topic, '*Is polio under control in China?*' and make an acceptable antecedent. This can be observed from the coordinated sentence, '*Is polio under control in China, and ~~is polio~~ under observation ~~in china~~?*'. However, Figure 6.3 (a) shows that the ellided text does not qualify as a catena. To qualify as a catena, a parse with an alternative prepositional phrase attachment is required (see Figure 6.3 (b)). At the same time, the ellided words '*is polio under control*' are also relevant to the topic and make an acceptable antecedent. This can be observed with the coordinated sentence, '*Is polio under control in China, and ~~is polio under control~~ in India?*'. Yet this time the ellided text does not qualify as a catena in Figure 6.3 (b). It requires the parse in Figure 6.3 (a). Notice that 'more accurate' parsing does not address such problems of syntactic ambiguity.

   The impact of this phenomenon can be observed for one of the 30 topics evaluated in the previous Section. The topic, '*What restrictions are placed on older persons renewing their drivers' licenses in the U.S.?*' had one of the smallest overlaps between catenae and successful antecedents, accounting for 11% of all missed antecedents. Notice that the topic wording makes it difficult to determine whether the user seeks information about restrictions placed on older persons, or restrictions placed on the renewal of drivers licenses. This distinction affects

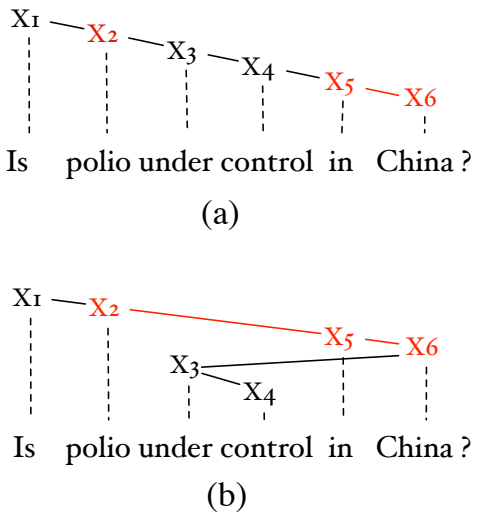



Figure 6.3: A parse in which '*polio China*' is not a catena (a), and a parse in which it is a catena (b) assuming removal of stopwords e.g. '*in*'.

dependency structure and the terms that might be selected to represent query meaning. Readers may infer that documents about either type of restriction meet the information need, but automated methods can select only one 'best' interpretation. Such problems might be circumvented by varying the number of parses used for identification of catenae. However, average improvements in term selection from such an approach are unlikely to be substantial (Chrupala et al., 2010).

3. **Rising:** Automatic extraction of catenae is limited by the phenomenon of rising. Let the *governor* of a catena be the word that licenses it (in Figure 6.4 '*used*' licenses '*a toxic chemical*' e.g. 'used what?'). Let the *head* of a catena be its parent in a dependency tree. Rising occurs when the head is not the same as the governor. This is frequently seen with *wh*-fronting questions that start *who, what* etc., as well as with many other syntactic discontinuities (Osborne and Groß, 2012). More specifically, rising occurs when a catena is separated from its governor by words that its governor does not dominate, or the catena dominates the governor, as in Figure 6.4. Note that in the risen structure, the words for the catena '*chemical as a weapon*' are discontinuous on the surface, interrupted by the word '*used*'. This means that the informative term '*chemical weapon*' cannot be identified directly from the risen structure.

   A consequence of rising is shown by another of the topics evaluated in Section 6.2.1. The topic, '*What security measures have been employed at train*

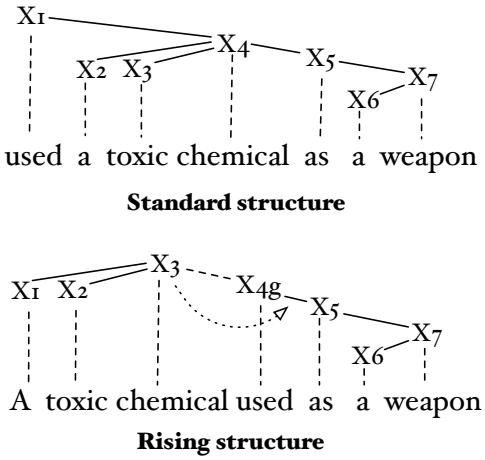



Figure 6.4: A parse with and without rising. The dashed dependency edge marks where a head is not also the governor and the g-script marks the governor of the risen catena.

*stations due to heightened security concerns?*' had the smallest overlap between catenae and successful antecedents, accounting for 25% of all missed antecedents. Notice that the phrase '*have been employed*' has risen from its position in a canonical order, '*What security measures at train stations have been employed due to heightened security concerns?*'. Moreover, an alternative canonical order (at least with respect to semantics) might be, '*Train stations have employed security measures due to heightened security concerns*'. Many more catenae identify successful antecedents when words take a canonical order.

**Overgeneration:** Catenae have been proposed as a necessary, but not sufficient condition of antecedence, leading to the expectation that a significant number of catenae will not identify informative terms for IR. Indeed, an analysis of catenae derived from 100 sample description topics shows that only around 25% of catenae improve NDCG compared to baseline query likelihood when used in a reformulated query. For this reason, although catenae describe relatively few of the possible word combinations in a description topic, it is necessary to discriminate between informative and uninformative catenae before they are applied in an IR model.

## 6.3 Selection method for catenae

Term weighting is a relatively well understood method for discrimination of informative terms. Word and term frequencies collected from retrieval collections, Wikipedia, google ngrams and commercial query logs are known to be indicative of term informa-

tiveness (Bendersky et al., 2011). Various linguistic features are also used in methods that learn term weights (Xue et al., 2010). There is less consensus about which features and constraints result in the greatest IR effectiveness when applied to dependency paths.

Some approaches treat all paths as equally informative (Moschitti, 2008; Park et al., 2011; Punyakanok et al., 2004) but this can generate noisy word relations and is computationally intensive. Heuristic filters are often applied because no explicit information in text indicates which paths are relevant. Unfortunately, these heuristic filters are sub-optimal. For example, consider the catenae captured by heuristic filters for the description topic, '*What role does blood-alcohol level play in automobile accident fatalities*' (Table 6.2). It may be obvious that the terms '*role play*' and '*level play*' are not representative of the topic, yet these catenae are described by parent-child relations that are commonly used to filter paths in text processing applications. Alternative filters that avoid such trivial word combinations omit description of key entities such as '*blood alcohol*'. They also identify longer catenae that may over-specify documents, resulting in poor retrieval (Cui et al., 2005; Lin and Pantel, 2001; Shen et al., 2005). Such shortcomings suggest that an optimized selection process may improve the performance of techniques that use dependency paths in ad hoc IR.

This Section describes a supervised method for selection of catenae as a simplified representation of paths that uses induced features of ellipsis. The method extends an approach to weighting noun phrases (NPs) presented by Bendersky and Croft (2008), as well as research on the variability of governor-dependent pairs (Song et al., 2008b). In contrast to prior work, it includes semantically motivated features specific to catenae and dependency paths, and selects among units containing more than two content-bearing words.

### 6.3.1 Methodology

Selection of catenae is framed as a supervised classification problem. Two training labels are explored: change in MAP, and binary human judgments of how well catenae represent the semantics of a description topic and discriminate between relevant and non-relevant documents in a collection (see Section 6.3.2). Formally, training data is considered as a sequence of labelled examples $\langle c_i, y_i \rangle$, where $c$ is a catena and $y$ is a label. A feature vector $v_c$ associated with each $c_i$ is used to predict the label for $c_i$. The Weka (Hall et al., 2009) AdaBoost.M1 meta-classifier (Freund and Schapire,

| **Query** *What role does blood-alcohol level play in automobile* accident fatalities*? (*abbreviated to `auto', `fatal')* | | | | |
|---|---|---|---|---|
| **Catenae** | **Gov-dependent** | **Predicate-argument** | **Nominal end slots** | **Seq dependence** |
| blood alcohol<br>level play<br>auto accident<br>accident fatal<br>role play<br>play fatal<br>blood alcohol play<br>play accident fatal<br>auto accident fatal<br>level play fatal<br>role play fatal<br>role level play | blood alcohol<br>level play<br>auto accident<br>accident fatal<br>role play<br>play fatal | auto accident<br>accident fatal<br>play fatal<br>play accident fatal<br>auto accident fatal | auto accident<br>accident fatal<br>auto accident fatal<br>level play fatal<br>role play fatal | blood alcohol<br>level play<br>auto accident<br>accident fatal<br>role blood<br>alcohol level<br>play auto |

Table 6.2: Catenae derived from dependency paths, as selected by heuristic methods. Selections are compared to sequential bigrams that use no linguistic knowledge.

1996) with unpruned C4.5 decision trees as base learners is applied to classify catenae as informative or the opposite. Adaboost.M1 boosts decisions over $T$ weak learners for $T$ features using weighted majority voting. At each round, predictions of a new learner are focused on incorrectly classified examples from the previous round. Adaboost.M1 was selected in preference to other algorithms because it performed better in preliminary experiments, it leverages many weak features to advantage, and usually does not overfit (Schapire et al., 1997). Classifier predictions are made using 10-fold cross-validation split by description topic.

### 6.3.2 Training data

Candidate catenae for testing and training contain one to four content-bearing words (stopwords were not permitted, see collapsed format, Section 3.2.2). The constraint on catenae length enables more efficient extraction and was based on evidence of effective terms for IR and text summarization. Text authors rarely assign keyphrases to documents with more than three words (Turney, 1999), and queries with less then four words account for 90.3% of all queries in a major search engine query log (Bendersky and Croft, 2009). Catenae were extracted from a dependency parse generated by a pseudo-projective joint dependency parser and semantic role labelling system (Johansson and Nugues, 2008) (see Section 3.2.3). This parser enables exploration of semantic classification features and is highly accurate. However, any dependency parser can be substituted. For comparison, 77% of catenae extracted from 500 description topics using the Stanford dependency parser (de Marneffe et al., 2006) are also extracted by the joint dependency parser.

For training labels based on change in MAP, terms were classified into two bins: improvement and no improvement. The second approach used binary human judgments of informativeness (see Chapter 1). This is based on the elicitation of human judgments about linguistic plausibility for candidate collocations in psycholinguistics (Evert and Krenn, 2001; Lapata et al., 1999), and related work on term selection for IR at IBM (Luo et al., 2013). Annotators at IBM identified text spans called "mandatory matching phrases" (MMPs) in the surface text of questions that were used to train a statistical classifier. MMPs correspond to semantic units that are likely to appear in QA answers. Two subjects were used to annotate 201 questions and no indication is given that the annotations overlap.

For the human annotated labels applied in this Chapter, annotations were provided for 11,188 catenae by one subject. Single annotations are used due to the expense of data acquisition and lack of expected improvement when annotations from multiple annotators are combined. This expectation is based on preliminary experiments in which two native speakers were prompted with terms automatically selected for 100 randomly chosen description topics. Inter-annotator agreement was borderline according to established criteria (Carletta, 1996; Fleiss, 1981) (Cohen's kappa $\kappa = 0.63$, taking into account expected chance agreement).[3,4] However, in test-retest conditions, both subjects failed to provide consistent judgments. Reliability for each subject was similar to the inter-annotator agreement (around $\kappa = 0.62$). By consequence, the cost involved in acquiring multiple sets of annotations was disproportionate to the likely gain in the reliability of labels. The results for training against human annotations are reported alongside MAP as an extension to the semantic focus for IR that is the purpose of this Chapter.

Overall, there are roughly three times the number of uninformative catenae compared to informative catenae and the number of training examples is relatively small (1295 to 5163 per collection). To improve classification accuracy, the training data for each collection is supplemented and balanced with data for other collections used in this dissertation, plus TREC8-QA. For example, training data for Robust04 includes

[3]Cohen's kappa measures pairwise agreement on category judgments, correcting for expected chance agreement. It is defined as $\kappa = \frac{P(A)-P(E)}{1-P(E)}$ where $P(A)$ is the proportion of times that annotators agree and $P(E)$ is the proportion of times that they are expected to agree by chance (Carletta, 1996). While some criteria classify $\kappa$ in the range of 0.40 to 0.75 as fair to good (Fleiss, 1981), researchers often think that $\kappa > 0.8$ is good reliability, and $0.67 < \kappa < 0.8$ permits tentative conclusions (Carletta, 1996).

[4]Inconsistency in user judgments is a known challenge. For example, one similar study asked users to identify five types of word combinations from automatically extracted governor-dependent word pairs: idiomatic expressions, technical terms, support verb constructions, stock phrases and entities (Pecina and Schlesinger, 2006). The $\kappa$ for each of these categories ranged from 0.29 to 0.49.

data from WT10G, GOV2 and TREC8-QA. Any examples that replicate catenae in the test collection are excluded. For Robust04, WT10G and GOV2 respectively, 30%, 82% and 69% of the training data is derived from other collections. This has the added benefit of not tying classifiers to corpus-specific features, even though a separate classifier is trained for the Robust04, WT10G and GOV2 collections.

### 6.3.3 Classifier features

Discrimination uses key aspects of heuristic filters as well as novel features that characterize catenae and paths. The basic idea is that selection, or weighting, of catenae can be improved by features that are specific to paths, rather than generic for all terms. Features from past work in IR are also incorporated. In all, there are four feature classes. These are presented in detail in Tables 6.3 and 6.4 and summarized below.

1. **Ellipsis candidates:** The ellipsis hypothesis suggests that informative catenae can be successfully ellided. In order to extract characteristic features of ellipsis, a two stage process is required. This involves: (1) construction of a *coordinated topic* by adding a topic to itself e.g. '*x* and *x*'; and (2) ellipsis of catenae from the second conjunct.

   A coordinated topic converts a description topic into the form required for successful ellipsis. Example (a) below shows the coordinated topic for '*Is polio under control in China?*'. Following ellipsis of the catena '*under control*', example (b) shows the *topic remainder*.

   (a) Is polio under control in China, and is polio under control in China?
   (b) Is polio under control in China, and is polio in China?

   Notice that the topic remainder is different from the remnant as defined in linguistics (see Figure 6.1). It is used to identify features detailed in Table 6.3:

   - Minimum perplexity of ngrams with 2, 3, and 4 words around the extraction site of ellided text. Perplexity is calculated using a language model of English Wikipedia. For example, for the topic remainder '*Is polio under control in China and is polio __ in India?*' the ngrams would be {'*and is polio in*', '*is polio in*', '*polio in*'};
   - Compliance with hand-coded rules for grammaticality (strict and relaxed classes, see Appendix F). Examples include unlikely token sequences such

as a double comma (,,), and orphaned words, such as an adjective without a noun;

- Partial noun phrase, prepositional phrase or FINITE CLAUSE in the topic remainder. For example, for the prepositional phrase '*in China*', a partial prepositional phrase might be just '*in*'.

2. **Dependency path features:** There are two major types of dependency path features. The first type is often used to filter dependency paths, and involves grammatical categories and semantic roles. For example, a feature may require that words in a governor-dependent relation are also a noun phrase or part of the same predicate-argument structure. These features leverage simple conversions between dependency and phrase structure grammars.

   The second type of feature focuses on the variability of word separation for governor-dependent word combinations. It has been proposed that words in governor-dependent relations are more likely to identify informative terms if they are consistently found in close proximity in surface document text (Song et al., 2008b). Indeed, words that would normally form catenae are separated in a risen structure both syntactically and in surface word order. By consequence, the ability of catenae in requests to match similar word combinations in documents may be limited by variability in their surface appearance. In contrast to previous work (Song et al., 2008b), I describe word separation distances efficiently using collection statistics as a whole, rather than per-document statistics for every document in a collection. I also do not dependency parse documents, and apply features to multi-word units instead of limiting to word pairs. Specific features based on those used to filter dependency paths in the past include:

   - Minimum perplexity of the part-of-speech tag sequence for a catena, computed using a language model of part-of-speech tags built on parsed English Wikipedia data (Baroni et al., 2009);
   - The phrasal class of a catena, with options of noun phrase, verb phrase, or Other. A catena has a class of noun or verb phrase if it is such a phrase, or is entirely contained by one. If one phrase is embedded in another, the larger phrase is used;
   - Whether a catena is a predicate-argument structure, or is entirely contained by one;

| Ellipsis candidate features (E) | |
|---|---|
| *R_ppl1* | Minimum perplexity of ngrams with length 2, 3, and 4 in a window of up to a 3 words around the site of catenae omission. This is the area where ungrammaticality may be introduced. For the remainder R=`ABCDE&ABE' we compute ppl1 for {&ABE, &AB, ABE, &A, AB, BE}. |
| *R_strict* | Compliance with strict hand-coded rules for grammaticality of a remainder. Rules include unlikely orderings of punctuation and part-of-speech (POS) tags (e.g. ,, ), poor placement of determiners and punctuation, and orphaned words, such as adjectives without the nouns they modify. |
| *R_relax* | A relaxed version of hand-coded rules for *R_strict*. Some rules were observed to be overly aggressive in detection of ungrammatical remainders. |
| *NP_split* | Unsuccessful ellipsis often results if elided words only partly describe a base NP. Boolean feature for presence of a partial NP in the remainder. NPs (and PPs) are identified using the MontyLingua toolkit. |
| *PP_split* | As for *NP_split*, defined for prepositional phrases (PP). |
| *F_split* | As for *NP_split*, defined for finite clauses. |
| **Dependency path features (D)** | |
| *c_ppl1* | Dependency paths traverse nodes including stopwords and may be filtered based on POS tags. We use perplexity for the sequence of POS tags in catenae before removing stopwords. This is computed using a POS language model built on ukWaC parsed wikipedia data (Baroni et al., 2009). |
| *phClass* | Phrasal class for a catena, with options *NP*, *VP* and *Other*. A catena has a NP or VP class if it is, or is entirely contained by, an NP or VP (Song et al., 2008). |
| *semRole* | Boolean feature indicating whether a catena describes all, or part of, a predicate-argument structure (PAS). Previous work approximated PAS by using paths between head nouns and verbs, and all paths excluding those within base chunks. |
| *nomEnd* | Boolean indicating whether the words at each end of the catena are nouns (or the catena is a single noun). |
| *sepMode* | Most frequent separation distance of words in catena *c* in the retrieval collection, with possible values *S = {1, 2, 3, long}*. 1 means that all words are adjacent, 2 means separation by 0-1 words, and *long* means containment in a window of size $4 * \|c\|$. |
| *H_c* | Entropy for separation distance *s* of words in catena *c* in the retrieval collection. $f_s$ is the frequency of *c* in window size *s*, and $f_S$ is the frequency of c in a window of size $4 * \|c\|$. All *f* are normalized for catena length using $\|c\|^{\|c\|}$ (Hagen et al., 2011). $$H_c = \sum_{s \in S} \frac{f_s + 0.5}{f_S + 0.5} log_2 \frac{f_s + 0.5}{f_S + 0.5}$$ |
| *sepRatio* | Where $f_s$ and $f_S$ are defined as for *H_c*: $$sepRatio_c = \frac{f_{s>2} + 0.5}{f_S + 0.5}$$ |
| *wRatio* | For words *w* in catena *c* where $f_S$ is defined as for *H_c*. $$wRatio_c = \frac{0.5 + \frac{1}{\|c\|} \sum_{w \in c} f_w}{f_S + 0.5}$$ |

Table 6.3: Ellipsis and dependency path classifier features.

| Co-occurrence features (C) | |
|---|---|
| *isSeq* | Boolean indicating if catena words are sequential in stoplisted surface text. |
| *cf_ow* | Frequency of a catena in the retrieval collection, words appearing ordered in a window the length of the catena. |
| *cf_uw* | As for *cf_ow*, but words may appear unordered. |
| *cf_uw8* | As for *cf_uw*, but the window has a length of 8 words. |
| *idf_ow* | Inverse document frequency (*idf*) where document frequency (*df*) of a catena is calculated using *cf_ow* windows. Let $N$ be the number of documents in the retrieval collection, then: $$idf(C_i) = log_2 \frac{N}{df(C_i)}$$ and $idf(C_i) = N$ if $df(C_i) = 0$. |
| *idf_uw* | As for *idf_ow*, but words may appear unordered. |
| *idf_uw8* | As for *idf_uw*, but the window has a length of 8 words. |
| *gf* | Google ngrams frequency (Brants and Franz, 2006) from a web crawl of approximately one trillion English word tokens. Counts from a large collection are expected to be more reliable than those from smaller test collections. |
| *qf_in* | Frequency of appearance in queries from the Live Search 2006 search query log (approximately 15 million queries). Query log frequencies are a measure of the likelihood that a catena will appear in any query. |
| *wf_in* | As for *qf_in*, but using frequency counts in Wikipedia titles instead of queries. |
| **IR performance prediction features (I)** | |
| *c_len* | Length of a stopped catenae. Longer terms tend to reduce IR recall. |
| *WIG* | Normalized Weighted Information Gain (*WIG*) is the change in information over top ranked documents between a random ranked list and an actual ranked list retrieved with a catena *c* (Zhou and Croft, 2007). $$wig(c) = \frac{\frac{1}{k}\sum_{d \in D_k(c)} log\, p(c\|d) - log\, p(c\|C)}{-log\, p(c\|C)}$$ where $D_k$ are the top *k=50* documents retrieved with catena *c* from collection *C*, and $p(c\|\cdot)$ are maximum likelihood estimates. A second feature uses the average WIG score for all pairwise word combinations in *c*. |

Table 6.4: Co-occurrence and IR performance prediction classifier features.

- Whether the first and last word of a catena are nouns, or the catena is only one word long and is a noun;
- The most frequent separation distance of catena words in a collection;
- Uncertainty about the separation distance of words in a retrieval collection, as measured by entropy;
- The ratio of frequencies of word separation in long and short search windows in a retrieval collection;
- The ratio between the average frequency of individual words in a catena to the frequency of all words in a catena within a specified search window.

3. **Co-occurrence features:** A governor $w_1$ tends to SUBCATEGORIZE for its dependents $w_n$, and so often determines the choice of $w_n$. By consequence, co-occurrence is likely to be an important feature of dependency relations (Mel'čuk, 2003). In addition, word co-occurrence frequencies and inverse document frequencies are commonly used in IR. Co-occurrence features previously proposed for filtering terms in IR (Bendersky and Croft, 2008) are employed by the classifier and include:

   - Whether words in a catena are sequential in surface text;
   - The frequency of a catena in a retrieval collection, as measured using short (e.g. 2 word) and long (e.g. 8 word), ordered and unordered windows;
   - The inverse document frequency of a catena in a retrieval collection, measured using short and long, ordered and unordered windows;
   - Catena frequencies in Google ngrams, a commercial query log, and English Wikipedia titles.

   Since catenae are of variable length, and word combinations with fewer words are more frequent, two methods are used to normalize co-occurrence counts for catenae of different lengths. This results in two additional features for every co-occurrence feature type listed in Table 6.4 (and weighted information gain - see IR performance predictors). The two normalizations are a factor $|c|^{|c|}$, where $|c|$ is the number of words in a catena $c$ (Hagen et al., 2011), and the average feature value over all pairwise word combinations in $c$. In addition, frequencies of catenae in both Wikipedia and a commercial query log are computed using exact and partial matches, resulting in two counts instead of one.

| | Feature Classes | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | E-D-CI | | | E-D | | | E-CI | | | D-CI | | |
| | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 |
| *Human annotation targets* | | | | | | | | | | | | |
| Robust04 | 86.2 | 72.8 | 78.9 | 83.5 | 67.4 | 74.6 | 86.2 | 71.7 | 78.2 | 86.2 | 72.0 | 78.4 |
| WT10G | 79.3 | 67.1 | 72.6 | 76.9 | 59.7 | 67.1 | 77.2 | 65.6 | 70.8 | 79.6 | 66.1 | 72.1 |
| GOV2 | 77.0 | 68.0 | 72.1 | 70.9 | 61.8 | 65.8 | 70.9 | 61.8 | 65.8 | 75.5 | 67.2 | 70.9 |
| *MAP improvement targets* | | | | | | | | | | | | |
| Robust04 | 81.9 | 71.5 | 76.3 | 81.3 | 69.8 | 75.0 | 82.1 | 71.1 | 76.2 | 82.5 | 71.5 | 76.6 |
| WT10G | 81.1 | 63.5 | 70.7 | 81.5 | 65.4 | 72.1 | 82.6 | 63.9 | 71.8 | 79.8 | 62.8 | 70.1 |
| GOV2 | 81.1 | 63.0 | 70.7 | 79.5 | 62.6 | 67.0 | 80.7 | 64.5 | 71.4 | 80.5 | 62.9 | 70.3 |

Table 6.5: Average classifier precision (Pr), recall (R) and F1 score over 10 folds. Performance is roughly equivalent with or without ellipsis features.

4. **IR performance predictors:** Collapsed catenae take the same form as typical IR search terms. For this reason, the feature set includes predictors of IR effectiveness previously applied to IR terms. Specifically, I use the number of words in a catena and the weighted information gain ($WIG$) (Zhou and Croft, 2007).

In general, path and co-occurrence features are similar to those applied by Surdeanu et al. (2011) but the features in this work do not require document parsing. They are also similar to the path features proposed by Song et al. (2008b), but are more efficient and suited to units of variable length. Co-occurrence and IR features reflect those used by Bendersky and Croft (2008) (see also (Bendersky et al., 2010, 2011)). Ellipsis features have not been used before. Substantial work has been done on the identification of existing ellipsis in parsing (Collins, 1999), but to my knowledge there is no comparable practical work on identification of candidate text for ellipsis.

### 6.3.4 Results

Average classification precision and recall is shown in Table 6.5 for combinations of 8 ellipsis candidate features (E), 8 dependency path features (D), 36 co-occurrence features (C) and 4 IR performance prediction features (I). Features in C and I are grouped together to form the largest and most important class (CI) for the prediction of informative catenae with 71% of all proposed features. Classifier results are reported for the best model that does not use ellipsis features (D-CI) and ellipsis features with D and CI, both separately (E-D and E-CI) and together (E-D-CI). Classifier performance using all features (E-D-CI) was roughly equivalent to, or marginally better than, performance using other feature combinations where features were grouped by class (E,

D, C or I). Both the E-D-CI and the D-CI filters are used in subsequent experiments to determine whether the addition of ellipsis features improves term selection. Further ablation studies focused on individual features were not explored as they were not warranted by search performance (see Section 6.4.2).

Catenae were predicted for all description topics. Predictions are more accurate for Robust04 than the other two collections. One potential explanation is that Robust04 topics are longer on average (up to 32 content words per topic, compared to up to 16 words for topics of other collections) so they generate a more diverse set of catenae for which informativeness is more easily distinguished. The proportion of training data specific to a retrieval collection may also be a factor. Longer topics produce a greater number of catenae, so less training data from other collections is required.

## 6.4 Feature evaluation

### 6.4.1 Methodology

Evaluation compares topics reformulated using catenae selected by various discrimination methods. Discrimination of informative subsets of catenae is achieved using one of five filters: three heuristic filters based on prior work with dependency paths and two supervised classifiers, one with and one without ellipsis features. Examples of the terms selected by the baseline filters are given in Table 6.2. The filters are:

- **NomEnd:** Catenae starting and ending with nouns, or containing only one word that is a noun. Paths between nouns are used by Lin and Pantel (2001).
- **SemRol:** Catenae in which all component words are either predicates or argument heads. This is based on work that uses paths between head nouns and verbs (Shen et al., 2005), paths between certain semantic roles (Moschitti, 2008), and all dependency paths that do not occur between words in the same base chunk (e.g. words in the same noun / verb phrase) (Cui et al., 2005).
- **GovDep:** Cantenae containing words with a governor-dependent relation. Many IR models use this form of path filtering (Gao et al., 2004; Wang et al., 2007). Relations are 'collapsed' by removing stopwords to reduce the distance between content nodes in a dependency graph.
- **Feat:** Catenae selected by a decision tree using the proposed features of dependency paths, word co-occurrence and IR performance prediction.

- **SFeat:** Catenae selected by a decision tree using the proposed features of dependency paths, word co-occurrence and IR performance prediction plus semantically motivated ellipsis features.

All query models are implemented using the Indri retrieval engine version 4.12 and evaluated using three TREC collections: Robust04, WT10G, and GOV2. Baselines are a unigram query likelihood (QL) model and the highly competitive sequential dependence (SD) variant of the Markov random field (MRF) model (Bendersky and Croft, 2008; Metzler and Croft, 2005; Park et al., 2011; Xue et al., 2010). SD uses a linear combination of three cliques of terms, where the first clique contains individual words (QL), and the second and third cliques contain query bigrams (see Section 2.2.4). Bigrams represent an obvious, effective alternative to catenae for term selection in IR. The SD model for the topic '*new york city*' in Indri query language takes the form:

#weight(
$\lambda_1$ #combine(new york city)
$\lambda_2$ #combine( #ow1(new york) #ow1(york city))
$\lambda_3$ #combine( #uw8(new york) #uw8(york city))

Clique weights are stable when optimized across different collections and are set at $\lambda_1 = 0.85$, $\lambda_2 = 0.1$ and $\lambda_3 = 0.05$. The query reformulation model for catenae uses the same format as SD, but the second and third cliques contain filtered catenae instead of query bigrams. In addition, because catenae may be multi-word units, the unordered window size is adjusted to $4 * |c|$ where $|c|$ is the number of words in a catena. For example, if two catenae, '*york*' and '*new york city*', are identified for a topic '*new york city*', the reformulated query in Indri query language takes the form:

#weight(
0.85 #combine(new york city)
0.1 #combine( york #ow1(new york city))
0.05 #combine( york #uw12(new york city))

This topic representation explicitly indicates word associations while maintaining efficient and flexible matching of catenae in documents. Moreover, it does not use dependency relations between words during retrieval, so there is no need to parse a collection.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Pr | MAP | R-Pr | MAP | R-Pr |
| QL | 25.25 | 28.69 | 19.55 | 22.77 | 25.77 | 31.26 |
| SD | 26.57† | 30.02† | 20.63 | 24.31† | 28.00† | 33.30† |
| *Human annotation targets* | | | | | | |
| NomEnd | 25.91† | 29.35‡ | 20.81† | 24.27† | 27.41† | 32.94† |
| GovDep | 26.26† | 29.63† | 21.06 | 24.23† | 27.87† | 33.51† |
| SemRol | 25.70† | 29.06 | 19.78 | 22.93 | 26.76 | 32.49† |
| Feat | 26.80 | **30.20** | 21.24 | 24.42 | 28.30 | 33.51 |
| SFeat | **27.04**† | 30.11† | 20.84† | 24.31† | 28.43† | 33.84† |
| SF-12 | 27.03† | **30.20**† | **21.62**† | **24.81**† | **28.57**† | **34.01**† |
| Gold | 27.92‡ | 31.15‡ | 22.56‡ | 25.69† | 29.65‡ | 35.08‡ |
| *MAP improvement targets* | | | | | | |
| NomEnd | **26.63**† | **30.02**† | **21.76**† | **24.73**† | 27.46† | 32.60† |
| GovDep | 25.25 | 29.01 | 20.57 | 23.61 | 26.78 | 32.10 |
| SemRol | 25.61 | 29.09 | 19.74 | 23.37 | 26.63 | 31.96 |
| Feat | 26.35† | 29.69† | 21.03† | 24.42† | 27.47† | 33.29† |
| SFeat | 26.18† | 29.56† | 20.80† | 23.95† | **28.01**† | **33.44**† |
| SF-12 | 25.87 | 29.00 | 21.50† | 24.45† | 27.03† | 32.31† |
| Gold | 29.33‡ | 32.02‡ | 24.38‡ | 26.32‡ | 31.97‡ | 36.84‡ |

Table 6.6: IR effectiveness using filtered catenae is improved by a supervised classifier but there is no significant improvement from the addition of induced features of ellipsis. Best results with selected catenae shown in bold are not close to oracle performance (Gold). Significance($p < .05$) shown compared to QL (†) and SD (‡).

### 6.4.2 Results

The effectiveness of ellipsis features for IR are evaluated in two ways. First, the performance of topics reformulated using catenae selected by different filters and training against alternative target labels (MAP improvement and human annotation) are compared to the performance of SD and QL. This contrasts the proposed approach with previous work. Second, topics are compared with catenae filtered using oracle knowledge. This explores the impact of catenae classification on subsequent query effectiveness, and establishes an upper bound on performance given the training data.

Results in Table 6.6 show that catenae selection using the supervised classifier is not significantly better than a sequential dependence (SD) model using simple word adjacency. There is also no significant difference between the results for the supervised classifier with and without ellipsis features. Given these observations, trends in the data suggest that best performance is achieved with catenae containing only 1-2 words (SF-12) trained against human annotation targets ($p = 0.40$, $p = 0.17$ and $p = 0.08$ for MAP compared to SD for Robust04, WT10G and GOV2 respectively). The relatively strong

performance of this SF-12 model suggests that most benefit derived from catenae is accounted for by governor-dependent and single word units. The contribution of single words can be inferred from comparison of results for SF-12 and GovDep (2-word catenae). Overall, changes are small and fairly robust, with one half to two thirds of all topics showing less than 10% change in MAP.

Ellipsis features also appear to be more effective when used in combination with human annotation targets. There is highly marginal improvement for SFeat compared to Feat for all collections when using human annotation targets, while the opposite trend is observed for SFeat trained against binned MAP improvement (in each case, effectiveness is slightly better or worse on two of three collections). In general, training against human annotation tends to result in the selection of terms that are more discriminative, but the difference is not significant. Manual review of queries suggests that this may be due to the influence of multiword terms that combine one highly discriminative word with other less discriminative words. Such terms can improve MAP but are less effective than terms with several highly discriminative words. It appears that even if humans are uncertain about the discrimination ability of terms, they are likely to choose robust terms that provide a reasonable signal for classifier training.

Filters that do not use a supervised classifier tend to decrease performance compared to SD. Governor-dependent relations for WT10G are an exception and I speculate that this is due to their ability to overcome a negative influence of 3-word catenae for this collection. All other discrimination methods permit 3-word catenae in queries, while governor-dependent relations permit only 2-word combinations. Manual inspection suggests that WT10G topics are short and have relatively simple syntactic structure (e.g. few prepositional phrase attachment ambiguities). This means that 3-word catenae tend to include uninformative words, such as '*reasons*' in '*fasting religious reasons*'. In contrast, 3-word catenae in other collections tend to identify topic subconcepts that operate over prepositional phrases, such as '*science plants water*' ('*the science of growing plants in water*').

The intuition that governor-dependent terms are successful because they exclude 3-word catenae is confirmed by classification results for catenae separated by length. A separate classifier was trained on examples of catenae with the same lengths (1, 2 or 3 words) using human annotation targets. The rejection rate for 3-word catenae was twice as high for WT10G as for other collections. In addition, it was more difficult to distinguish discriminative 3-word catenae than discriminative catenae with 1-2 words.

With respect to the impact of classifier accuracy on IR effectiveness, performance

of queries generated using a classifier were compared to the performance of queries using oracle knowledge of catenae annotations and effect on MAP (Gold). The effectiveness of term selection is limited by the accuracy of the classifier used to identify discriminative catenae, and the results for all models are significantly short of the performance for oracle queries (e.g. $p = 0.00001$, $p = 0.02$ and $p = 0.001$ for SF-12 compared to Gold using human annotated targets for Robust04, WT10G and GOV2 respectively). It may be that the training signal is not strong enough to show a clear effect of semantically motivated features, but the results tend to confirm a lack of association between semantics and discrimination as reported in Chapter 5.

Finally, a review of selected catenae for queries that perform substantially better or worse than SD ($> 75\%$ change in MAP) suggests that the best IR effectiveness occurs when selected catenae clearly focus on the most important aspect of a topic. Poor performance is caused either by a lack of focus in a catenae set, even when selected catenae seem reasonable from a semantic perspective, or an emphasis on words that are not central to the topic. The latter can result when words that are not essential to topic semantics appear in many catenae due to their position in a dependency graph.

## 6.5 Conclusion

In this Chapter, I explored the hypothesis that for natural language description topics, induced features of ellipsis help to identify query terms that improve search precision and recall. To support this approach, catenae were applied and selected using a supervised classifier with a comprehensive set of features, including novel semantically motivated features specific to paths, features of language structure and typical IR features such as word and term frequencies. Catenae were used in part because they have been proposed as a precondition for the identification of salient text units in ellipsis (Osborne and Groß, 2012). The reported approach aimed to identify discriminative query terms with a strong semantic alignment.

Query reformulation using catenae with a supervised selection technique incorporating all features performed consistently well on three diverse collections but did not significantly improve over the effectiveness of a robust sequential dependence (SD) model. There was also no significant difference between the results for the supervised classifier with and without ellipsis features. It is encouraging that the proposed features of ellipsis did not decrease IR effectiveness (and in preliminary experiments, manually annotated candidate antecedents predicted terms that improve NDCG with 73% pre-

cision), but performance gains were not commensurate with the amount of additional processing required.

A secondary consideration is the utility of catenae as a flexible implementation of dependency paths that do not require dependency parsing a retrieval collection. It is not possible to directly compare performance of the reported approach with ad hoc techniques for IR that parse a collection because the structures of the retrieval indexes are incompatible. However, a recent result using topic translation based on dependency paths (Park et al., 2011) reports 14% improvement over query likelihood (QL). The approach reported in this Chapter achieves 7% improvement over QL on the same collection. It appears that catenae modelled in documents by word proximity are not a substitute for path-based techniques, but may offer some insight into their application. They can also have particular value when it is not practical to parse target documents to determine text similarity.

Overall, these experiments highlight the difficulty of extracting semantic relationships by syntactic methods and suggest that word associations that align with request semantics do not necessarily have a strong relationship with the discriminative ability of terms. Finally, they indicate the importance of selecting query terms that form a cohesive set. In the next Chapter, I explore an alternative approach to term selection that significantly improves IR effectiveness using simpler and fewer features to select a focused set of interpretable query terms.

# 7

# Term Selection Using Topically Related Text

In this Chapter, I propose a novel term ranking algorithm called PhRank (phrase rank) that uses no explicit linguistic knowledge and takes nterms (random combinations of words in a request) as term candidates. This contrasts with the term selection method in Chapter 6 that took a semantically motivated approach.

Evaluation of PhRank shows that queries with a few precise terms selected on the basis of word co-occurrence can be more effective than highly competitive IR models that use up to 30 terms, even when the latter are optimized using a linguistically diverse set of features. This is briefly illustrated in Table 7.1 for GOV2 topic #756. Terms selected by three top performing IR models are shown. The sequential dependence (SD) model uses bigram word associations and is straightforward and robust (Metzler and Croft, 2005). The key concept (KC) model (Bendersky and Croft, 2008) aims at a highly succinct representation using noun phrases filtered by a method that uses no syntactic features. The subset distribution (SDist) model (Xue et al., 2010) learns terms and weight variables using a diverse set of features from various linguistic theories. It optimizes the selection of ten best-performing subqueries, the weights for those subqueries, and interpolation weights between the original query and the distribution of subqueries on a per query basis. In addition, it is biased towards longer terms with 3-6 words. As shown, terms selected by PhRank can be much more precise. One to five terms selected by PhRank in an unweighted model can deliver up to 14% performance improvement compared to these highly competitive alternatives.

PhRank extends work on Markov chain (random walk) frameworks for query expansion and has three core elements. First, it uses pseudo relevance feedback to select

**Query:** *Locations of volcanic activity which occurred within the present day boundaries of the U.S. and its territories.*

| PhRank | Sequential Dependence |
|---|---|
| volcanic<br>volcanic boundaries<br>volcanic territories<br>volcanic activity<br>volcanic occurred | locations volcanic<br>volcanic activity<br>activity which<br>which occurred<br>occurred within<br>within present<br>present day<br>day boundaries<br>boundaries us<br>us territories |

| Key Concept | Subset Distribution |
|---|---|
| present day boundaries<br>volcanic activity | volcanic day boundaries<br>day boundaries territories<br>volcanic activity occurred day boundaries<br>present day boundaries<br>volcanic boundaries territories<br>volcanic activity occurred<br>activity occurred day boundaries<br>volcanic activity occurred boundaries<br>volcanic present day boundaries<br>volcanic occurred boundaries<br>+ 20 bigrams (*if weights collapsed*) |

Table 7.1: Terms selected by four highly effective query reformulation models for TREC GOV2 topic #756. PhRank queries are more precise.

compact and focused terms from within a query itself, rather than expansion terms that are not in a query. Second, it captures query context with an affinity graph constructed using word co-occurrence in pseudo-relevant documents. Third, it combines a random walk of the graph with discriminative weights to rank candidate terms. The top-ranked term candidates are applied in a query reformulation.

To my knowledge, it is the first method to use pseudo relevance feedback for in-query term selection. Pseudo relevance feedback has previously been used either to select terms that are not in a query (see e.g. (Lavrenko, 2004)) or to weight individual query words without selection (Croft and Harper, 1979; Ruthven and Lalmas, 2003). It has also been used to smooth language model representations of queries without term selection (Cai et al., 2007b; Hoenkamp et al., 2009). In contrast, approaches to in-query term selection have used highly localized *word context*, in the form of syntactic relations and co-occurrence, and *global context* in the retrieval collection. They do not consider *query context*, such as a topic identified from pseudo relevant documents.

This Chapter presents PhRank and reports on evaluation experiments. It begins with principles for term selection that ground the approach. This is followed by background on Markov chain frameworks for query reformulation, a formal description of the PhRank algorithm, and a presentation of the evaluation framework and empirical results. It concludes with a summary of results and directions for further improvement.

## 7.1 Principles for term selection

PhRank is grounded by four principles for word and term informativeness. These principles are proposed here for the first time:

**An informative word:**

1. Is informative relative to a query
2. Is related to other informative words

**An informative term:**

3. Contains informative words
4. Is discriminative in the retrieval collection

It is evident that words should accurately represent the meaning of a query. A novel aspect of PhRank is how it considers the informativeness of words relative to a query. Queries do not provide much context so PhRank uses pseudo relevant documents to better represent an information need. Pseudo relevance is an established means of enhancing query representation (Rocchio, 1971). In addition, query context is represented as an affinity, or co-occurrence, graph in which the semantic significance of a word is correlated with the degree of its corresponding vertex (Ferrer i Cancho and Solé, 2001). By consequence, the form of the graph may help to estimate the semantic significance of words. Finally, PhRank uses a random walk of the affinity graph. This reinforces words that capture query 'essence' more strongly than words that are peripheral to query meaning, making informative words more readily apparent.

The second principle holds that an informative word is related to other informative words. The Association Hypothesis (van Rijsbergen, 1979b) also states that, "if one index term is good at discriminating relevant from non-relevant documents, then any closely associated index term is also likely to be good at this". PhRank implements this hypothesis using a Markov chain framework. A random walk of an affinity graph determines the informativeness of a word *i* by the informativeness of other words connected to *i*, and the number of words connected to *i*. This abstracts away from specific word associations that can impose unnecessary limits on word combinability (see discussion in Chapters 3, 4 and 6). It also has an ability to capture lexical relations that cannot be detected by ngrams or syntax but are implicit in contextual relations. Finally, the graph captures aspects of both dependency and syntagmatic word associations. A co-occurrence graph that defines edges by word proximity has the same form as a global

dependency graph that defines edges using governor-dependent relations (see Section 4.3.1.2). Given this pseudo-equivalence, a co-occurrence graph is preferred because it avoids the computational overhead of document parsing.

With respect to informative terms, I expect that they contain informative words. Consider a base case in which a term has only one word. It is obvious that this term must also display the properties of an informative word. By extrapolation, I assume that all terms must contain informative words. PhRank incorporates knowledge of word informativeness in term ranking by averaging weighted word scores.

Finally, an informative term must be discriminative in a retrieval collection. Analysis in Chapter 5 shows the importance of discrimination for IR, and Chapter 6 confirms the effectiveness of frequency-based features for discrimination. PhRank weights terms with a normalized *tf.idf* inspired weight.

The PhRank implementation of these four principles for term selection results in significant gains in IR effectiveness compared to highly effective baselines. Moreover, gains are achieved using a small number of compact terms selected on the basis of word co-occurrence features. Co-occurrence features are not new, so the power of PhRank lies in the use of query context and graph analysis. An affinity graph flexibly captures many word associations and leverages these to determine query word informativeness. The next Section presents the Markov chain framework used for this purpose.

## 7.2 Markov chain frameworks for query reformulation

Markov chain frameworks and spreading activation networks are well-studied in IR with origins in associative word networks (Crestani, 1997) and webpage authority (Page et al., 1999). They have previously been used in a principled way to smooth and expand queries in a language modeling framework (Collins-Thompson and Callan, 2005; Huang et al., 2010; Lafferty and Zhai, 2001; Mei et al., 2008; Zhai and Lafferty, 2001) but are novel for *unexpanded* term selection.

A Markov chain framework uses the stationary distribution of a random walk over an affinity graph $G$ to estimate the importance of vertices in $G$. Vertices can represent words, in which case edges represent word associations. If the random walk is ergodic, affinity scores at vertices converge to a stationary distribution that can be used to establish a ranking over words.

A random walk describes a succession of random or semi-random steps between vertices $v_i$ and $v_j$ in $G$. Let $\ell_{ij}$ be a transition probability (or edge weight) between $v_i$

and $v_j$. The path of the walk is determined by a square probability matrix $H = (h_{ij})$ with size $n$, where $n$ is the number of unique vertices in $G$. The probability $h_{ij} = \ell_{ij}$ if $v_i$ and $v_j$ are connected, and $h_{ij} = 0$ otherwise. Affinity scores are computed recursively. Let $\pi_j^t$ be the affinity score associated with $v_j$ at time $t$. Then $\pi_j^{t+1}$ is the sum of scores for each $v_i$ connected to $v_j$, weighted by the possibility of choosing $v_j$ as the next step on the path from $v_i$:

$$\pi_j^{t+1} = \sum_i \pi_i h_{ij} \tag{7.1}$$

It is usual to introduce some minimal likelihood that a path from $v_i$ at time $t$ will randomly step to some $v_j$ at time $t+1$ that may be unconnected to $v_i$. Otherwise, clusters of vertices interfere with the propagation of weight through the graph. This likelihood is often defined to be the uniform probability vector $u = 1/n$, although any other vector can be chosen (Huang et al., 2010). A corresponding factor reflects the likelihood that a path will follow the structure of edges in $G$. A damping factor $\alpha$ controls the balance between them:

$$\pi^{t+1} = \alpha \pi^t H + (1-\alpha)u \tag{7.2}$$

A prominent example of a Markov chain framework in IR is the PageRank algorithm. In PageRank, vertices in a graph represent web pages and edges are hyperlinks between web pages. The random walk in PhRank is similar to PageRank, but operates over vertices that represent stemmed words in a document set, rather than documents in a collection. In addition, PageRank assumes a directed graph in which all edges have equal relevance while PhRank uses an undirected graph with weighted edges that reflect the variable significance of word relationships. Weights are determined using retrieval collection frequencies and could easily use another measure, such as frequencies in a commercial query log. Finally, PageRank assumes that all vertices and edges have the same origin (the web). PhRank exploits the fact that co-occurrence counts derive from multiple documents by weighting counts from a document $D$ by the probability of $D$ given a query.

Markov chain processes are also used for keyphrase extraction, a task similar to term selection.[1] TextRank (Mihalcea and Tarau, 2004) and several related algorithms (Wan and Xiao, 2008) use a random walk of an affinity graph to identify salient sequences of nouns and adjectives in documents (keyphrases that are noun phrases). TextRank forms keyphrases using only the most important words identified by this process.

[1] Keyphrases are words and word combinations that capture the main topics of a text (Turney, 1999).

The keyphrases can be precise but are generally outperformed by a *tf.idf* metric (Hasan and Ng, 2010).

SingleRank (Wan and Xiao, 2008) and ExpandRank (Wan and Xiao, 2008) improve over TextRank but still generally achieve only 30-40% task accuracy (Hasan and Ng, 2010). Performance gains are mostly due to a difference in the way phrases are scored after graph iteration (Hasan and Ng, 2010). Unlike TextRank, both algorithms consider all longest-matching sequences of nouns and adjectives to be candidate keyphrases, even if they include words of low importance. SingleRank also weights edges in a graph to reflect frequencies of word co-occurrence, and defines co-occurence using a larger window of text than TextRank. ExpandRank builds on SingleRank, using pseudo relevant documents to supplement an initial text, and weights the resulting graph with co-occurrence counts and document similarity. Results with the DUC2001 dataset show that ExpandRank improves keyphrase extraction compared to both TextRank and SingleRank. However, the DUC2001 dataset contains only 308 news articles that are concise and tightly focused on specific topics. Hence, any improvements in performance using feedback will not necessarily translate well to the open domain where there are many more documents that may also be longer or less focused (Hasan and Ng, 2010).

PhRank is more flexible and better suited to open domain IR than these keyphrase extraction algorithms. It uses multiple sources of co-occurrence evidence and the discriminative ability of terms in a collection. It also produces an unbiased ranking over terms of mixed lengths, does not rely on syntactic word categories such as nouns, and permits terms to contain words with long distance dependencies.

## 7.3 PhRank

PhRank captures query context with an affinity graph constructed from stopped, stemmed pseudo-relevant documents. Vertices in the graph represent unique stemmed words (or simply, stems). Edges connect stems that are adjacent in the processed pseudo relevant set. Graph transition probabilities (edge weights) are computed using a weighted linear combination of stem co-occurrence, the certainty that the document in which they co-occur is relevant, and the salience of sequential bigram factors in the pseudo relevant set. The edge weights thus represent the tendency for two stemmed words $w_i$ and $w_{j\neq i}$ to appear in close proximity in documents that that reflect a query topic.

Stems in the affinity graph are scored using a random walk algorithm. Following

```
k = 5
resourceList = [ C, wikipedia ]

for q in queryList:

  N = set()
  for rsc in resourceList:
    N.add( retrieve_top_k( q, rsc ) )
  N = retrieve_top_k( q, N )
  N.add( q )

  # one word type per row and column
  G = arrayStruct()
  for ( doc, docRel ) in N:
    doc.stopStem()
    G.grow( buildGraph( doc, docRel ) )

  G.idfWeightEdge()              # bigram wt r
  G.normalize()
  G.iterate()
  G.weightVertex()                # word wt s

  T = q.terms
  for term in q:
    term.wt = G.score( term )
    term.wt *= term.globalWt( C ) # term wt z
  T.sortByWeight()
```

```
def buildGraph( doc, docRel ):
  docG = index( doc )
  docG.linearWt( uw2, uw10 )
  docG.weight( docRel )

  return docG


def score( term ):
  S = 0

  for w in term.wordSplit():
    S += self.affinityScore( term )

  return S /= term.length()


def globalWt( C ):
  l = self.length()
  wt = C.tfidf( self ) * l^l

  return wt
```

Figure 7.1: Pseudocode for the PhRank algorithm.

convergence, stem scores are weighted by a *tf.idf* style weight that further captures salience in the pseudo relevant set. This aims to compensate for potential undesirable properties of the random walk. Finally, term scores are computed using the average score for stemmed words in a term, weighted by term salience in the retrieval collection. The $m$ highest scoring terms are employed to reformulate $Q$. Pseudo code for the algorithm is shown in Figure 7.1. The rest of this Section describes the algorithm in more detail, including three heuristic weights (factors $r$, $s$ and $z$). A number of choices are possible for these factors and specific choices are analyzed in Section 7.5.1.

### 7.3.1 The PhRank algorithm

1. **Graph construction:** Let a query $Q = \{w_1, ...w_n\}$ and $C$ be a document collection. The top $k$ documents retrieved from $C$ using $Q$ are assumed to describe a similar topic to $Q$. I define $C$ to be the retrieval collection plus English Wikipedia but also explore the effectiveness of the retrieval collection alone. Wikipedia is included because it improves IR results for query expansion using a random walk (Collins-Thompson and Callan, 2005). The top $k$ documents in $C$, together with $Q$ itself encoded as a short document $d_0$, comprise *neighboring* documents in the neighborhood set $N = \{d_0, ....d_k\}$.

Documents in $N$ are stopped using a minimal list of 18 words (Manning et al., 2008) and stemmed using the Krovetz stemmer. This improves co-occurrence counts for content-bearing stems and reduces the size of an affinity graph $G$ constructed from the processed documents. Stoplisting with a longer list hurt IR effectiveness. Edges in $G$ connect stemmed words $i$ and $j$ at vertices $v_i$ and $v_j$ if $i$ and $j$ are adjacent in $N$. Documents in $N$ with only one word (e.g. some queries since queries are included in $N$) are discarded to ensure that all vertices have at least one connecting edge.

2. **Edge weights:** Transition probabilities (edge weights) $\ell_{ij}$ are based on a weighted linear combination of the number of times $i$ and $j$ co-occur in windows $W$ of size 2 and 10. This is motivated by the idea that different degrees of proximity provide rich evidence for word relatedness in IR (Metzler and Croft, 2005; Mihalcea and Tarau, 2004; Wan and Xiao, 2008). Edge weights are defined by:

$$\ell_{ij} = r \cdot \sum_{d_k \in N} p(d_k|Q)(\lambda c_{ij_{W_2}} + (1-\lambda)c_{ij_{W_{10}}})$$

where $p(d_k|Q)$ is the probability of a document in which the stems $i$ and $j$ co-occur given $Q$ (Lavrenko, 2004), and $c_{ij_{W_2}}$ and $c_{ij_{W_{10}}}$ are the counts of stem co-occurrence in windows of size 2 and 10 in $N$. $\lambda$ is set to 0.6. The relevance of $d_0$ to $Q$ is set to be high but reasonable (-4 for Indri log likelihood scores). The exact setting has very little effect on term ranking.

Factor $r$ is a *tf.idf* style weight that confirms the importance of a connection between $i$ and $j$ in $N$. $G$ includes many stemmed words, so unweighted affinity scores can be influenced by co-occurrences between highly frequent, but possibly uninformative, stems such as '*make*'. Factor $r$ minimizes this effect. Since the *tf* component is already accounted for by $\lambda c_{ij_{W_2}} + (1-\lambda)c_{ij_{W_{10}}}$, $r$ is reduced to an *idf* style component:

$$r_{ij} = log_2 \frac{\sum_{ij \in N} c_{ij_{W_2}}}{1 + c_{ij_{W_2}}}$$

3. **Random Walk:** A random walk of $G$ follows the standard Markov chain framework presented in Section 7.2. Edge weights are normalized to sum to one and $\pi_j$ is the affinity score of the stem associated with $v_j$. $\pi_j$ indicates the importance of a stem in the query context. $\pi_j$ is initialized to 1 and iteration of the walk ceases when the difference in score at any vertex does not exceed 0.0001.

This translates to around 15 iterations but may be optimized for efficiency. The damping factor $\alpha = 0.85$ is equivalent to a walk along five connected edges in $G$ before the algorithm randomly skips to a possibly unrelated vertex. The average sentence length in English is around 11-15 words so this equates to skipping at or near the boundary of a sentence around one half of the time.

4. **Vertex weights:** Following the random walk, stemmed words in $G$ are further weighted to capture both the *exhaustiveness* with which they represent a query, and their global *saliency* in the collection (Spärck Jones, 1972). Exhaustiveness indicates whether a word $w_1$ is a sufficient representation of the query. If $w_1$ appears many times in $N$ then it is less likely that a term $x$ containing $w_1$ will benefit from additional words $w_2 ... w_n$. For example, the term *geysers* quite exhaustively represents the TREC query #840, '*Give the definition, locations, or characteristics of geysers*'. A term containing additional words, e.g. *definition geysers*, is not more informative. However, common stems, such as '*definition*', tend to have high affinity scores because they co-occur with many words.

   Factor $s$ balances exhaustivity with global saliency to identify stems that are poor discriminators between relevant and non-relevant documents. Specifically, $s_{w_n} = wnf_{avg} * idf_{w_n}$, where $wnf_{avg}$ is the frequency of a word $w_n$ in $N$, averaged over $k+1$ documents (the average frequency) and normalized by the maximum average frequency of any term in $N$. As usual, $idf_{w_n}$ is the inverse document frequency of $w_n$ in the collection, so $idf_{w_n} = log_2 \frac{|D|}{1+df_{w_n}}$ where $|D|$ is the number of documents in the collection $C$, and $df_{w_n}$ is the number of documents in $C$ containing $w_n$. An advantage of factor $s$ is that it enables PhRank to be independent of an IR model. A model may treat the component words of terms as independent or dependent. Factor $s$ helps to ensure that the selected terms are discriminative irrespective of this representation.

5. **Term ranking:** To avoid a bias towards longer terms, a term $x$ is scored by averaging the affinity scores for its component words $\{w_1, [...w_n]\}$. Term rank is determined by the average score multiplied by a factor $z_x$ that represents the degree to which a term is discriminative in a collection:

$$z_x = f_{x_e} * idf_{x_e} * l_x$$

   Let $x_e$ be a proximity expression such that the component words of $x$ appear in an unordered window of size $W = 4$ per word. Thus, a term with two words

appears in an 8-word window, and a term with three words appears in a 12-word window. The frequency of $x_e$ in $C$ is $f_{x_e}$ and $idf_{x_e}$ is defined analogously to $idf_{w_n}$ above. $l_x$ is an exponential weighting factor proposed for the normalization of ngram frequencies during query segmentation (Hagen et al., 2011). This factor favors longer ngrams that tend to occur less frequently in text. Multiplication of ngram counts by $l_x$ enables comparison of counts for terms of varying length. Let $|x|$ be the number of words in $x$, then $l_x = |x|^{|x|}$.

In summary, the PhRank algorithm describes the informativeness of a term $x$ for a query $Q$ compared to other terms. This is computed using the function:

$$f(x,Q) \stackrel{rank}{=} z_x * \frac{\sum_{w_n \in x} \pi_{w_n}}{n} \tag{7.3}$$

### 7.3.2 Diversity filter

PhRank uses an average word affinity score so it often assigns a high rank to multi-word terms that contain only one highly informative word. On the one hand this is desirable. Informative terms can contain words that are uninformative individually. For example, given a query about '*the destruction of Pan Am Flight 103 over Lockerbie, Scotland*' (adapted from Robust04 #409), the term '*pan flight 103*' is informative even if the polysemous word '*pan*' is uninformative by itself. On the other hand, it can result in low diversity of top ranked terms. Every term containing a particularly informative word will be ranked above terms not containing that word. For this reason, a simple, heuristic filtering technique with top-down constraints is used to increase diversity.

Given a ranked list, all terms with a score of zero are discarded. Starting with the second most highly ranked term $x_n$ and iterating through the list until $k$ top terms have been checked, $x_n$ is checked against the list of terms with a higher rank $x_{n<m}$. Let $A$ be the set of component words in $x_n$ and $B$ be the set of component words in any single term $x_{m>n}$. If $A \subset B$, or every component word of $x_n$ is contained in at least one $x_{m \neq n}$, then $x_n$ is discarded. For example, if $x_n =$'*birth rate*' and there is some $x_{m<n} =$'*birth rate china*' then the filter discards $x_n$ on the assumption that the longer term better represents the information need. If $x_n =$'*declining birth rate*' and there is some $x_{m<n} =$'*birth rate*' and some $x_{m<n} =$'*declining birth*' then the filter discards $x_n$ on the assumption that the shorter terms better represent the information need and the longer term is redundant. This process ensures that no vital information is lost, but clearly presents an opportunity for further improvement.

## 7.4 Evaluation framework

### 7.4.1 Baseline models

There are two aspects to evaluation: the effectiveness of term selection and the impact of applied pseudo relevance feedback.

#### 7.4.1.1 Models for term selection

Evaluation of PhRank term selection compares queries reformulated using top-ranked PhRank terms with three existing IR models. These models are selected to be robust, precise, succinct and highly competitive. Comparison is also made with the query likelihood (QL) model even though QL does not select terms. This is because all models include a query likelihood component. The baselines for term selection are:

- **Sequential dependence model:** Given that evaluation applies across three TREC collections and uses both description topics and title queries, the sequential dependence model (SD) (Metzler and Croft, 2005) provides a highly effective *robust* baseline. This baseline is also used for comparison in related work (see Sections 2.2.4 and 6.4.1) (Bendersky and Croft, 2008; Park et al., 2011; Xue et al., 2010). SD queries combine simple unigrams and bigrams with no weighting, so they are very easy to generate. Highly effective weighted variants have also been developed (Bendersky and Croft, 2012; Park et al., 2011; Xue et al., 2010).
- **Subset distribution model:** To my knowledge, the subset distribution model (sDist) (Xue et al., 2010) has the highest reported mean average *precision* that is relevant to a discussion of term selection without query expansion or application of higher order dependencies (Bendersky and Croft, 2012). This competitive performance is achieved by jointly optimizing over possible subqueries and subquery weights using a wide variety of syntactic and statistical features. Features include: ngrams, noun phrases, part-of-speech tags, predicate-argument relations, named entities, dependency tree features, mutual information, IR performance predictors and more. sDist optimizes weights for ten subqueries, where a subquery is a linear combination of a default SD query and one term treated as a bag-of-words. sDist is the most effective model for stringent comparison that ensures real progress has been made. For this reason, it is included in the evaluation even though queries for Robust04 are not available from the authors.

- **Key concept model:** The Key Concept model (KC) (Bendersky and Croft, 2008) is a *succinct* yet competitive model that selects two terms. Comparison with KC evaluates the performance of short queries, in contrast with the distributed queries used by SD and sDist. Distributed queries typically contain many terms, and these terms contain words from the full length of a query in order to ensure complete representation of an information need. In contrast, succinct queries aim to eliminate many words from a query in order to focus on a core information need. KC is a weighted linear feature model that combines two cliques. The first clique ($\lambda_1 = 0.8$) contains individual words from the original query (QL), and the second clique ($\lambda_2 = 0.2$) combines a weighted bag-of-words representation for each of two weighted noun phrases. These noun phrases are selected using a decision tree with features of word co-occurrence. The model reduces to a weighted representation of the original query with word independence. If '*city*' and '*new york*' are the top two terms, it takes the following form in Indri query language, where $\delta_n$ is the decision tree confidence score associated with a term:

  #weight(
  $\lambda_1$ #combine(new york city)
  $\lambda_2$ #weight( $\delta_1$ #combine(new york) $\delta_2$ city ))

#### 7.4.1.2 Applied pseudo relevance feedback

The contribution of pseudo relevance feedback to PhRank effectiveness is assessed by comparing the performance of queries using PhRank terms with the performance of two alternative models. The Relevance Model (RM) (Lavrenko, 2004) uses pseudo relevance feedback to expand queries and the Croft and Harper (1979) model uses feedback to weight query terms without expansion.

- **Relevance model:** The RM (Lavrenko, 2004) approximates relevant and non-relevant classes of documents using pseudo relevance feedback and generates a language model representation of relevance from the pseudo relevant set (see Section 2.2.2.2). Expansion words are generated by the language model, so the technique relies only on word co-occurrence frequencies. If IR models are classified on the basis of how many passes they require over a data collection, the RM is comparable to models that use top PhRank terms (assuming PhRank does not incorporate additional resources such as Wikipedia). Conversely, if models are

classified on the basis of techniques they apply, then the RM is not comparable to PhRank because it expands queries whereas PhRank does not. Expansion can improve performance but results in longer queries. RM queries are considerably slower to run than PhRank queries, with up to 1000 additional terms required for optimal performance. For a query '*#combine(new york city)*' used to retrieve pseudo relevant documents, a RM query has the following format, where word weights $\delta_n$ represent the probability of a word given the query:

#weight(
$\lambda_1$ #combine(new york city)
$\lambda_2$ #weight( $\delta_1 word_1$ $\delta_2 word_2 ...$ $\delta_n word_n$ ))

Five parameters can be tuned for a RM: (1) the smoothing on initial retrieval; (2) the amount of interpolation between the RM and the initial query model; (3) the number of feedback documents used to determine expansion words; (4) the number of expansion words used in the RM; and (5) the smoothing on the final retrieval. In addition, the form of the initial query model must be selected.

The initial query model was chosen to be the sequential dependence model (SD). Smoothing in the initial retrieval was set at $\mu = 2500$ as for SD queries. This is the same smoothing used for final retrieval since the parameters are tied in the Indri retrieval engine. The amount of interpolation between the original query model and the expansion model was set independently for description topics and title queries of each collection following a grid parameter search on the interval [0.05, 0.95] with increments of 0.05. The weights assigned to the original query models for Robust04, WT10G and GOV2 respectively were 0.1, 0.3 and 0.2 for title queries and 0.15, 0.3 and 0.3 for description topics. The number of feedback documents was also set independently for each collection and query type, although the optimal number required for each collection was stable for title queries and description topics. The settings were 10, 5, and 30 documents respectively for Robust04, WT10G and GOV2.[2] Note that the number of feedback documents was not optimized for PhRank which may bias results slightly in favour of the RM. PhRank uses 5 feedback documents for all collections. The number of expansion words was set at 200 for both description topics and title

[2]Note that these numbers are lower than typically expected for the RM due to the use of an SD query for initial retrieval rather than query likelihood (QL). The SD model achieves significantly higher precision than QL.

queries following a grid search on Robust04 title queries and description topics using the values [5, 10, 20, 30, 40, 50, 100, 200, 300, 400, 500]. This setting was optimal for both query types. It was not tuned specifically for other collections due to the very long running time of RM queries on larger collections when using more than 100 expansion words. Finally, no smoothing was used on the relevance model itself. This setting was selected following a grid search on the interval $\mu = [0, 3500]$ with increments of 500 for all collections and query types.

- **Croft and Harper model:** The CH model (Croft and Harper, 1979) is similar to models that include PhRank terms because it does not perform query expansion. Instead, it uses word frequencies in pseudo relevant documents to weight original query words. If a query '*#combine( new york city )*' is used to retrieve pseudo relevant documents, CH takes the form in Indri query language:

  #weight($\delta_1$ new $\delta_2$ york $\delta_3$ city )

  Word weights $\delta_n$ approximately reflect a combination of a simple match to a relevant document set, and a match using inverse document frequency weights. Weights in these experiments are determined using an unsmoothed RM computed from five pseudo-relevant documents (the same number used for PhRank). Any words that do not appear in the top five documents are eliminated during reformulation unless the initial query retrieves no relevant documents (as for WT10G title queries '*nativityscenes*' and '*angioplast7*'). If no documents are retrieved by the initial query then query likelihood is applied.

In summary, evaluation of PhRank assesses the degree to which queries using top ranked PhRank terms are robust, precise and succinct, and leverage pseudo relevance feedback. The next Section describes comparison models that include PhRank terms.

### 7.4.2 PhRank models

Baseline models are matched to models that replace selected terms (i.e. bigrams, noun phrases, query subsets) with top ranked PhRank terms. This better isolates the effects of term selection from choices about query reformulation. Several of the baseline models use term weighting, and although weighted models are usually compared against each other (weighting is known to improve IR effectiveness) PhRank models are unweighted in order to more clearly demonstrate the effects of term selection alone. The PhRank models are:

- **Comparison with SD, sDist, CH:** Compared to PhRank model PR-.F (see Section 7.5.1 for an explanation of notation) that uses the same query format as SD, except the second and third cliques contain PhRank terms instead of query bigrams. In addition, because PhRank terms may be 1-3 words long, the unordered window operator is adjusted in accordance with the proposal for the full dependence variant of the MRF model (Metzler and Croft, 2005). The window size is 4 multiplied by the number of words in a term (see Section 6.4). PR-.F uses five terms for description topics and feature analysis experiments, and three terms for title queries (or less, if the required number of terms is not available after rank filtering). See Section 6.4.1 for an example of how queries are implemented in Indri using PhRank terms in place of catenae.
- **Comparison with KC:** KC is compared to model (PR-zF2). This takes the same form as KC in that the two top terms selected by PhRank are treated as bags-of-words. However, PR-zF2 does not benefit from term weights $\delta_n$.
- **Comparison with RM:** It is possible to perform both PhRank term selection and RM term expansion with a single pass over feedback documents (i.e. the process is in the same class as the RM with respect to computational efficiency). Therefore, results with RM and CH are compared to PR¬W and PR¬W2 model queries (corresponding to PR-.F and PR-zF2 respectively without use of Wikipedia), but reformulated to include both PhRank terms and RM expansion terms (PR¬We and PR¬W2e). These models are referred to as PR¬W and PR¬W2. Note that RM only uses a retrieval collection, so Wikipedia is excluded to enable fair comparison. Essentially, these models replace the sequential dependence query in the RM with a PhRank query.[3] If two terms '*york*' and '*new york city*' are selected by PhRank, this takes the form:

  #weight(
  $\lambda_a$ #combine(new york city)
  $\lambda_b$ #combine( york #1(new york city))
  $\lambda_c$ #combine( york #uw12(new york city))
  $\lambda_d$ #weight( $\delta_1 word_1$ $\delta_2 word_2$... $\delta_n word_n$ )))

  Here, $\lambda_a + \lambda_b + \lambda_c$ for PR¬We and PR¬W2e sum to $\lambda_1$ for the RM, and $\lambda_d$ is

[3] In the original implementation of the RM, a query likelihood query is embedded in the relevance model itself, rather than being separate. However, the implementation of these models in the Indri query engine makes it is possible to extract the query expansion words alone.

equivalent to $\lambda_2$ for the RM. The expansion terms are identified using the same model parameters described for the RM, and use the same number of feedback terms. Importantly, the number of feedback documents is constant, so only one pass over these documents is required. The weight assigned to the PhRank portion of the resulting query is set using grid search on the interval [0.05, 0.95] in increments of 0.05. For the description topics of Robust04, WT10G and GOV2, the weights are respectively 0.15, 0.3 and 0.45. For title queries and the same collections the weights are respectively 0.1, 0.15 and 0.4.

## 7.5 Experiments

The performance of PhRank is evaluated in three stages. First, versions of the algorithm in which specific features are omitted are compared with each other. Second, the performance of queries reformulated using PhRank top ranked terms are compared against highly effective models for both description topics and title queries. Third, the robustness and performance error for PhRank are compared on a query by query basis against a distributed approach to term selection (SD).

The evaluation uses three TREC collections (Robust04, WT10G and GOV2, see Section 2.4) and version 4.12 of Indri with Dirichlet smoothing, $\mu = 2500$. All collections and queries are stopped and stemmed using the INQUERY stoplist and Krovetz stemmer. Queries are further stopped to exclude 18 TREC stopwords such as '*describe*' (Allan et al., 1995). Candidate terms are all units of 1-3 words in the power set $\wp(Q)$ of content-bearing words in a query. IR models are defined in Section 7.4.

### 7.5.1 Feature analysis

This Section explores the impact of PhRank feature removal on IR effectiveness assessed using model PR-.F. Note that 'F' stands for 'False', so '.F' models exclude the feature represented by '.' and models '.T' include the feature. For simplicity, features that are used are typically not referenced, so 'PR-zF' is equivalent to 'PhRank rTsTzF'.

This Section also offers a limited interpretation of the word dependence principles used during ranking. These dependence assumptions are clarified by reference to four models of *phrase belief* presented by (Croft et al., 1991) (Figure 7.2, a-d). The models show how belief in a document $d_c$ in a collection $C$ flows to belief in a query $Q$ in an inference network, and thus how words and terms can be dependent. PhRank does not

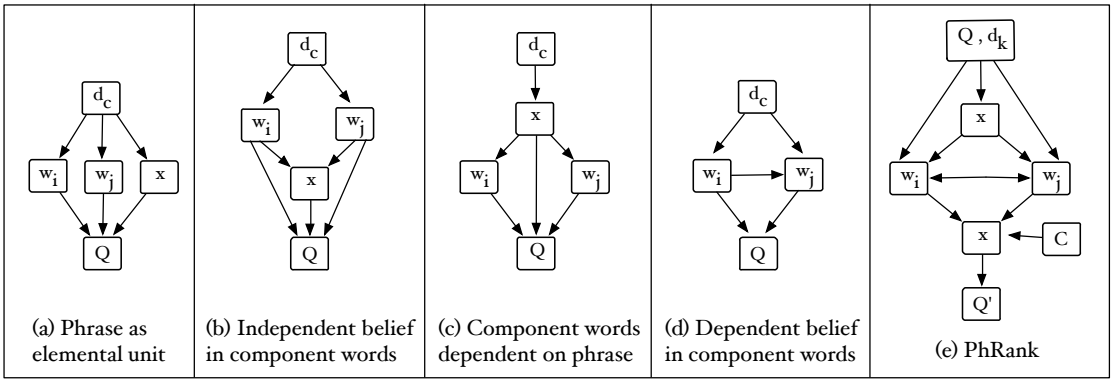



Figure 7.2: Four models of phrase belief proposed by (Croft et al., 1991) (a-d). Word dependence in PhRank can be understood as a hybrid with features of all these models (e) for term $x = \{w_i, w_j\}$ and documents $d_k \in N$.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Pr | MAP | R-Pr | MAP | R-Pr |
| *Description topics* | | | | | | |
| rTsTzT | 26.65 | 30.05 | 22.60 | 26.14 | 28.83 | 34.55 |
| zF | **27.32** | **30.32** | **23.68** | **26.71** | 28.64 | 34.13 |
| sF | 26.03 | 29.61 | 21.00 | 25.10 | 27.93 | 33.67 |
| rF | 26.67 | 30.02 | 22.44 | 25.70 | **28.93** | **34.65** |
| *Title queries* | | | | | | |
| rTsTzT | 24.87 | 29.04 | 21.78 | **25.73** | 31.49 | 37.26 |
| zF | 26.14 | 30.13 | 20.85 | 24.72 | 30.73 | 36.26 |
| sF | 25.90 | 30.03 | 20.72 | 24.30 | 31.30 | 36.91 |
| rF | **26.32** | **30.25** | **21.81** | 25.70 | **31.59** | **37.42** |

Table 7.2: Feature analysis results. Description topics perform best with omission of the global term weight $z$ (zF). Title queries perform best with the omission of bigram salience weight $r$ (rF).

perform inference, but by analogy these models aid interpretation of PhRank features.

Of the four models in Figure 7.2, the dependence assumption (d) is used by PhRank to score words, and term ranks are computed using an independence assumption (b). Even if component words of terms are not connected in $G$, weight is propagated through the graph such that word dependencies affect evidence for a term. PhRank factors $z$, $s$ and $r$ reflect Figure 7.2 models (a), (b) and (c) respectively. Results following the removal of each feature are shown in Table 7.2 and discussed in more detail below.

1. **Factor $r$ words dependent on term:** Factor $r$ imperfectly captures belief in component words dependent on belief in a term (Figure 7.2c). It uses global bigram statistics to scale edge weights in $G$. During a random walk, this affects

the affinity scores for individual stemmed words. However, bigram statistics are only an approximate measure of termhood. More problematically, $r$ relies on terms having their component words connected in $G$. This is likely for highly informative terms, but not guaranteed. Perhaps due to these limitations, $r$ had minimal impact on IR effectiveness for title queries and could be omitted to improve algorithm efficiency.

However, $r$ is useful for description topics. I speculate that this is because the query words for description topics may be peripheral to the core information need. Spurious adjacent words in $Q$ tend to appear in the pseudo relevant set because bigrams are a feature in the IR model employed for initial retrieval. Thus, if word co-occurrence in $Q$ reflects query meaning, as typically occurs with title queries, the edges and weights used to construct $G$ are likely to be adequate. If word co-occurrence is spurious, the construction may be suboptimal. Factor $r$ ameliorates misleading initial edge weights for description topics.

2. **Factor $s$ word independence:** Factor $s$ contributes to belief in a term dependent on belief in individual words (Figure 7.2b). It weights each vertex in an affinity graph by its salience in the query context $N$ balanced by its salience in the document collection. Omission of $s$ substantially hurt IR effectiveness. Among all the features tested it had the most impact on overall performance, perhaps because independent belief in words is the most important factor for IR effectiveness (Metzler and Croft, 2005). It may also be that $s$ is effective because salience in $N$ approximates semantic closeness to the query.

3. **Factor $z$ term as elemental unit:** Factor $z$ represents belief in a term independent of belief in its component words (Figure 7.2a). It resembles a standard *tf.idf* weight and reflects the principle that a term should be discriminative in the retrieval collection. Given the established effectiveness of *tf.idf* weighting, it is surprising that omission of $z$ improves IR effectiveness for description topics. However, $z$ is based on observations of a term in an unordered proximity window in the retrieval collection. The way such observations are made may not provide an accurate estimate of term salience. In addition, it has recently been suggested that global statistics rarely improve retrieval performance and that local, document level evidence is sufficient (Macdonald and Ounis, 2010).

   Both $r$ and $z$ also account for the discrimination ability of multi-word units in the collection: $r$ applies to bigrams and $z$ applies to words in unordered windows.

This encoding is partially redundant, so description queries may not require $z$ because they use $r$, and title queries may require $z$ because they do not use $r$. $z$ is removed for the final runs for description queries, and retained for title queries.

4. **Factor $k$ pseudo relevant documents:** Results in Table 7.3 show that the most improvement in IR effectiveness is achieved with 2 to 5 pseudo relevant documents. Higher $k$ decreases effectiveness due to the introduction of non-relevant information. However, PhRank is quite robust to variation in $k$ due to the weighting of co-occurrence relations by document relevance. Even with construction of the affinity graph from the original query only (¬PRF), PhRank performs better than sDist and comparably with SD. Variations on $k$ using passage retrieval and other document length filters (Allan, 1995) were not explored.

In summary, results show that not all the features proposed consistently improve term selection. Description topics are most effective when factor $z$ is omitted, and title queries are most effective when $r$ is omitted. The implementations of PhRank used in subsequent experiments reflect these observations. Description topics use models 'zF' and title queries use models 'rF'.

## 7.5.2 Retrieval performance

### 7.5.2.1 Robustness

For description topics, the results in Table 7.4 show highly significant or significant improvement in mean average precision (MAP) and R-precision compared to the SD baseline for GOV2 and WT10G. Substantial improvements in precision on Robust04 are just short of significance. For title queries, improvement is highly significant for WT10G and comparable to the baseline for other collections. Increased precision occurs for top ranked documents (top 5 and 10) as well as being a general trend in the results. Exclusion of Wikipedia has a small negative effect as shown by PR¬W and PR¬W2 corresponding to PR-.F and PR-zF2 respectively.

To assess the quality of ranked lists of terms, queries were reformulated using 1-10 top ranked terms (or as many terms as possible - some short queries generated fewer than 10 terms). The results in Figure 7.3 show that the effectiveness of reformulated queries is stable as more terms are included beyond the first 2-5 ranked terms. Further, a large part of the gain in precision is attributed to the top two terms. This suggests that most important information is retained by the term selection process.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Pr | MAP | R-Pr | MAP | R-Pr |
| ¬PRF | 26.44 | 29.59 | 21.88 | 25.36 | 27.85 | 33.28 |
| k2 | 26.86 | 30.05 | 22.76 | 25.42 | 28.81 | **34.38** |
| k5 | **27.32** | **30.32** | **23.68** | **26.71** | 28.64 | 34.13 |
| k10 | 27.29 | 30.05 | 22.33 | 25.02 | 28.64 | 34.16 |
| k50 | 27.09 | 30.11 | 23.04 | 26.21 | **28.82** | 34.27 |
| k100 | 26.80 | 29.82 | 22.78 | 26.11 | 28.34 | 33.91 |

Table 7.3: IR effectiveness for description topics using $k$ pseudo relevant documents. Best IR effectiveness is achieved using the top few documents only.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Pr | MAP | R-Pr | MAP | R-Pr |
| *Robust and precise* | | | | | | |
| QL | 25.25 | 28.69 | 19.55 | 22.77 | 25.77 | 31.26 |
| SD | 26.57 | 30.02 | 20.63 | 24.31 | 28.00 | 33.30 |
| sDist | – | – | 21.14 | 24.93 | 27.64 | 33.50 |
| PR-zF | **27.32** | **30.32** | **23.68**‡ | **26.71**‡ | **28.64**† | **34.13**‡ |
| PR¬W | 27.19† | 30.12 | 22.90† | 26.57 | 28.18 | 33.77 |
| *Succinct* | | | | | | |
| KC | 25.62 | 28.89 | 20.15 | 22.58 | 26.88 | 32.73 |
| PR-zF2 | 25.91 | 28.92 | 22.02† | 25.69‡ | 27.04 | 32.75 |
| PR¬W2 | 25.76 | 28.33 | 21.43 | 25.40† | 26.05 | 31.75 |

(a) TREC description topics

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Pr | MAP | R-Pr | MAP | R-Pr |
| QL | 24.37 | 28.52 | 19.48 | 23.08 | 28.55 | 34.41 |
| SD | 26.16 | 30.25 | 20.97 | 23.75 | 31.25 | 36.88 |
| PR-rF | 26.32 | 30.25 | **21.81**‡ | **25.70**‡ | **31.59** | **37.42** |
| PR¬W | **26.44** | **30.40** | 21.76† | 25.57‡ | 31.50 | 37.14 |

(b) TREC title queries

Table 7.4: Retrieval results for description topics and title queries. PhRank significantly outperforms a highly effective baseline for description topics and is strongly competitive for title queries. † shows significant ($p < .05$) and ‡ highly significant ($p < .01$) results compared to SD and KC respectively as determined by a sign test.

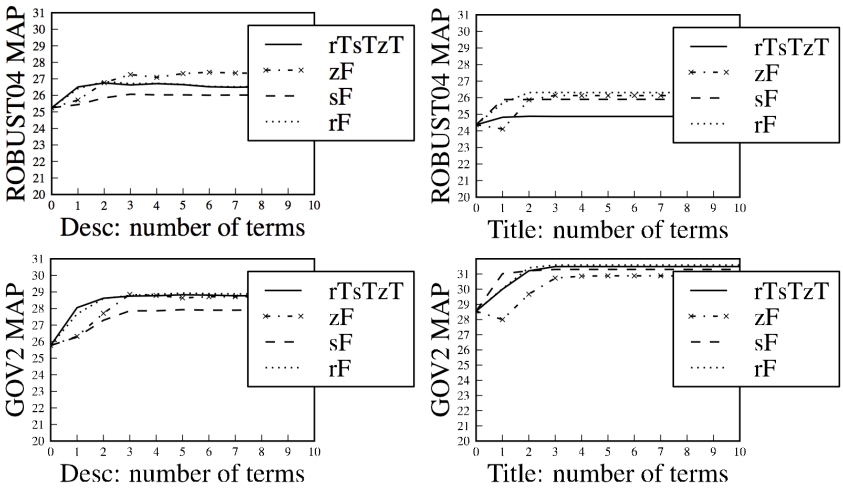



Figure 7.3: IR effectiveness with feature analysis and variable threshold. In many cases PhRank achieves performance gains with two terms, and is robust to variance in the number of terms selected.

In addition, for each collection I manually reviewed the ranked term lists for queries that perform significantly better or worse than SD (>100% change in MAP), and 10 queries with comparable performance. Across all queries observed, there is a strong tendency for PhRank to single out one word, or a pair of words, as the main concept of a query, and rank all terms that contain that concept above terms that do not according to the contributions of any additional words. For example, for query #663, '*What were the health effects of Vietnam veterans' exposure to Agent Orange?*', the five top ranked terms, in order, were {*exposure, veterans exposure, vietnam exposure, veterans exposure orange, veterans exposure agent*}. This high risk, high reward strategy can negatively affect the robustness of PhRank on a query by query basis as shown in Figure 7.4 for description topics. Around a quarter of all queries have more than a 5% decrease in MAP. Title queries exhibit similar behavior.

For example, one of the best performing queries for GOV2 is #756 as shown in Table 7.1 ('*Locations of volcanic activity which occurred with the present day boundaries of the U.S. and its territories*'). For this query, identification of '*volcano*' as the main concept greatly helped retrieval. The same strategy hurt query #780, one of the worst performing queries for GOV2, '*How much of planet Earth is arable at present? Area must have plenty of water, sun and soil to support plant life*'. Table 7.5 shows that PhRank selected '*earth*' as the main concept that subsequently appeared in all the top five terms. These terms were representative of the query, but not well distributed. In contrast, the SD baseline benefited from terms such as '*soil support*' and '*plant life*'.

Nevertheless, Figure 7.4 shows consistent improvement for queries that are known

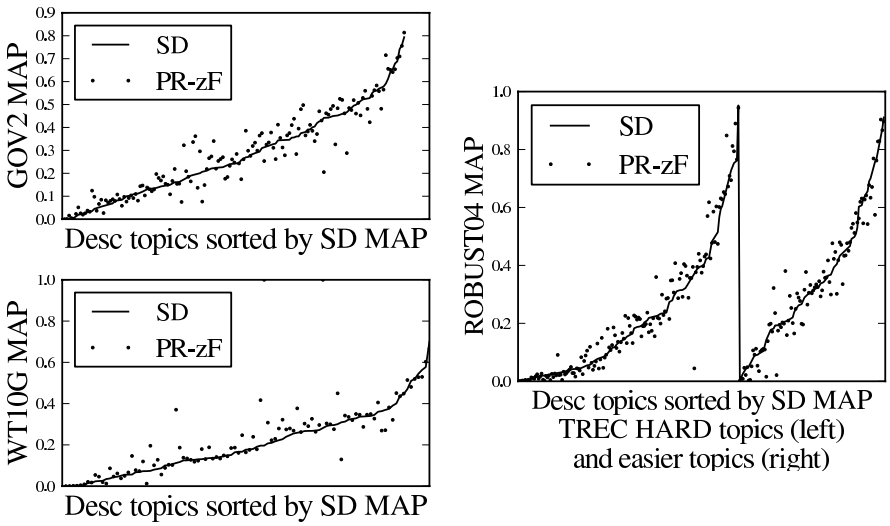



Figure 7.4: MAP difference with PhRank compared to SD per query for description topics. A focus on one concept usually helps, but can significantly hurt some queries.

| Q: *How much of planet Earth is arable at present? Area must have plenty of water, sun and soil to support plant life.* | | |
|---|---|---|
| **PhRank terms** | **SD terms** | |
| earth | planet earth | water sun |
| earth arable | earth arable | sun soil |
| planet earth | arable present | soil support |
| earth life | present area | support plant |
| earth water | area water | plant life |

Table 7.5: TREC query #780: poor performance for PhRank compared to SD.

to be harder (Robust04 HARD track) or easier (high baseline MAP). Notice in the plots that queries are sorted by baseline MAP, so harder queries are shown towards the left and easier queries are on the right. It is more likely that PhRank selects an appropriate main concept for easy queries because the pseudo relevant documents are of high quality. Difficult queries are less clearly defined and often benefit from the strong directional focus provided by PhRank terms.

In comparison, models like SD and sDist, take a robust approach to term selection with a distribution of possibly relevant terms. This presents a very different term selection strategy. Naturally, one potential avenue for improvement is interpolation of PhRank term selection with bigrams in SD. However, the robustness of a distributed term selection approach can come with a tradeoff in overall effectiveness. Initial interpolation experiments with a weighted linear combination of SD and PhRank terms did not yield any benefit over PhRank terms alone.

Alternatively, the properties of $G$ may be turned to advantage. It has been observed that a Markov field framework can select general and robust query expansion terms if edges in $G$ are identified using co-occurrence in large resources such as Wikipedia (Collins-Thompson and Callan, 2005). A combination of query expansion and term selection using a Markov field framework may balance complementary high reward and robust query reformulation strategies and result in significant overall gains.

#### 7.5.2.2 Precision

Results in Table 7.4 show significant improvement in MAP and R-precision for PhRank compared to sDist for both GOV2 and WT10G. PhRank terms are significantly more precise on average than the highest precision models even though terms are unweighted. Scenarios in which a high precision term selection strategy negatively affected query effectiveness were determined by a manual review of queries and results. First, PhRank sometimes picks a suboptimal concept. This is demonstrated with the high rank for '*earth*' in query #780 (Table 7.5). Selection of a sub-optimal concept occurred particularly in the presence of polysemous or highly co-occurrent words in a query, or irrelevant documents in $N$.

In the case of highly co-occurrent words, I speculate that their representative nodes in $G$ have a higher in-degree so they tend to accumulate weight during a random walk. A reduction in the number of iterations may help to address this problem. In addition, irrelevant documents in $N$ seem to affect the adequacy of an affinity graph $G$ constructed using $N$. $G$ is highly reliant on the quality of the initial query, the precision

| | Term Length | | | | | | |
|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| KC | 37% | 40% | 15% | 6% | 1% | <1% | <1% |
| PhRank | 22% | 54% | 24% | | | | |

Table 7.6: Percentage of PhRank and KC terms with various lengths.

| **PhRank** (1-3 words) | **SD** (2 words) | **sDist** (3-6 words) | **KC** (1-7 words) |
|---|---|---|---|
| ROBUST04 | 23% | - | 12% |
| WT10G | 28% | 11% | 18% |
| GOV2 | 27% | 15% | 16% |

Table 7.7: Percentage of PhRank terms selected by other models. Low figures show that PhRank detects novel terms with long-range dependencies.

of the document similarity metric, and the adequacy of the collection being searched. If non-relevant documents occur in $N$ there will be reduced connectivity in $G$, and this has an undesirable impact on the balance of word affinity scores. One solution to this problem may be to merge ranked lists computed by PhRank using different resources. Accurate predictions of term informativeness made by different instances of PhRank are likely to be more consistent than errors.

Second, more than one focus can occur, particularly in long queries. For example, there are two focal concepts of query #336: "*A relevant document would discuss the frequency of vicious black bear attacks worldwide and the possible causes for this savage behavior*". The two core concepts are '*black bear*' and '*savage behavior*' but PhRank largely misses the importance of black bears. Instead, its top ranked terms for this query are {*savage, savage behavior, bear savage, vicious savage, attacks savage*}. The strong focus on '*savage behavior*' has a negative impact on IR effectiveness.

#### 7.5.2.3 Succinctness

Results show that the performance of the top two PhRank terms in the same query structure as KC with no term weighting perform comparably to KC with term weighting. The length of the terms is similar in both models, with around 75% of terms having a length of 1-2 words. This suggests that improved performance of unweighted PR-zF2 queries is likely due to differences in the strategy for term selection. Note that KC shares the distributed approach to term selection with SD and sDist. KC selects two distinct concepts, whereas the top two terms selected by PhRank typically overlap.

More generally, it is observed that the succinct terms selected by PhRank are also novel. Table 7.7 shows that although PhRank and KC have the same number of 1-2

word terms overall, they display less than half of their potential overlap. Moreover, around 50% of PhRank terms contain two words, but only around half of them are also selected by SD. Terms that are three words long dominate sDist (69% of all terms) yet less than half of the terms with three words in PhRank are also found in sDist queries. One likely explanation for these findings is that PhRank is not limited by syntactic or adjacency relations that are used in the other models. It detects distant word dependencies because repeat co-occurrences of word combinations reflect the associations in which they take part.

#### 7.5.2.4 Pseudo relevance feedback

The performance of models that leverage pseudo relevant documents for query reformulation are shown in Table 7.8. PhRank using only the retrieval collection significantly outperforms the CH model. Conversely, in most cases the RM significantly outperforms PhRank. However, a combination of PhRank term selection and RM query expansion is at least as good as, and sometimes outperforms, the RM. Significant improvements are observed for PR¬W.e compared to RM for WT10G description topics ($p = 0.013$) and GOV2 title queries ($p = 0.03$). In addition, PhRank plus RM expansion words (PR¬W.e) delivers a significant improvement over PhRank (PR¬W.) for WT10G description topics and a highly significant improvement over PhRank for GOV2 title queries. In contrast, there is no significant improvement with the RM for these query sets. Finally, for both description topics and title queries, improvements in IR effectiveness for expanded versus unexpanded PhRank are significant for all collections. Psuedo relevance feedback can be thought of as a combination of local context analysis and query expansion. The effectiveness of PhRank is not due to query expansion, but may be partly attributed to local context analysis.

## 7.6 Conclusion

This Chapter presented PhRank, a term ranking algorithm that extends work on Markov chain frameworks to select focused and succinct terms from within a query. PhRank captures query context with an affinity graph constructed from word co-occurrence in pseudo relevant documents. A random walk of the graph leverages associative relations between words to identify words that are most salient. Word salience is integrated with frequency-based weights to identify the most informative query terms and achieve highly effective retrieval.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Pr | MAP | R-Pr | MAP | R-Pr |
| *Description topics* | | | | | | |
| SD | 26.57 | 30.02 | 20.63 | 24.31 | 28.00 | 33.30 |
| PR¬W | 27.19 | 30.12 | 22.90 | 26.57 | 28.18 | 33.77 |
| CH | 23.06 | 25.85 | 18.16 | 20.68 | 22.51 | 28.42 |
| RM | 30.32‡ | 32.24‡ | 23.46 | 26.49 | 30.64‡ | 35.12 |
| PR¬We | **30.33**‡ | **32.36**‡ | **24.30**† | **26.97**† | **30.76**‡ | **35.53**‡ |
| *Title queries* | | | | | | |
| SD | 26.16 | 30.25 | 20.97 | 23.75 | 31.25 | 36.88 |
| PR¬W2 | 26.44 | 30.40 | 21.76 | **25.57** | 31.50 | 37.14 |
| CH | 21.22 | 24.92 | 17.48 | 19.46 | 20.89 | 26.38 |
| RM | 29.91‡ | 32.36‡ | 22.16 | 24.26 | 33.43 | 37.28 |
| PR¬W2e | **30.03**‡ | **32.66**‡ | **22.41** | 24.03 | **34.28**‡ | **38.72**‡ |

Table 7.8: Applications of pseudo relevant documents for query reformulation. Query expansion significantly improves over queries without expansion. PhRank with expansion (PR-.Fe) makes the same number of passes over the data as the Relevance Model (RM) and can be significantly more effective. † shows significant ($p < .05$) and ‡ highly significant ($p < .01$) results compared to PR¬W. as determined by a sign test.

PhRank focuses on a limited number of words that represent a core query concept. Overall, this strategy is more effective for both description topics and title queries than a distributed approach to term selection. Empirical evaluation using newswire and web collections demonstrates that both recall and precision of reformulated queries is significantly improved for description topics and at least as good for short, keyword queries compared to highly competitive IR models. In addition, the queries generated are interpretable by users because they are unweighted and contain a few, short terms (typically 1-2 words). Competing models use query subsets up to 7 words long and can have up to 90% more terms (PhRank uses 1-5 terms in comparison with the subset distribution model that always selects 10 query subsets). One to five terms selected by PhRank in an unweighted model deliver up to 14% performance improvement compared to highly competitive models that use up to 30 terms. PhRank also avoids weights that are difficult for users to interpret, particularly when they have precision up to 4 decimal places.

Nevertheless, the PhRank term selection strategy is risky and less robust than competing methods. For all collections, around 26% of queries have more than 5% decrease in MAP compared to SD (significant change is around 3-6%). The two main issues are variation in the quality of pseudo relevance feedback, and the handling of queries with multiple concepts. The first issue may be ameliorated by adaptive methods

for the selection of $k$ that address challenges with the depth of coverage in a collection. The second issue may be addressed in several ways, including non-linear interpolation of PhRank term selection with a highly robust retrieval model applying distributed term selection. Query expansion is one such interpolation that achieves highly significant improvements in IR effectiveness over unexpanded PhRank-based queries without an additional pass over pseudo relevant documents.

In conclusion, the insights gained from analysis and experimentation in earlier Chapters contributed to an accurate prediction of the success of a novel ad hoc IR model using PhRank terms. PhRank queries significantly improve retrieval effectiveness for verbose queries compared to highly effective IR models. Opportunities for further gains are explored in the next Chapter.

# 8

# Word Association and Term Discrimination

The contribution of word associations to IR may seem paradoxical. On the one hand, word associations have a marginal relationship with the discriminative ability of terms. Semantically representative terms for a query may be too frequent to discriminate relevant documents in a collection. On the other hand, word associations can be successfully leveraged in feature-driven probabilistic frameworks. It seems they make term selection and weighting for IR easier because they help to exclude terms that might otherwise confuse data-driven processes. They can also provide valuable context for the interpretation of words in word dependence models, and may compensate for suboptimal query formulation. In principle, syntactic and statistical methods identify cases in which words could be trivially and meaningfully related by linear sequence in an alternative lexicalization of a query.

In the past, these opposing trends have made it difficult to determine how word associations can most profitably be applied in IR. Consider two contrasting examples in which phrase structures and statistics for word co-occurrence are applied to query term selection. In the 1960s, the well known SMART system used collection statistics to identify candidate word associations, and syntactic features to differentiate discriminative terms in this candidate set (Salton, 1964b). Salton claimed that phrase structure accounts for both word relations and "the most useful word groupings" but cannot describe syntactic variation. Statistical phrases were thought to overcome the problem of syntactic variation but did not identify the type of relation between concepts. They also tended to identify word combinations that were not discriminative. Salton re-

solved these difficulties by using syntactic phrases to eliminate statistical phrases that composed words without an identifiable linguistic relation.

In contrast, a recent competitive model for verbose queries took the opposite approach. The Key Concept model (Bendersky and Croft, 2008) identifies candidate word associations with grammatical categories (instead of statistical word co-occurrence), and uses co-occurrence statistics to differentiate discriminative terms (instead of grammatical categories). The fact that well-reasoned arguments exist for each approach indicates the challenge of predicting where language processing can most usefully be applied in IR.

The strong performance of the Key Concept model relative to IR in the 1960s highlights how features of word association may be more or less effective depending on the circumstances of their application. The Key Concept model is more effective than Salton's model because it uses a statistical approach to ameliorate the effect of term selection based on grammatical categories. Notably, parsing has also improved since the 1960s, but this is unlikely to be a significant factor. More accurate parsing does not overcome limitations of grammatical categories in the presence of syntactic ambiguity, textual economy, and lexical relations (see Sections 4.3.2 and 6.2.2).

I contend that these models illustrate a general principle. Word association only helps to determine document relevance when incorporated in statistical or probabilistic techniques. Of course, it is easy to argue for an alternative view that word associations, and particularly syntactic word associations, should always contribute to search effectiveness because they more accurately represent text semantics. Indeed, both the SMART system and the Key Concept model make this erroneous assumption when using grammatical categories as determinate constraints to select discriminative terms.[1] The impact of this is ameliorated in the Key Concept model by the addition of a further selection mechanism and term weights. Nevertheless, we should expect, and in fact can observe, better performance if the same or similar mechanism is applied to a less restricted set of candidate terms. This was confirmed in more recent work by the same authors (Bendersky et al., 2010).

The reason that a probabilistic framework is so important for linguistic features can be understood by analogy to the Association Hypothesis (van Rijsbergen, 1979b) (see also Chapter 7). The Association Hypothesis states that, "if one index term is good at discriminating relevant from non-relevant documents, then any closely associated

[1]The first novel term selection method proposed in this dissertation also makes this error. This facilitates some useful comparisons, as reported in Section 8.1.2.

index term is also likely to be good at this". The relationship between words (referred to as 'terms') is statistical and quantified (see measures, Section 4.5.7). To formalize this statement, let $I(x_i, r)$ be a measure of the information that a word $x_i$ contains about relevance $r$, and let $I(x_i, x_j)$ be a measure of the information that $x_i$ contains about another word $x_j$. The hypothesis claims that if $I(x_i, r)$ is large then $I(x_j, r)$ is also large provided that $I(x_i, x_j)$ is large (van Rijsbergen, 1983).

If we wish to make a similar claim about the information that a syntactically or semantically related word combination contains about document relevance, we merely need to substitute a linguistic word association for $x_j$. For example, let $I(x_i, NP)$ be a measure of the information that $x_i$ carries about the grammatical category of noun phrase ($NP$) (recall from Section 3.3.1 that individual examples of word combinations are used to identify the existence of grammatical categories). By analogy, if $I(x_i, r)$ is large then $I(NP, r)$ is also large provided that $I(x_i, NP)$ is large. In other words, if $x_i$ is a prototypical noun phrase and always appears as a noun phrase, then noun phrases carry a large amount of power for the selection of $x_i$ as a highly discriminative term.

This is a powerful statement. If a word combination in a query is a noun phrase, then its ability to discriminate relevant documents is not determined by the fact that it is a noun phrase *per se*. Rather, it is determined by the characteristics of the retrieval collection and the frequency with which that combination appears as a noun phrase versus the frequency with which it appears with some other presentation. Unfortunately, the flexibility of language means that many word combinations are not restricted to one type of linguistic unit. This means we cannot guarantee that $x_i$ will always appear as a noun phrase. For example, if $x_i$='*bank account*' we might talk about an account with a bank (noun phrase) and '*...the banks account for the crisis by pointing elsewhere...*' (verb phrase). When word combinations in a corpus regularly have relationships other than the one used for selection this tends to reduce the discriminatory power of the selected relationship. The matter can be further complicated by IR systems that interpret word combinations using proximity measures. Most IR systems do not parse entire collections and thus are limited in their ability to discern word relationships in documents.

The end result is that the strength of a relationship between a feature of word association, such as noun phrases, and document relevance depends on the particular lexicalized examples of that feature in a query. There is no generally valid statement about the relationship between features of word association and the discriminative ability of terms because the relationship can change with every query.

Under these conditions, it is of little surprise that word association methods appear to be of marginal utility for the identification of discriminative terms (see Chapter 5). A solution to this predicament uses features of word association in probabilistic frameworks that can represent two types of uncertainty: whether a word combination represents a particular type of linguistic unit (including uncertainty about parse accuracy and the validity of theoretical linguistic choices), and whether the combination will present as that linguistic unit (or an approximation to it) in documents.

When probabilistic systems are not employed, features of word association generally contribute to the selection of informative terms only if they are used appropriately as functors of word association. If they are used as determinate features that identify discriminative ability, assumptions about their presentation in large collections can be inaccurate and cause IR performance to degrade. In light of this, a simple and convenient guideline for the application of word associations in IR is to use features of word association for tasks of word association, and not selection, unless some statistical interpretation of their discriminatory power is available.

Experiments in this Chapter clarify this strategy with two straightforward extensions to highly competitive PhRank-based queries presented in Chapter 7:

1. **Term candidate filters:** There is an argument that pre-filtered term candidates tend to increase the diversity in top-ranked PhRank terms and may therefore make the resulting queries more robust. However, filtering uses word association methods for a purpose that is essentially selective and determinate: the removal of certain terms from the candidate pool. Experiments show that such a pre-filtering strategy hurts IR performance.

2. **Triangulation:** Results in Chapter 7 suggest that interpolation of high precision PhRank-based queries with a robust IR model may result in queries that are even more effective. One possible interpolation strategy involves the selection of terms associated with top-ranked PhRank terms. The technique matches word associations to an appropriate task in query reformulation. Experiments show that this triangulation strategy delivers small gains in IR performance.

This Chapter reports the methodology and experimental results for these two extensions. Evaluation is limited to description topics since many methods of word association rely on syntactic processing. I conclude with a general discussion of the application of word association to improve relevance in ad hoc IR.

## 8.1 Term candidate filters

Any discriminative word appears in many nterms because nterms include all possible combinations of 1-3 words. The presence of a highly discriminative word in an nterm tends to promote its PhRank ranking and contributes to low diversity of top-ranked terms. Low diversity subsequently limits the effectiveness of queries when top ranked terms do not focus on the most important query concept, or when the most informative representation of a query is a term distribution. Diverse queries are more robust.

To address this problem, it may be helpful to replace nterms with terms identified by a word association method introduced in Chapters 5 and 6. Word associations are used to filter terms that are ranked by PhRank. This significantly reduces the number of terms to be ranked from several hundred to around 20 or 30. It also tends to improve the diversity of top-ranked terms. Despite these advantages, the filter adds a constraint to term selection that is derived from determinate features of word association. This has a negative impact on IR effectiveness.

### 8.1.1 Methodology

The effectiveness of queries reformulated using five top-ranked terms pre-selected by a word association method are compared to queries reformulated using five top-ranked nterms. The word association methods applied separately are: unigrams (Uni), bigrams (Seq2), trigrams (Seq3), noun phrases (NPs), bounded phrases (BPhr), governor-dependent pairs (GDep), catenae (Cat), mutual information terms (MI), and log likelihood terms (LogL) (see Chapters 5 and 6).

All term rankings are generated by PhRank model PR-zF (see Chapter 7), and the selected terms are incorporated in the same robust, highly effective linear feature model used in previous experiments (see Chapters 5 - 7). The reformulation model uses the framework of a sequential dependence model (SD, see Section 2.2.4) but replaces query bigrams with selected terms in ordered and unordered windows (cliques 2 and 3). Five terms are used in each query reformulation, or fewer if there are less than five ranked nterms (i.e. this parameter is not tuned). The Robust04, WT10G and GOV2 collections are used for evaluation, and all experiments are implemented using version 4.12 of the Indri search engine with Dirichlet smoothing, $\mu = 2500$.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Prec | MAP | R-Prec | MAP | R-Prec |
| PR-nterm | **27.32**‡ | **30.32**† | **23.68**‡ | **26.71**† | **28.64** | **34.13** |
| PR-Uni | 25.29 | 28.78 | 19.79 | 22.88 | 25.92 | 31.44 |
| PR-Seq2 | 26.41 | 29.90 | 21.05 | 24.06 | 28.37 | 33.72 |
| PR-Seq3 | 25.52 | 28.95 | 20.15 | 23.73 | 27.12 | 32.96 |
| PR-NPs | 25.98 | 29.31 | 21.91 | 25.03 | 28.11 | 33.62 |
| PR-GDep | 26.04 | 29.35 | 20.81 | 23.68 | 27.95 | 33.58 |
| PR-Cat | 26.75 | 29.78 | 22.47 | 25.92 | 28.21 | 34.06 |
| PR-BPhr | 26.62 | 30.01 | 21.79 | 25.53 | 28.39 | 33.67 |
| PR-MI | 25.83 | 28.99 | 20.15 | 23.90 | 27.74 | 33.53 |
| PR-LogL | 25.70 | 29.18 | 20.23 | 24.21 | 27.58 | 33.28 |

Table 8.1: Results for description topics using methods of word association to preselect candidate terms for PhRank (PR). All methods decrease IR effectiveness compared to ranking all possible combinations of 1-3 words (nterm). † shows significant ($p < .05$) and ‡ highly significant ($p < .01$) results compared to PR-Cat (the best performing filter) as determined by a sign test.

### 8.1.2 Results

Results are reported in Table 8.1. For all collections, ranking only those terms identified by a single word association method reduces IR effectiveness compared to ranking all possible combinations of 1-3 words. Catenae are the most informative of all association methods explored for two of three collections (Robust04 and WT10G), suggesting that they are able to constrain word associations while throwing away minimal useful information. Nevertheless, queries that use ranked nterms still achieve highly significant improvements in MAP compared to queries that use ranked catenae for the same collections. This corroborates the conclusion from Chapter 5 that determinate word associations have a marginal relationship with the discriminative ability of terms. Methods of word association are not an appropriate determinate constraint in IR.

To investigate whether word associations make effective features in statistical or probabilistic techniques, I compare the effectiveness of queries reformulated by PhRank to the effectiveness of queries reformulated using the supervised classifier presented in Chapter 6. Recall that the classifier includes features of word co-occurrence, dependency structure and phrase structure, as well as metrics that leverage term and document frequencies, all applied in a decision tree framework. Some of these features are undefined for nterms,[2] so I compare results with catenae instead.

[2]The full classifier cannot be applied to nterms because some ellipsis features cannot be computed for random combinations of words. To compare nterms identified by the classifier to nterms identified by PhRank, a limited version of the classifier would be required.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Prec | MAP | R-Prec | MAP | R-Prec |
| PR-nterm | **27.32**‡ | **30.32**† | **23.68**‡ | **26.71**† | **28.64** | **34.13** |
| PR-Cat | 26.75 | 29.78 | 22.47 | 25.92 | 28.21 | 34.06 |
| SF-12 | 27.03 | 30.20 | 21.62 | 24.81 | 28.57 | 34.01 |

Table 8.2: Queries containing catenae selected by PhRank (PR-Cat) do not generally perform as well as queries containing catenae selected by a classifier tailored with features specific to dependency paths (SF-12) for 2 of 3 collections. † shows significant ($p < .05$) and ‡ highly significant ($p < .01$) results compared to PR-Cat as determined by a sign test. Nterms ranked by PhRank perform best.

Table 8.2 shows that the supervised classifier selects more informative catenae than PhRank for Robust04 and GOV2, although the differences are not significant. Conversely, catenae selected by the classifier for WT10G are not as effective as those selected by PhRank. This might be explained by the fact that gold standard catenae used to train the WT10G classifier were not as informative as they were for other collections (oracle WT10G queries also did not perform particularly well). Results for Robust04 and GOV2 show that statistical combination of many different features of word association may indeed outperform techniques that use less comprehensive sets of features, but the gains are not significant and may come at the cost of substantially increased computational complexity.

## 8.2 Triangulation

In Chapter 7, the sequential dependence model (SD) and PhRank models were shown to pursue different highly effective IR strategies. SD is more robust while PhRank is more precise. Assuming the two strategies are complementary, interpolation of these models should perform better than either model alone. For example, consider the query '*The frequency of vicious black bear attacks worldwide and the possible causes for this savage behaviour*' (adapted from Robust04 #336). Only one of the top-5 nterms ranked by PhRank mentions bears, namely '*bear savage*'. However, this term can be triangulated with the sequential dependence term '*black bear*' to achieve a much more effective query. The results in Chapter 5 (see Table 5.4) also suggest that term combinations result in more robust improvements in IR effectiveness.

In this Section, the terms selected by the word association methods introduced in Chapters 5 and 6 are interpolated with PhRank top-ranked nterms. This is achieved by triangulation: PhRank terms are paired with a second term with which they share

one word. The query reformulation model is derived from the sequential dependence (SD) model (Section 6.4.1). A combination of an nterm and its triangulation term replace each individual term in the ordered and unordered windows (cliques 2 and 3) for all except the $m$ most highly ranked terms by PhRank. The top $m$ terms are not triangulated because it is assumed that triangulation will decrease query effectiveness for highly discriminative terms. For example, given a query '*new york city*' with two terms, '*city*' (rank 1) and '*new york*' (rank 2), and with $m$=1 and '*york city*' as the triangulation term for '*new york*', this results in the following Indri query. Notice the combination of '*new york*' and '*york city*' in cliques 2 and 3:

#weight(
$\lambda_1$ #combine(new york city)
$\lambda_2$ #combine( city #combine( #1(new york) #1(york city)) )
$\lambda_3$ #combine( city #combine( #uw8(new york) #uw8(york city)) )

Triangulation terms derive from a single word association method, and are selected using a PhRank ranking of terms identified by that method. While any query term should discriminate between documents, the primary purpose of triangulation terms is to associate one term with another (e.g. '*york*' from '*new york*' with '*city*' in the example). Using word association methods in this way tends to produce insignificant gains in IR effectiveness, rather than a significant decrease in IR effectiveness.

### 8.2.1 Selection of triangulation terms

The SD model diversifies term selection by applying a sliding window of two words along the length of a query. Similarly, a deterministic algorithm for the selection of triangulation terms ensures identification of a diverse and informative term set. To begin, the top $m$ nterms in a PhRank ranking (following diversity filtering - see Section 7.3.2) are set aside. Candidate triangulation terms are then identified using a word association method. These terms are filtered to exclude terms with one word since individual words cannot be used for triangulation. For the $5-m$ nterms that are triangulated, Figure 8.1 illustrates the following procedure. Given the $m+1$ nterm, the algorithm extracts component words of the nterm that have not been seen before (initially none). For each word that has not been observed, the algorithm identifies terms in the set of triangulation candidates that contain the word. The identified candidate that is most highly ranked by PhRank is the triangulation term. All component words in the nterm

**Q#336:** *A relevant document would discuss the frequency of vicious black bear attacks worldwide and the possible causes for this savage behavior.*

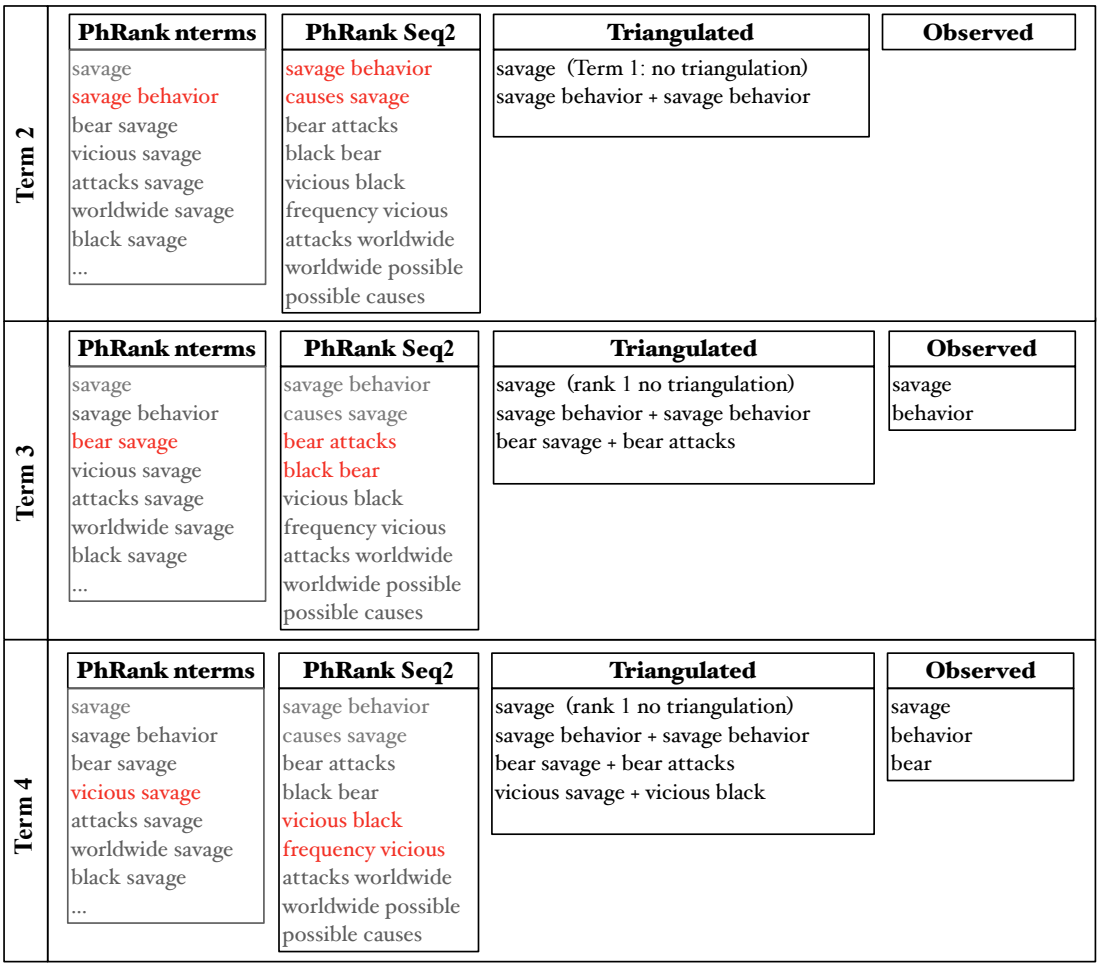



Figure 8.1: The process followed by the triangulation algorithm. The top-ranked nterm is not triangulated (term 1). Here, bigrams are used as triangulation candidates.

are added to the set of previously observed words, and the algorithm iterates to the next mostly highly ranked nterm until all of the top 5 nterms are exhausted.

Notice that it is possible for a triangulation term to be identical to its paired nterm (see Term 2, Figure 8.1). In this case, the triangulation term is dropped since it does not add provide additional information.

### 8.2.2 Methodology

Experiments compare the effectiveness of queries reformulated with five top-ranked nterms identified by PhRank to the effectiveness of queries reformulated with the same nterms triangulated by associated words. The word association methods used to generate triangulation term candidates are: bigrams (Seq2), trigrams (Seq3), noun phrases (NPs), bounded phrases (BPhr), governor-dependent pairs (GDep), catenae (Cat), mu-

tual information terms (MI), log likelihood terms (LogL) and nterms (see Chapters 4 and 6). All rankings are generated by PhRank model PR-zF (Chapter 7). Baselines replicate the baselines described in Chapter 7: query likelihood (QL), sequential dependence model (SD), Key Concept model (KC) and the subset distribution model (sDist).

Triangulation is applied to all nterms in the query reformulation except the $m$ most highly ranked by PhRank. $m$ is set to 1 following a grid search on the interval [0, 5] for Robust04 triangulated queries. Robust04, WT10G and GOV2 description topics are used for evaluation, and all experiments are implemented using version 4.12 of the Indri search engine with Dirichlet smoothing, $\mu = 2500$.

### 8.2.3 Results

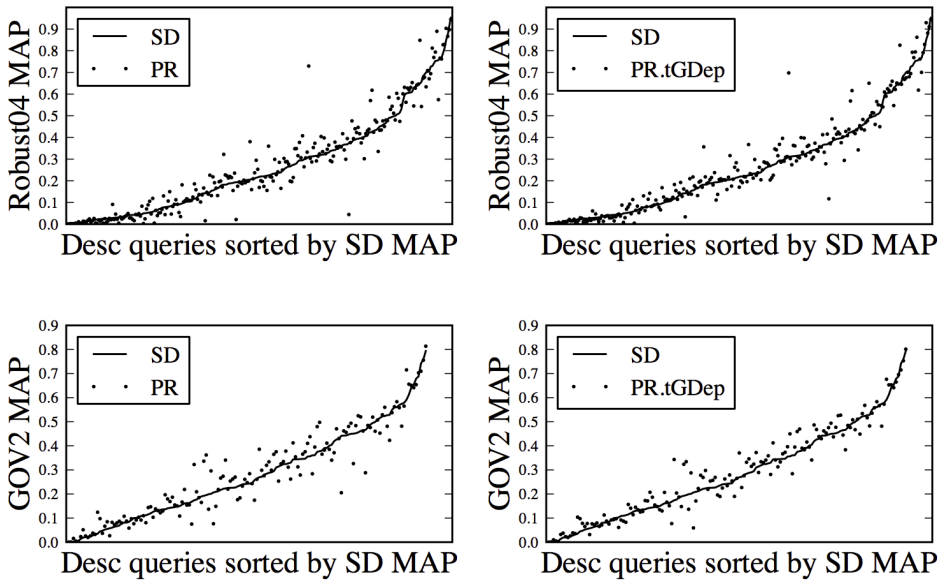
Results are shown in Table 8.3. Only triangulation with governor-dependent pairs and sequential bigrams consistently produces more effective queries than PhRank terms without triangulation. Improvements are robust but not significant. For WT10G, triangulation sometimes produces a very small decrease in effectiveness. For GOV2, triangulation with terms selected by most methods increases query effectiveness.

Overall, the benefit of triangulation is most noticeable in query by query analysis. Triangulation improves the effectiveness of the worst performing description topics, and in many cases does not diminish the effectiveness of topics that perform well (Figure 8.2). For example, for GOV2, 44 of 97 topics whose performance improved with PhRank compared to SD baseline (out of 150 topics in total) were negatively affected by triangulation. Of these, 14 experienced more than 5% negative change in MAP compared to SD. All of these 14 were relatively poorly performing topics so a large percent change translates to a small effect. In addition, out of the 97, a further 19 queries that were previously improved were even more effective.

There is very little difference between the performance of queries that use triangulation terms identified by different word association methods. Nevertheless, the performance of governor-dependent pairs (GDep) suggests that associations made by dependency relations are relatively reliable, just as they were for term filtering in Section 8.1.2. Catenae do not perform quite as well as GDep, presumably because longer terms are too restrictive in this context. However, the performance of bigrams (Seq2) is virtually indistinguishable from that of GDep, and entails less text processing. Bigrams are preferred for their simplicity.

| | ROBUST04 | | WT10G | | GOV2 | |
|---|---|---|---|---|---|---|
| | MAP | R-Prec | MAP | R-Prec | MAP | R-Prec |
| QL | 25.25 | 28.69 | 19.55 | 22.77 | 25.77 | 31.26 |
| SD | 26.57 | 30.02 | 20.63 | 24.31 | 28.00 | 33.30 |
| KC | 25.62 | 28.89 | 20.15 | 22.58 | 26.88 | 32.73 |
| sDist | 25.62 | 28.89 | 21.14 | 24.93 | 27.64 | 33.50 |
| PR-nterm | 27.32 | 30.32 | 23.68 | 26.71 | 28.64 | 34.13 |
| PR-tnterm | 27.35 | 30.45 | **23.77** | 26.82 | 28.77 | 34.22 |
| PR-tSeq2 | **27.56** | 30.58 | 23.69 | 26.65 | 28.97 | 34.28 |
| PR-tSeq3 | 27.11 | 30.04 | 23.34 | 26.57 | 28.81 | 34.17 |
| PR-tNPs | 27.32 | 30.36 | 23.59 | 26.60 | 28.98 | 34.37 |
| PR-tGDep | 27.52 | **30.59** | 23.73 | 26.72 | **29.08** | 34.37 |
| PR-tCat | 27.39 | 30.35 | 23.41 | 26.11 | 28.99 | 34.35 |
| PR-tBPhr | 27.38 | 30.47 | 23.42 | 26.52 | 28.97 | **34.49** |
| PR-tMI | 27.32 | 30.43 | 23.71 | 26.49 | 28.86 | 34.20 |
| PR-tLogL | 27.27 | 30.20 | 23.57 | **27.07** | 28.78 | 34.09 |

Table 8.3: Results of PhRank (PR) triangulation ('t.') for description topics. Triangulation with GDep and Seq2 marginally improves performance compared to PhRank without triangulation (PR-nterm) but the improvements are not significant.




(a) PhRank (untriangulated) (b) PhRank triangulated with GDep

Figure 8.2: Per query MAP difference compared to SD baseline for Robust04 and GOV2 description topics reformulated using (a) top-ranked PhRank nterms, and (b) top-ranked PhRank nterms triangulated with GDep terms. The triangulated queries are more robust with fewer queries resulting in a decrease in IR effectiveness compared to baseline.

Statistical word association measures do not perform particularly well. This suggests that features based on collection frequencies are better suited to determine the discriminative ability of terms than word association. I speculate that this is because global frequencies do not distinguish between different word senses and the particular semantics of a query versus general word use.

In addition, nterms as triangulation terms are not as effective as GDep pairs. The task of triangulation is predominantly one of word association, and nterms make no distinction between words that are associated and those that are not, even though PhRank ranking allows the possibility that the top-ranked nterms are associated (nterms include all terms identified by any word association method). Triangulation with terms selected by word association methods is marginally better than triangulation with nterms for two of three collections. This tends to confirm the hypothesis that a positive impact on search performance results when an IR technique matches the purpose of word association methods to their role in query reformulation.

## 8.3 Conclusion

It is difficult to come to a general conclusion about the utility of word association methods in IR, as shown by years of experimentation. Exploration of the relationships between word association and semantics or discrimination in this work also led to conclusions that were tempered by contrasting information.

For example, lexicalism seemed to be the best basis for term selection in principle, but noun phrases were remarkably effective in practice. Probabilistic techniques were thought to overcome specific conceptual problems with linguistic theory, but could not account for the success of heuristic techniques. Co-occurrence-based methods seemed more likely to identify all word relations with semantic significance than syntax or syntagms, but did not always improve IR effectiveness as much as those methods. In addition, word association was shown to have a marginal relationship with the discriminative ability of terms, yet probabilistic techniques that use many language features are known to be highly effective.

Given this evidence, it has not always been easy to for IR researchers to foresee the circumstances under which word associations might provide information about document relevance. This Chapter presented a simple, coherent guideline in this regard. When features of word association are combined with word and term frequencies in a probabilistic or statistical framework they can deliver small gains in discriminatory

power. However, as experiments in this Chapter show, they tend to hurt IR effectiveness if they are used determinately (for example, if a term is considered to be only a noun phrase). Application of word association methods to tasks that are predominantly focused on word association may not hurt effectiveness if they are used determinately, but are unlikely to help significantly.

Whether or not syntactic features are worth processing in IR is an important issue because feature extraction is usually highly coupled with indexing and storage systems of search engines. To date, the small improvements in ad hoc IR made possible by syntactic features have not been sufficient to drive change in commercial search engine development. The reason language processing might not deliver more than incremental improvements in IR effectiveness is considered in the next Chapter, along with directions for future work.

# 9

# Conclusions and Future Work

Current and future challenges in IR demand an understanding of the role and value of NLP in IR. User interaction and dialogue in search systems, verbose query representation, integration of structured data and unstructured text, and labelling of heterogenous information, all require a strong understanding of how humans and machines interpret the meaning of text, and the reliability and congruence of these interpretations.

It is sometimes assumed (Lease, 2007) that competing linguistic theories have the same value for the interpretation of text in IR. Yet theoretical evidence and empirical experiments in this dissertation show that three schools of linguistic thought emphasize different structures. Moreover, I contend that these structures play distinctive roles in the organisation of language, and that this affects their joint utility for search.

At a basic level, word co-occurrence, as the focus of lexicalism, forms core elements of meaning. It seems that word sequences are the most minimal arrangement for communicating semantic associations, so word order is a preferred mechanism for the organisation of language and selection of text terms. Semi-stable lexical elements consisting of pre-constructed phrases are embedded in language (Sinclair, 1991) where they behave like flexible units in structures that describe their inter-relationships. They are central to the construction of meaning.

The *intra*-element structure of language seems to be properly described by dependency grammars (Nivre, 2005). Unlike frequent word sequences, dependency relations do not indicate boundaries of lexical elements. Instead, they form an internal mesh between words that compose each element. These relations play a fundamental role in the normalization of meaning, especially in circumstances that cause language structures to be altered from a prototypical expression. The utility of dependency paths in IR, and particularly QA, suggests the value of this normalization function. Conversely, dis-

agreement about how certain words participate in dependency relations, and whether they need to participate in dependency relations at all (Karlsson, 1990), suggests that dependency structures offer limited information about how elements link together (see Section 3.2.6).

The *inter*-element structure of language seems to be more completely described by phrase structure grammars. This is the structure that operates between elements (rather than within elements) and also works to build up meaning. The easy substitution of units sharing any grammatical category is indicative of this inter-element structuring (Chomsky, 1957). In addition, the notion of transformation focuses on the organisation of lexical elements into clauses and sentences (rather than how words in lexical elements are arranged with respect to each other). Inter-element structuring enables phrase structure to discriminate meaningful elements such as noun phrases that work together to identify text meaning (see Chapter 5).

When communication is subject to constraints caused by complexity, brevity and other factors, as it may be in verbose query formation, the stress on a language system may be transmitted to all structures simultaneously. This impacts the cohesiveness of lexical elements and the expression of intra- and inter-element structures. The deformation[1] of one structure tends to reveal itself in the deformation of other structures. For example, consider the prototypical statement in example 9.1 and a similar statement that responds to a constraint on sentential focus (wh-fronting) in example 9.2:[2]

(9.1) Prototypical: *used toxic chemicals as a weapon*

(9.2) Wh-fronting: *how are toxic chemicals used as a weapon?*

Figure 9.1 shows that phrase structures and dependency structures are affected by communicative constraints. For the prototypical statement, if stopwords are excluded, a relation between '*toxic chemical*' and '*weapon*' is identifiable from the surface phrase structure parse in Figure 9.1 (a). Likewise, the dependency parse in Figure 9.1 (c) reveals '*toxic chemical weapon*' as a collapsed catenae. Moreover, these word associations can be trivially detected as an ngram in stopped text.

Now consider the statement in example 9.2. The movement of the lexical element '*toxic chemicals*' changes all three linguistic structures (phrase structure, dependency structure, and syntagmatic structure). In Figure 9.1 (b) this can be accounted for

[1] *Deformation* is used to describe alterations in language structure because it is theory-neutral, unlike *transformation* which describes the re-arrangement of grammatical categories.

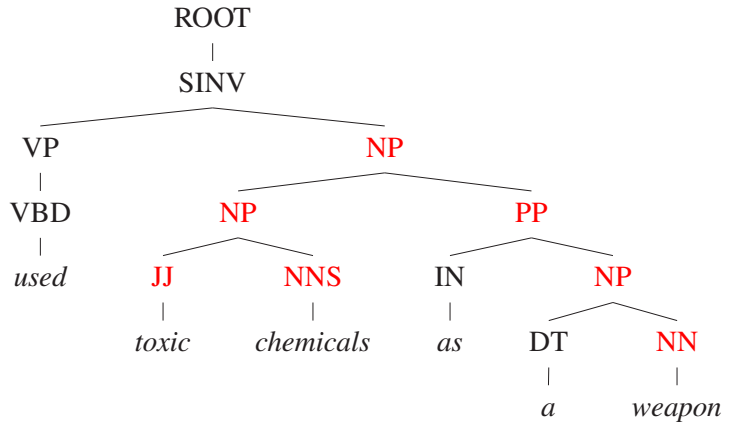
[2] These examples were first presented in Chapters 4 and 6.

by transformational rules, which are pivotal to explain the inter-element relationships in phrase structure but not accessible from the surface representation. Dependency grammars focus on intra-element structure, so the relationship between '*chemical*' and '*weapon*' is lost when '*toxic chemical*' rises to dominate its governor in Figure 9.1 (d) (the dashed dependency edge marks where a head is not also the governor and the g-script marks the governor of the risen catena, see Section 6.2.2). Finally, the sequence of words is also changed, interrupted by the non-stopword '*used*'.

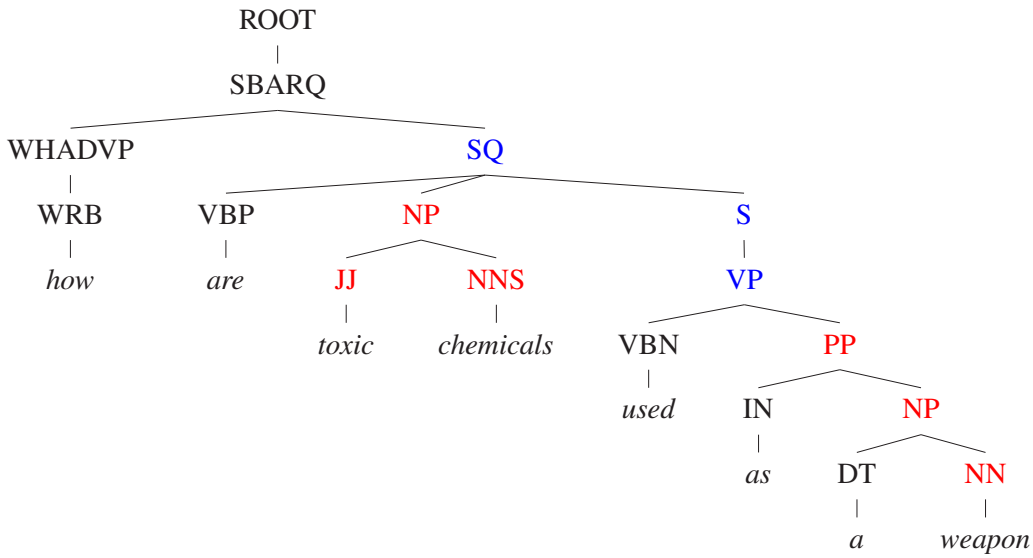
I contend that the systemic effects demonstrated in this example are common. We might expect to rectify the 'deficiencies' of simple methods by deep linguistic processing when meaning is not helpfully encoded in word order. However, when bigrams are not sufficient, other methods for the detection of word association are less likely to help. The failure of word sequences is diagnostic of cases where more nuanced features may not function as expected. By consequence, word sequences are often sufficient to approximate deep linguistic processing in ad hoc IR. This results in the (by now) unsurprising effectiveness of relatively simple IR models such as the sequential dependence model (Metzler and Croft, 2005) and ngram language models for IR (Song and Croft, 1999). Indeed, performance gains are small for IR models that use deep linguistic processing compared to models that rely solely on unigrams and bigrams.

To handle systemic deformation in language structures, an approach that permits inexact matching between queries and documents is required. This can be implemented simply using proximity measures, or more comprehensively modelled using a translation-based model for IR. Both these techniques are shown to be highly effective (Metzler and Croft, 2005; Park et al., 2011). However, translation techniques are computationally intensive and require parsing a document collection, which is likely to be impractical for many real world search applications.

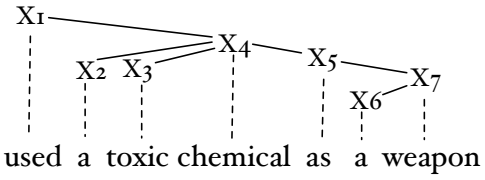
From a mathematical standpoint, systemic effects in language can violate the assumptions of probabilistic techniques. Linguistic features are correlated if they fail to unravel language meaning in the same circumstances, even if the word combinations they identify are different (as shown to be largely true in Chapter 5). Evidence for this correlation comes in the form of marginal, or non-existent, improvements in task performance when more language features are included in a model. We might conclude that new machine learning methods will do little to manage recalcitrant language phenomena and thus improve IR performance. Rather, language features themselves are the problem and any probabilistic technique that relies on them will likely fall short of perfect language interpretation.



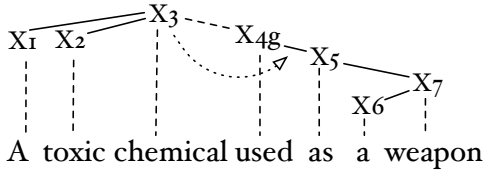

(a) Prototypical phrase structure




(b) Deformed phrase structure




(c) Prototypical dependency structure




(d) Deformed dependency structure

Figure 9.1: Deformation in linguistic structures in response to constraints on sentential focus. The relations between '*toxic*', '*chemical*' and '*weapon*' are accessible in the prototypical structures, but not the deformed structures. Notice that syntagmatic structure is also deformed in the non-prototypical case. The sequence '*toxic chemical weapon* is disrupted by '*used*' (stopwords are ignored in all cases).

Of course, linguistic features can deliver small improvements in IR performance. This is because some linguistic features are more likely to be correlated than others. For example, since phrase structure grammars do not abstract away from word order, features of phrase structure (such as part-of-speech tags and noun phrases) are more likely to be correlated with word sequence (such as ngrams) than features of dependency grammars (such as headedness) that can abstract from word order.

Nevertheless, experiments in this dissertation show that term selection techniques, such as PhRank, which use simple features may deliver search performance equivalent to that of probabilistic frameworks that discover, and make use of, many forms of linguistic evidence (see Chapter 7).

Perhaps this should be expected given that linguistics and IR tackle different problems. Mainstream linguistics describes the principles governing language production and understanding, and in practice these principles can be modified (resulting in ungrammatical, but nevertheless effective, communication). In contrast, IR focuses on the principles that govern document relevance and must cope with numerous exceptions to linguistic patterns when petabytes of data are consumed. This incongruity brings us back to the question asked at the beginning of this dissertation: can language structure help to identify word associations that improve search performance more than associations identified using simple word adjacency? Three conclusions about ad hoc IR can be drawn from the evidence presented in this work:

- **For ad hoc IR, syntactic analysis is not necessary.** Syntax does not help to identify terms that are significantly better than terms identified using simple word proximity. Given a large text collection, for every syntactic word association there is a word association that carries at least as much information about topical relevance and can be identified from unannotated text.
- **A statistical framework is necessary to interpret word associations.** Word associations that are either syntactic or based on word proximity reduce the potential for IR effectiveness when applied as determinate constraints on term selection.
- **More work is needed on semantics in IR.** Word associations identified by various methods are not strongly related to request semantics (as operationalized by user nominated terms), and request semantics is not strongly related to document discrimination. The way in which terms are combined appears to be more important than close semantic alignment with a query. This may be due to the

influences of textual economy, world knowledge, and language interpretation by users. However, much more work on semantics in IR is warranted to validate these findings.

It must also be pointed out that certain conclusions cannot be drawn from this work. Experiments do not demonstrate the failure of linguistics in IR. They also do not show that word proximity or bigrams are always as informative as syntax. Simply, the evidence suggests that while syntax captures essential word relationships, it does not capture *all and only* essential word relationships. In particular, syntax may still be useful for IR tasks focused at the sentence-level (such as question answering), or where fine-grained analysis is critical to success.

In this dissertation, I explored theoretical points with respect to language structure in IR and presented two novel, highly effective methods for query biased term selection. The first focused on a phenomenon of semantic word association with a discriminative filter that leveraged diverse features. The second did not use syntactic information, but instead incorporated word co-occurrence with frequency counts in a network model of query context (noting that a co-occurrence network mirrors the form of a global dependency graph).

The success of the second strategy illustrates that syntactic processing might improve the selection of word associations but does not necessarily translate into improved IR performance. Rather, just as vague terms seem to provide robustness to communication between people in the real world (Klein and Rovatsos, 2011), query terms that are "sufficiently compatible" (ibid.) seem to work well enough for IR. Highly effective methods of term selection can be devised using simple features that are substantially more efficient than deep language processing. Moreover, at least in English, strong performance still relies heavily on one of the simplest features of all: words.

## 9.1 Future Work

The insights developed in the course of this dissertation present important advances for the understanding and manipulation of language in IR. They also suggest three directions for ongoing work: (1) further improvement of the PhRank approach for IR; (2) applications of precise term selection; and (3) further exploration of language systems and applied language understanding.

### 9.1.1 Advances with PhRank

1. **Query expansion:** One of the main issues affecting robustness of PhRank term selection is the handling of queries with multiple concepts. In addition, Chapter 7 showed that interpolation of PhRank-based queries with Relevance Model expansion improves IR effectiveness. However, the efficiency of the Relevance Model is quadratic in the size of the retrieval collection (Chen and Fu, 2005). Query expansion might therefore be more simply and efficiently achieved by combination of PhRank term selection with expansion terms generated within a Markov chain framework. This framework for query expansion has already been proposed (Lafferty and Zhai, 2001; Collins-Thompson and Callan, 2005; Mei et al., 2008; Huang et al., 2010). However, the novel addition of a vertex weight in PhRank, based on word salience in pseudo-relevant documents, may further benefit performance.

2. **Probabilistic interpretation:** The current implementation of PhRank is chiefly heuristic, making it difficult to tune and adapt to new retrieval applications (Croft et al., 2010). An opportunity exists for development of a probabilistic interpretation of PhRank that capitalizes on analysis of query context. In particular, this might leverage the expected difference between the pseudo-relevant document set and general language. The probability of observing a word a second time depends more on lexical content than word frequency, and this adaptation is greater for content words than function words (Church, 2000). The probability of such positive adaptation might be used in place of node weights.

3. **Term weighting:** The results previously reported for PhRank were unweighted in order to highlight the impact of term selection alone. Yet it is well-established that discriminative term weighting can significantly improve IR model effectiveness. Some improvement is expected for a weighted implementation of PhRank even though term selection and term weighting are inversely related such that effective term selection tends to eliminate the need for term weighting. This is because weighting can ameliorate cases in which PhRank focuses on a peripheral concept by de-emphasizing uninformative terms.

   The effectiveness of a weighted implementation for PhRank using ranked nterms was compared to a top-performing dependency model, the Weighted Sequential Dependence Model (WSDM) (Bendersky et al., 2010).[3] Both WSDM

[3]WSDM was chosen on the basis of a systematic comparison of dependency models that shows

and weighted PhRank use a form of parameterized term weighting (Bendersky et al., 2011; Bendersky and Croft, 2012) that employs a coordinate ascent algorithm to converge on an optimal set of weights (Metzler, 2007). The algorithm iteratively optimizes over a number of feature parameters $w_i$ with respect to query MAP scores by performing a series of one-dimensional line searches. The procedure cycles through parameters $w_i$, while fixing all parameters $w_{j\neq i}$, and stops when the gain in MAP score is below a specified threshold. A term weight is calculated as a weighted sum of numeric feature values, each of which is multiplied by its corresponding weight parameter.

The original implementation of WSDM is a weighted extension of the sequential dependence model that optimizes over 18 features. Subsequent work showed that the number of features can be reduced without diminishing IR performance (Bendersky and Croft, 2012). These features include term frequency in a retrieval collection, Google n-grams, Wikipedia titles, and a commercial query log, plus document frequency in a retrieval collection and an a priori constant weight. An implementation by Huston (2013) is used to generate both WSDM and weighted PhRank queries. For each collection, parameters are trained separately for clique 1, and cliques 2 and 3 combined, using 5-fold cross validation.

Weighted models were investigated using the description topics of Robust04 and GOV2. WT10G was not considered since runs are compared to optimized parameters learnt by Huston (2013) that were not available for this collection. All description topics were stopped and stemmed using the Krovetz stemmer as for previously reported experiments. Weighted runs use the Galago Search Engine[4]. Baseline runs, copied for convenience from Chapter 7, use the Indri Retrieval Engine.[5] Both IR systems use Dirichlet smoothing, $\mu = 2500$. All nterm rankings are generated by PhRank model PR-zF (Chapter 7). Baseline runs for SD and PR-zF use default clique weights as previously reported: 0.85, 0.1, 0.05 (Metzler and Croft, 2005), and clique weights are optimized for weighted runs. Default weights are known to be relatively stable across collections and optimized weights have minimal impact on performance (Huston, 2013).

Results in Table 9.1 show that weighted PhRank performs comparably with

WSDM significantly outperforms all bi-word dependency models in several settings (Huston, 2013). A variant of BM25 was the only model to outperform WSDM and only in a few instances.

[4] `http://www.lemurproject.org/galago.php`

[5] In direct comparison tests, Indri and Galago exhibit minimal (0.01 - 0.02) differences in reported MAP scores. These differences are very small and can often be ignored.

| | ROBUST04 | | GOV2 | |
|---|---|---|---|---|
| | MAP | R-Prec | MAP | R-Prec |
| SD | 26.57 | 30.02 | 28.00 | 33.30 |
| PR-zF | 27.32 | 30.32 | 28.64 | 34.13 |
| WSDM | **28.03** | 31.03 | 29.65 | 34.58 |
| wt-PR-zF | 27.90 | **31.13** | **29.86** | **35.03** |

Table 9.1: Comparison of weighted and unweighted by PhRank and SD queries. There is no significant difference between weighted models, but PhRank uses fewer terms.

WSDM and slightly improves performance compared to an unweighted PhRank model. The major point of interest is that weighted PhRank is highly effective using shorter queries than WSDM. This confirms that better term selection reduces the need for term weighting in IR.

### 9.1.2 Applied term selection

Term selection developed for verbose queries may also be useful in other search tasks. For example, PhRank term ranking was applied to tagged image retrieval in preliminary experiments with 15 queries.[6] A list of 2325 possible tags were ranked by PhRank using five pseudo relevant documents for different formulations of text queries (title, description and narrative). Documents were retrieved from various resources including Google and English Wikipedia. An alternative approach ranked concepts using the Kullback-Leibler (KL) divergence between a language model constructed from the top 100 documents retrieved by Google with title queries, and a language model of the corresponding description topics. It was found that the rankings produced by these approaches were equally effective, but in many cases significantly different. A strategy that combined the two approaches was better at selecting representative image tags than either approach alone.

One of the challenges for PhRank during this work was the handling of out-of-vocabulary (OOV) words. For non-ambiguous queries with sufficient relevant documents, an affinity graph produced reliable tag rankings. However, errors were made where some tag words were not included in the affinity graph. Future work might improve the performance of tag ranking through graph expansion.

In other areas, improvements in term selection might feed into the identification of more effective combinations of terms. Analysis of term combinations in Chapter

[6] Part of the Aladdin project at the Centre for Intelligent Information Retrieval (CIIR) at the University of Massachusetts, Amherst. Collaborative work with Jeff Dalton.

5 suggests the potential strength of this line of development. There is presently little work on modeling higher-order term dependencies for IR (Bendersky and Croft, 2012).

### 9.1.3 Applied language understanding

I have asserted that the principles governing language production and understanding are malleable and respond to communicative constraints. This can be observed, for example, with tweets, status updates and similar texts in the face of requirements for brevity. In addition, the adaptation of language may be driven by a need for rapid production and interpretation of information. In developed societies, people must cope with a vast and growing amount of information on a daily basis. This leads us to consider whether language processing in the future may require something more than statistical text processing currently has to offer.

One constant in the process of change is likely to be the basic semantics of placing related things near to each in space and time. The relations are predictable, but the nature of related units is indeterminate. A lexicalist approach that analyzes patterns in condensed language corpora (such as collections of tweets) may provide insight into changes as they evolve, for example the collapse of phrases into acronyms that are used as new words (btw, lmao, lol etc.). These adaptations may demand models of retrieval that are are similar to the ones used at present, but based on characters instead of words.

Moving further into the future, as language becomes increasingly compressed, there may be greater reliance on context to disambiguate meaning. With the search tools currently available, this means that IR will only become harder. Context may be discovered in the form of brief communications that immediately precede utterances of interest, such that meaning is built up not from the relationships between words or phrases, but from connections between sentences and ideas. If this occurs, the window of interest around language processing in IR may widen to encompass the study of pragmatics and topic focus. While current text processing techniques appear to be reaching a zenith, these new frontiers offer almost infinite potential for further development.

# Glossary

**adjunct** An optional element of sentence structure.

**adposition** A class of words that typically express spatial or temporal relations (e.g. *in, under, towards, before*) or mark syntactic functions and semantic roles (e.g. *of, for*). An adposition combines with another constituent (called its complement) to form a phrase, relating the complement to the context in which the phrase occurs. Adpositions include prepositions that precede a complement, and postpositions that follow a complement'.

**adverb** A word that modifies or qualifies an adjective, verb, other adverbs or clauses, typically expressing a relation of place, time, circumstance, manner, cause, degree, etc. (e.g. *quickly, very, quite, well, then, there*).

**argument** In linguistics, arguments are expected or required dependency relations for a predicate. For example, take the predicate '*give*'. It requires one argument that is an actor (e.g. '*Bob gives*'). It also predicts other arguments that are not required, such as a theme and goal (e.g. '*Bob gives a lecture*' and '*Bob gives a lecture to help the students*'). An argument completes the meaning of a predicate, just as a complement completes the meaning of an expression.

**chunking** In computational linguistics, chunking is a type of shallow parsing that provides partial syntactic structure of sentences. It is typically more efficient than full parsing, and used to detect noun, verb, and prepositional phrases.

**collection** In IR, a corpus from which relevant documents may be selected in response to a query.

**complement** A word, phrase or clause that is necessary to complete the meaning of an expression. Complements and arguments largely overlap in meaning. However, in some definitions, subjects (e.g '*he*' in '*he reads*') are complements and in other definitions they are not.

**complex nominal** A combination of nouns, possibly also including prepositions and determiners such as {*of, in, a*}.

**compositional semantics** A way to define the meaning of a phrase as a function of the meanings of its parts and the way they are put together.

**compound** Combinations of words that are either hyphenated (e.g. '*data-base*') or joined (e.g. '*database*'). They are frequently indistinguishable in meaning from phrases constructed using the same words and occur with equal regularity. For example, *'database', 'data-base'* and *'data base'* are equally probable in a large corpus (Krovetz, 1995).

**compound noun** A combination of two or more nouns e.g. '*data base*'. As with all compounds, the nouns may be separated by whitespace, hyphenated or combined into a single word. However, compound nouns are often combinations of nouns that are unlikely to appear as a single or hyphenated word in English, e.g. '*orange juice*'.

**conjunction** A word that links elements of a sentence together. Conjunctions are usually divided into coordinating conjunctions, such as *and, but* and *or*, that link elements of equivalent status, and subordinating conjunctions, such as *because, if, when* etc. that identify an element within a larger construction (Bauer, 2007).

**context free grammar** A grammar represented by a finite set of rules of the form $X \rightarrow Y$ where $X$ and $Y$ are low level grammatical categories. Phrase structure grammars are context free grammars.

**coreference resolution** The identification of cases in which two text units refer to the same entity. For example, in '*Tweety is a penguin. He cannot fly.*', both '*he*' and '*Tweety*' refer to the same bird.

**crossing** Crossing dependencies occur when a word or phrase is separated from another word or phrase that it modifies in such a way that a direct connection cannot be established between the two without incurring crossing lines in the representative tree structure.

**deep structure** In Chomsky's transformational grammar, deep structure represents the logical structure of sentences.

**dependency** In linguistics, a syntactic or semantic relation between a head (or governor) and a dependent word, morpheme or phoneme, such as defined in a depen-

dency grammar. In IR, a dependency is a statistical relation between two words or phrases, where the syntactic or semantic nature of the relation is unspecified.

**dependency grammar** A type of grammar that defines syntactic structure using binary asymmetric relations between words. For each relation, one word is the head (or governor) and the other is the dependent. The relations are primarily semantic in nature, even though they describe syntax. No relations are defined for phrases (such as those seen in phrase structure grammar).

**descriptive linguistics** A description (at a given point in time) of a language with respect to its phonology, morphology, syntax and semantics without value judgments (e.g. linguistic annotations). The term 'descriptive linguistics' is sometimes used specifically to refer to Structuralism (a major school of descriptive linguistics).

**determiner** A word that expresses a notion such as quantity, definiteness or possession e.g. {*this, his, one, every*} (Bauer, 2007).

**effectiveness** In IR, effectiveness refers to the ability of an IR system to satisfy a user's information need. There are a number of measures of effectiveness, of which the most frequently mentioned are recall and precision.

**ellipsis** The omission from speech or writing of a word or words that are superfluous or able to be understood from contextual clues.

**finite clause** A clause containing a finite verb. A clause is the smallest text unit that can express a complete proposition, typically a subject and a predicate e.g. '*he ran*'. Simplifying greatly, a finite verb is a verb that is fully inflected e.g. {*run, runs, ran*}.

**functionalism** A school of linguistics holding that language structure is conditioned by its use (function) as a means of communication. Functionalism aims to explain the relationship between valency roles and the grammatical and communicative organization of sentences. It is a subgenre of Structuralism (Graffi, 2005).

**grammar** The system and structure of a language, or languages in general, usually consisting of syntax and morphology and sometimes also phonology and semantics (Proffitt, 2013).

**grammatical** Of or pertaining to the rules of grammar for a language. Grammatical sentences are those that accord with a given grammar (set of grammatical rules).

**grammatical category** A role played by a text unit. Grammatical categories include, but are not limited to: sentence, noun, verb, adverb, adposition (preposition or postposition), and their phrasal projections (e.g. noun phrases, verb phrases) etc.

**headedness** In linguistics, the state or quality of having a particular type of head. There are many different ways to determine headedness, or identify a head, that use morphological, syntactic and semantic criteria. For example, a head may be a functor, such as a predicate, that takes semantic arguments.

**idiom** A phrase in which an exact combination of words is fixed, and the meaning of the whole expression cannot be predicted from the usual meanings of the component words. The substitution of one word changes the meaning of the word combination. For example, '*kick the bucket*' means '*to die*' and this cannot be predicted from the usual meanings of the individual words '*kick*' and '*bucket*'. The substitution of a word, as in '*kick the pail*', demands a literal, or compositional, interpretation.

**information need** In IR, an information need is an abstract concept perceived by a user. The user's perception of an information need can change during the course of a searching session so it is not necessarily static (unlike queries, which are fixed).

**informative** In IR, informative terms represent an information need and discriminate between relevant and non-relevant documents in a retrieval collection.

**lexical relation** A word association that is interpreted using the semantics of words and encyclopaedic knowledge of the world around us, rather than the semantics of words and syntactic relations. The words do not need to be syntactically related or contiguous.

**lexicalism** Lexicalism is a theory of language that is broadly 'of or relating to words'. It assumes that meaning is formed irregularly and more or less directly grasped from entries in a lexicon without consideration for how the parts are assembled. Like dependency grammars, lexicalism focuses on relations between words, not relations between phrases, yet holds that meaning is essentially phrase based.

**linguistic** Referring to the study of linguistics, including the study of language form, meaning and variation. Phonetics, phonology, morphology, syntax, semantics, sociolinguistics, and pragmatics are branches of linguistics.

**linguistic representation** A structured interpretation of language that can include assignment of grammatical or syntactic roles, assignment of relations between words and phrases, and identification of the type of those relations, among other attributes. A representation may be graphical or flat e.g. word tagging.

**link grammar** Link grammar builds relations between pairs of words in a manner similar to dependency grammars. There are two basic parameters to these relations: directionality (+ or -) and distance. In contrast, dependency grammars include head-dependent relationships and lack directionality in the relations between words (Sleator and Temperley, 1993).

**mentalist** In linguistics, mentalist theories describe internalized grammars that underlie linguistic behaviour. They focus on the language that we 'should' speak, rather than the language we do speak.

**model-theoretic semantics** Roughly, a model-theoretic semantics assigns meanings to terms or symbols from a universe of possible elements, and 'true' or 'false' values to propositions constructed for those elements, by the application of a recursively specified set of interpretation functions in various ways.

**morpheme** Morphemes are the smallest semantically meaningful units in language, including prefixes, affixes, suffixes such as '*-ing*' in '*finding*', and freestanding words such as '*in*'.

**ngram** A contiguous sequence of *n* words in text or speech. In the previous sentence, '*sequence of*' if one of many possible ngrams.

**paradigmatic** Relating to the functional role, or class value, of a text unit. Paradigmatic relations hold between members of conceptual sets. For example, synonyms are a paradigmatic set.

**phoneme** Phonemes are the smallest units of sound used to make contrasts between utterances.

**phrase** In linguistics, a phrase refers specifically to a multi-word unit in which words are linked by syntactic relations. In IR, phrases are typically multi-word units in which words form a continuous sequence in text. Phrases are a subset of terms..

**phrase structure grammar** A type of grammar that emphasizes compositionality, such that the meaning of a phrase is a function of the meaning of its parts and the way they are put together syntactically. Phrase structure grammars use a finite set

of rules that compose grammatical categories such as noun, adjective, noun and verb phrases etc. Phrase structure theory is the dominant theory in modern linguistics.

**polysemous** With respect to words, having multiple meanings.

**precision** In IR, the number of relevant items retrieved divided by the total number of items retrieved.

**predicate** Based on propositional logic, a predicate is an expression that can be true of something. This includes verbs, nouns and certain adjectives. It can be true that that someone '*is reading*', for example, or that something '*is organised*' or '*is a book*'. Another definition derives from predicate calculus. A predicate is seen as a function over arguments. The predicate assigns a property to a single argument, or relates two or more arguments to each other.

**prescriptive linguistics** An account of how a language should be used, instead of how it is actually used, by prescription of the 'correct' phonology, morphology, syntax and semantics.

**proximity measure** In IR, proximity measures are a way of specifying that two query terms appear within some distance of each other in a document, as measured by a number of intervening terms or containment within a structural unit such as a sentence.

**pseudo relevance** In IR, pseudo relevance is relevance determined by some method other than explicit user judgment. Typically, pseudo relevant feedback identifies the top $k$ documents returned by an IR system for a query.

**query reformulation** Alteration of query representation to improve query effectiveness. Reformulation techniques include tokenization, stopping, stemming, term selection, query expansion, query reduction and term weighting.

**recall** In IR, the number of relevant items retrieved divided by the number of relevant items in the retrieval collection.

**relevance** In IR, the most common definition of relevance is a binary relation between a document and a user request. A document is considered relevant if it meets an information need, where an information need is perceived by a user and a request is a lexicalisation of that need.

**relevance feedback** In retrieval, relevance feedback identifies documents that users explicitly judge to be relevant to a query.

**representation** In the context of IR, a static interpretation of an information need, request, document or data, often in structured form. For example, in the case of text queries, a representation may be a parse tree or a proximity window over a set of words.

**request** In IR, an expression of an information need. A request is sometimes synonymous with a query submitted to a search engine, but can be distinct. For example, verbose natural language requests are converted into queries prior to search.

**semantics** Meaning in language. Semantics focuses on the relation between signifiers such as words, phrases, signs and symbols, and what they denote in the world.

**stemming** The removal of word inflections such as suffixes.

**stop structure** Phrases that do not provide any information about the topic of a text, as typically found in queries e.g. '*Can any one help me out with a*'.

**stopword** Highly frequent words that tend to appear in any text, irrespective of the text topic, e.g. {*with, the, do, if, he*}.

**Structuralism** The dominant school of linguistics prior to 1957. Structuralism takes a systemic approach that views language as a 'system of signs', each of which has no intrinsic value but whose value is determined solely by its relationships with the other members of the system (Graffi, 2005).

**subcategorization** In linguistics, a definition of the number and types of arguments that lexical items (often, but not always, verbs) require in order to achieve a 'minimal maximal projection'. Subcategorization is almost synonymous with the concept of valency but is often associated with phrase structure grammars. For example, the verb '*gives*' requires three arguments, even if they are sometimes unspecified e.g. '*John gave the students a lecture*' or '*John gave a lecture*' (in which the audience of the lecture is not specified but understood to exist).

**symbolic linguistics** The study of language using written symbols, rather than some other means of representation such as audio recording.

**syntagmatic** Relating to the environment of a text unit. Syntagmatic relations hold between ordered sequences in space or time, such as words in text or speech. For example, ngrams are syntagmatic units.

**syntax** Principles, rules or patterns that govern the arrangement of words and phrases to create well-formed sentences in a language.

**term** As defined in this dissertation, a text unit containing one or more words.

**term discrimination** A process by which language terms are distinguished from each other with respect to some specific criterion or point of reference. In IR, the criterion is relevance to an information need.

**term selection** In IR, the identification of a small subset of all possible terms that maximize the retrieval effectiveness for an information need. Query-based term selection identifies these terms from within a query.

**tf.idf** A common weighting scheme in IR that balances word salience with discriminatory ability. There are many variants on the basic formula that multiplies Term frequency (*tf*) by inverse document frequency (*idf*).

**trace** In transformational grammar, a trace is a syntactic placeholder for an empty (non-vocalized) grammatical category. It occupies a position in syntactic structure, and can therefore be leveraged to identify word and phrase relations.

**transitivity** The syntactic requirements and restrictions for verbs, for example with respect to direct and indirect objects.

**triangulation** As defined in this dissertation, the process of selecting two terms that share a common word, or the state of having triangulated terms in a query.

**valence** Requirements and restrictions placed on the type and number of arguments and complements.

**verbose** In IR, refers to queries that express a complex or specific information need, including *long keyword* queries and *natural language* queries.

**word** A word is a meaningful element of language that is normally moved as a unit in text, has elements with a fixed order, and cannot have another word freely inserted within it. Words are often separated by whitespace in English, but this is not always the case, e.g. '*database*', '*data base*'.

**word association** A relation between words such that the presence of one word provides information about semantic meaning of the other, the grouping of words to form the semantic meaning of some larger unit, or the possibility that the other word will be observed. Word association is often, but not always, defined in a general context.

# Appendix A

# Stoplists

The following stoplist of 418 words was used throughout this dissertation:

all, whoever, hath, slung, hereto, go, slunk, seemed, whose, stave, to, whatsoever, under, inwards, include, sent, worse, far, exception, every, yourselves, sang, round, be, wherefore, notwithstanding, further, yippee, even, what, henceforth, above, thereabouts, ever, never, here, spake, whichsoever, let, others, alone, along, wherever, hither, via, till, indoors, whereunto, yourself, use, from, spoke, would, sake, next, few, therefore, themselves, thru, until, more, becomes, hereby, herein, everywhere, must, me, none, whensoever, this, anywhere, can, ms, mr, my, something, want, exclude, provide, get, how, instead, may, after, hereupon, ff, such, a, whenever, maybe, rather, so, worst, excepted, exclusive, indeed, over, whilst, including, still, thyself, its, before, whereon, thence, selves, inward, whereof, meantime, choose, ours, might, then, them, someone, somebody, thereby, thee, ye, underneath, they, front, now, day, nor, hereafter, always, whither, each, upward, everyone, whosoever, doing, year, our, beyond, slew, out, shown, nowadays, furthermore, since, excepting, howsoever, forth, thereupon, whereinto, sideways, quite, whereupon, besides, anyhow, could, ltd, hence, onto, first, already, seeming, thereon, spoken, thereafter, thereof, one, another, doesnt, little, slept, anyone, their, too, mostly, that, et, nobody, somewhat, herself, than, albeit, kind, double, see, i, were, toward, and, beforehand, thereto, have, need, seen, seem, saw, any, hitherto, these, latter, also, which, towards, unless, though, who, most, amongst, plenty, nothing, why, shalt, kg, wherewith, noone, sometimes, km, mrs, whomsoever, ugh, anyway, outside, should, only, do, hindmost, his, meanwhile, cannot, during, him, seldom, she, through, where, farthest, namely, are, said, wow, whereabouts, halves, behind, unable, between, neither, nope, across, we, how-

ever, staves, both, cos, last, thou, many, whereafter, according, against, somewhere, became, whole, otherwise, among, afterwards, seems, cf, whatever, hast, moreover, throughout, cu, smote, been, whom, much, hardly, thrice, latterly, else, doth, hers, those, myself, save, thenceforth, unlike, wilt, will, while, almost, is, thus, it, itself, dual, in, ie, if, etc, perhaps, same, wherein, beside, several, week, used, upon, supposing, off, whereby, thy, nevertheless, no, well, anybody, without, very, the, yours, lest, just, less, being, not, farther, yet, dost, sprang, had, except, hereabouts, has, ought, around, whichever, using, apart, like, excluding, either, become, whomever, therein, canst, because, often, some, somehow, ourselves, vs, for, per, everything, does, everybody, sprung, nowhere, although, by, on, about, ok, anything, of, whence, or, contrariwise, seeing, own, formerly, into, within, down, wherefrom, wheresoever, your, her, there, inasmuch, whereto, forward, was, spat, himself, elsewhere, enough, becoming, but, with, he, whether, inside, up, us, whoa, below, certain, thereabout, am, an, as, sometime, at, av, inc, again, nonetheless, whereas, when, whereat, other, whew, you, really, insomuch, included, upwards, furthest, together, once, did, done

The following supplemental stoplist of 18 words was added for description topics (and not title queries):

can, definition, describe, description, discuss, document, documents, find, give, identify, information, kinds, provide, relevant, s, what, who, u

# Appendix B

# Textual Economy in Queries

The Table on the following pages surveys 22 description topics for which neither the phrase structure nor dependency relations of the most probable parse were able to detect at least one pseudo-informative word association. Pseudo-informative word associations are assumed between words in a corresponding title query. For example, given the query '*profiling motorists police*', word associations are assumed for {*profiling motorists, profiling police, motorists police*}. Examples here suggest that there are many cases in which semantic associations cannot be trivially identified using syntax.

| QID | Title Query | Description Topic | Word Mapping | Title Query Word Association | Pseudo-Informative Word Association (*= not detected) |
|---|---|---|---|---|---|
| 306 | african civilian deaths | How many civilian non-combatants have been killed in the various civil wars in Africa? | african → africa<br>deaths → killed | african civilian<br>african deaths<br>civilian deaths | africa civilian<br>africa killed<br>*civilian killed |
| 318 | best retirement country | Aside from the United States, which country offers the best living conditions and quality of life for a US retiree? | retirement → retiree | best retirement<br>best country<br>retirement country | *best retiree<br>*best country<br>*retiree country |
| 359 | mutual fund predictors | Are there reliable and consistent predictors of mutual fund performance? | | mutual predictors<br>mutual fund<br>fund predictors | mutual fund<br>*fund predictors |
| 391 | rd drug prices | Identify documents that discuss the impact of the cost of research and development (R&D) on the price of drugs. | | rd prices<br>*rd drug<br>drug prices | *rd drug<br>drug prices |
| 411 | salvaging shipwreck treasure | Find information on shipwreck salvaging: the recovery or attempted recovery of treasure from sunken ships. | | salvaging shipwreck<br>salvaging treasure<br>shipwreck treasure | salvaging shipwreck<br>*salvaging treasure<br>*shipwreck treasure |
| 432 | profiling motorists police | Do police departments use profiling to stop motorists? | | profiling motorists<br>profiling police<br>motorists police | *profiling motorists<br>*profiling police<br>*motorists police |
| 604 | lyme disease arthritis | What evidence is there to link tick-borne Lyme disease with arthritis? | | lyme disease<br>lyme arthritis<br>disease arthritis | lyme disease<br>*lyme arthritis<br>*disease arthritis |

| | | | | | |
|---|---|---|---|---|---|
| 610 | minimum wage adverse impact | Claims made by US businesses regarding the adverse impact on their business of raising the minimum wage. | | minimum adverse<br>minimum impact<br>minimum wage<br>wage adverse<br>wage impact<br>adverse impact | minimum wage<br>*wage adverse<br>*wage impact<br>adverse impact |
| 611 | kurds germany violence | What violent activities have Kurds, or members of the Workers Party of Kurdistan (PKK), carried out in Germany. | violence → violent | kurds germany<br>kurds violence<br>germany violence | *kurds germany<br>*kurds violent |
| 624 | sdi star wars | What are the pros and cons of developing the Strategic Defense Initiative (SDI) also known as “Star Wars”? | | sdi star<br>sdi wars<br>star wars | *sdi star<br>*sdi wars<br>star wars |
| 633 | welsh devolution | What is the history of the Welsh devolution movement | | welsh devolution | *welsh devolution |
| 666 | thatcher resignation impact | Find documents that discuss the impact Prime Minister Margaret Thatchers’ resignation may have on U.S. and U.K. relations. | thatcher → thatchers | thatcher resignation<br>thatcher impact<br>resignation impact | thatchers resignation<br>*thatchers impact<br>resignation impact |
| 734 | recycling successes | What recycling projects have been successful? | successes → successful | recycling successes | *recycling successful |
| 750 | john edwards womens issues | What are Senator John Edwards’ positions on women’s issues such as pay equity, abortion, Title IX and violence against women. | | john womens<br>john issues<br>john edwards<br>edwards womens<br>edwards issues<br>womens issues | john edwards<br>*edwards womens<br>*edwards issues<br>womens issues |

| | | | | | |
|---|---|---|---|---|---|
| 754 | domestic adoption laws | Provide any legal information about domestic human adoption. | laws → legal | domestic laws<br>domestic adoption<br>adoption laws | domestic adoption<br>*adoption legal |
| 762 | history physicians america | Who have been considered “doctors” since the first European settlement in America? | physicians → doctors<br>history → settlement | history america<br>history physicians<br>physicians america | settlement doctors<br>*doctors america |
| 789 | abandoned mine reclamation | Find information on abandoned mine reclamation projects. | | abandoned reclamation<br>abandoned mine<br>mine reclamation | *abandoned mine<br>*mine reclamation |
| 794 | pet therapy | How are pets or animals used in therapy for humans and what are the benefits? | | pet therapy | *pet therapy |
| 808 | north korean counterfeiting | What information is available on the involvement of the North Korean Government in counterfeiting of US currency. | | north counterfeiting<br>north korean<br>korean counterfeiting | *north korean<br>*korean counterfeiting |
| 829 | spanish civil war support | Provide information on all kinds of material international support provided to either side in the Spanish Civil War. | | civil support<br>spanish civil<br>spanish war<br>civil war<br>spanish support<br>war support | *spanish civil<br>spanish war<br>civil war<br>*spanish support<br>*war support |
| 836 | illegal immigrant wages | What level of wages are paid to illegal immigrants? | | illegal immigrant<br>illegal wages<br>immigrant wages | illegal immigrants<br>*illegal wages<br>*immigrants wages |

# Appendix C

# User Study Guidelines

This appendix contains the instructions given to users for two tasks:

- **user nominated terms:** Users are asked to identify all word combinations in a query that are representative with respect to an information need (represent all or part of an information need). No candidate terms are provided.
- **user annotated terms:** Users are prompted with a list of candidate word combinations and asked to judge whether each combination is informative with respect to an information need.

## C.1 Task 1: User nominated terms

In the first task, you will be given a list of queries. For each query, you are asked to submit the word combinations that you think capture the meaning of the query. A word combination can be any single word in the query, or a group of up to four words in the query. Please provide all word combinations that you think represent the query, in compliance with the following guidelines:

- Combinations can only use words exactly as they appear in the query.
- Words can be used in any order.
- Combinations cannot contain words from the following list: *a, am, an, and, are, as, at, be, been, being, by, did, do, does, doing, done, for, from, had, have, has, he, in, if, is, it, its, of, on, or, that, the, to, was, were, will, with*.
- Shorter combinations of 1-3 words are strongly preferred even though groups of 1-4 words are permitted.
- Some query words might not be used in any combination.

The word combinations may represent the query as a group or set. This means that each combination is not required to summarize the entire query. For example:

**Q:** *Who is the current president of the United States of America?*

You might write some, or all, of the following word combinations (and possibly other combinations as well):

- America president
- president United States
- president
- United States America

The following would not be permitted:

- currently president America
- politics America

**The task process:** When you run the task code (see instructions), you will be presented with a query and asked if you want to submit a word combination or move on to the next query. To submit a word combination, write it at the prompt, and then hit "enter". Do this for as many word combinations as you think are important, hitting "enter" after each one. When you are ready for the next question, type "n" (for "next", no quotation marks) followed by "enter". You will be provided with the next query.

If you make a spelling mistake or other error and only notice after you have hit "enter", then you can input "b" (back) and the previous input will be deleted. This works like the "undo" key in your word processor, but you can only delete input for the current question. Once you enter "n" (next) all your input for a question is saved and cannot be changed.

You can quit at any time by entering "q" (quit). When you run the task code again, the system will automatically re-start at the point where you left off.

## C.2 Task 2: User annotated terms

In the second task, you will be given a query and a list of word combinations. A word combination might be a single word, or a group of up to four words. Word combinations are not necessarily grammatical units of text, and the words do not necessarily

appear in a logical order. You are asked to annotate whether each combination is informative with respect to the query. A combination is informative if either (1), (2) or (3) is true:

1. If the combination is used as a new query, it is more likely to retrieve documents that are relevant to the original (given) query, than documents that are not relevant.
2. The combination is a meaningful part of the query e.g. it is likely to retrieve documents that refer to some part of the topic of the query.
3. Given a set of documents that are partially relevant to the query, the combination would help to isolate those documents that are most relevant (even if the combination is very frequent and would likely retrieve irrelevant documents from an unrestricted document collection).

For example, consider the following query:

**Q:** *Find information on the breakup of Czechoslovakia into the Czech Republic and Slovakia and its social and political impact on the two countries' people.*

The following are examples of combinations that meet condition (1). They are likely to retrieve relevant documents:

- breakup Czechoslovakia impact
- breakup Czechoslovakia political
- Czech Slovakia impact
- Czechoslovakia Slovakia social people

The following are examples of combinations that meet condition (2). They are likely to retrieve documents that are relevant to some part of the query:

- Czech Slovakia
- breakup Slovakia
- republic Slovakia impact

The following are examples of combinations that meet condition (3). They would help to isolate the most relevant documents in a partially relevant set:

- Czechoslovakia

- Czech Republic
- Slovakia

The following are examples of combinations that are not informative:

| **Word Combination** | **Why it is Uninformative** |
|---|---|
| political | Ambiguous; this word is not a representative part of the query. |
| Czechoslovakia countries people | Unnecessarily long due to the presence of "countries". In general, combinations may be too long when one of the words replicates information given by another word (Czechoslovakia *is* a country; a dachshund *is* a dog). They can also be too long when one or more of the words is likely to appear in documents about any number of topics e.g. "find". |
| breakup people | Misleading; could refer to the breakup of a personal relationship, not a country. Note that some combinations are misleading in subtle ways, e.g. "conviction Feb 1993" could be about a sentence passed in Feb 1993, or conviction for a crime that took place in Feb 1993. |
| impact social political | Restrictive; documents are more likely to cover either social impact or political impact, not both. Restrictive combinations can arise from lists of examples e.g. "social, political" or alternatives connected by *and, or* e.g. "social and political". Combinations including an unusual word, such as "scrutinize" can also be too restrictive. |

You may identify as many, or as few, informative word combinations as you see fit. There will be cases in which it is difficult to decide whether a combination is informative. Do not be intimidated if you feel you are making a lot of guesses. There is no correct answer and you are asked to simply use your best judgment, erring on the side of annotating a combination as informative if you are unsure.

During this task, it may simplify matters for you if you imagine the types of word combinations that would be informative for the query before reading the candidate word combinations. You can then seek to identify combinations that are similar to the ones you imagined, and their variants. Always bear in mind that there may be some

informative word combinations that you didn't think of in advance, and combinations that are not in the list of candidates. Do not annotate a combination as informative just because you cannot find the combination you wanted, or because it was the 'best of a bad bunch'. Only identify combinations as informative if you think they are informative. It is OK not to annotate any combinations as informative if none of the provided combinations meet the conditions (1), (2) or (3) of our definition.

**The task process:** When you run the task code, you will be provided with a query and a list of candidate word combinations. Each combination will be numbered. To make it easier to read the list of candidate combinations, the system can present you with a list of no more than 20 candidate combinations at a time (use "-a" in the command, as per the instructions). In the case when there are more than 20 candidate combinations for a given query, the system will repeatedly present you with the same query and a different subset of 20 candidate combinations for consideration. This will be repeated until you have processed all the candidates for that query. Not using "-a" makes the candidate combinations appear as a single list.

At the bottom of the list of candidate combinations you will be prompted to either enter the number of a combination you think is informative, or move on to the next question. For each combination you identify as informative, type its number at the prompt and hit "enter". Do this for every combination you think represents the query well. The system will process the information you entered and immediately present you with the same prompt, which asks you to either enter another combination number, move onto the next question by typing "n" (no quotation marks, followed by "enter"), or quit by typing "q" (then "enter").

Move onto the next question by typing "n", followed by "enter", when you believe you have identified all the informative combinations in the current list. You may also quit the task at any time. If you quit, you can start again at any time, and the code will automatically pick up at the point where you left off.

# Appendix D

# Results: Combination of Methods of Word Association

Figures on the following pages show the relative importance of word association methods used in pairwise combinations for the prediction of terms that improve NDCG.

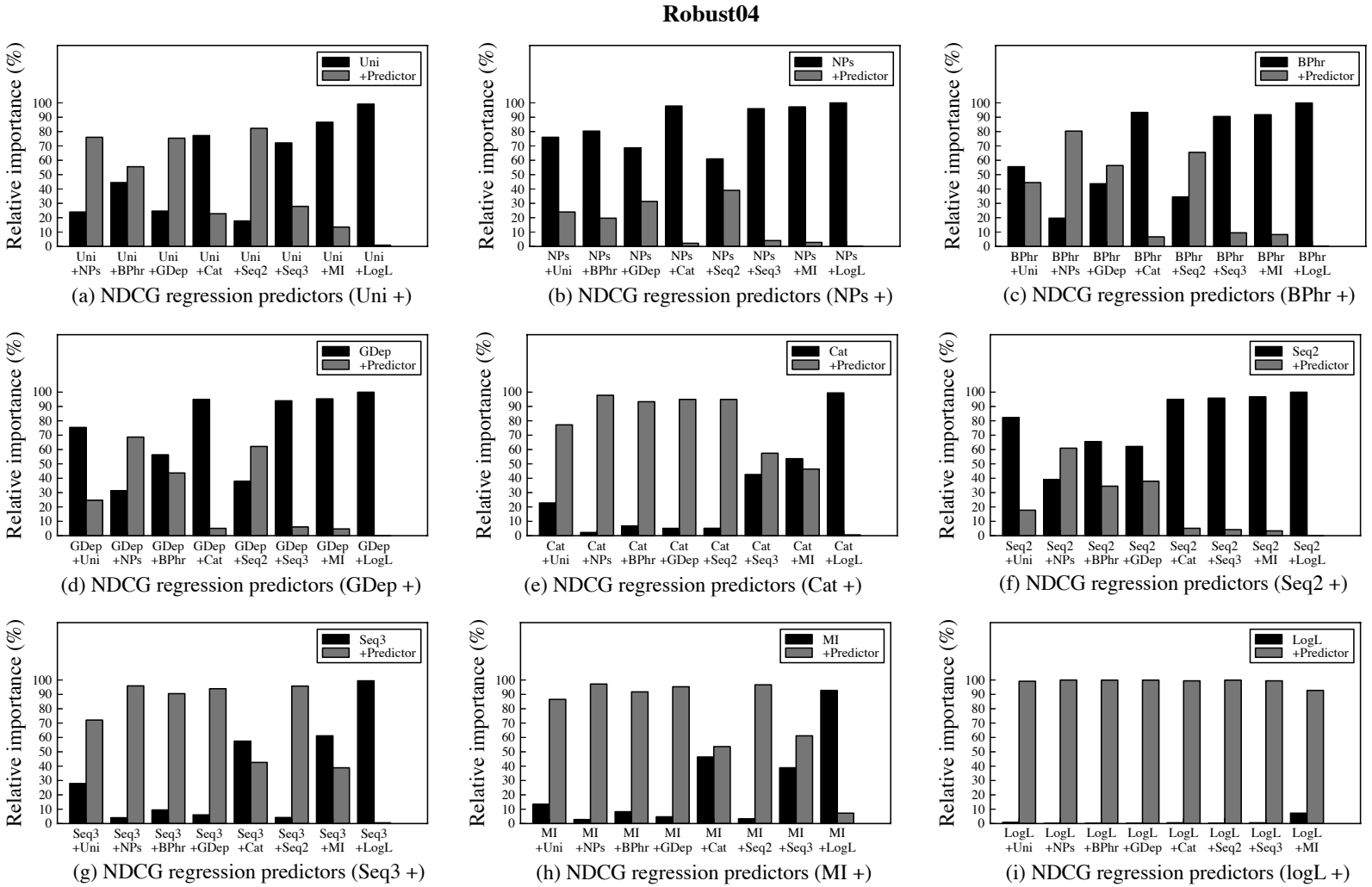
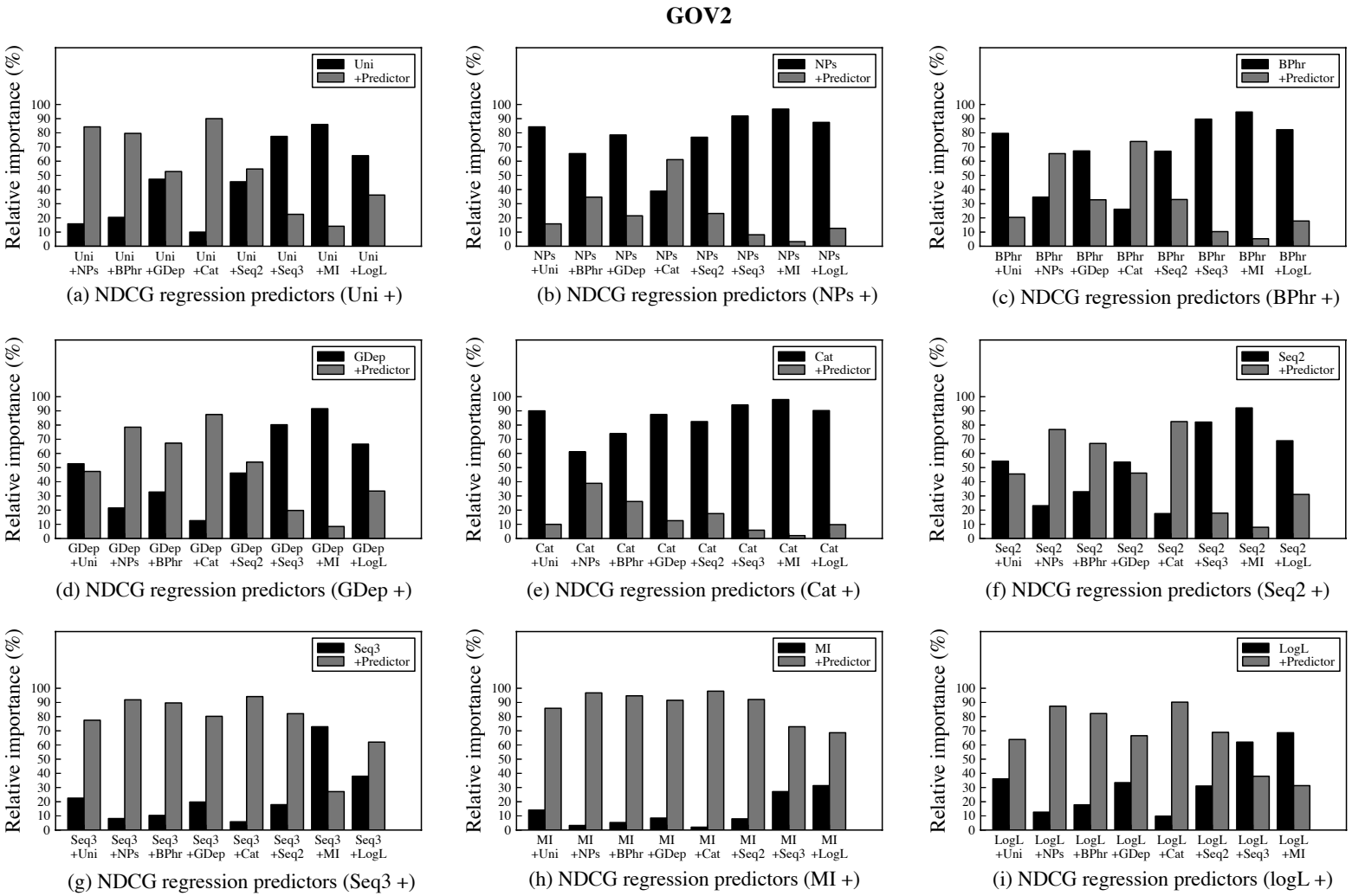



Figure D.1: Relative importance of word association methods used in combination for the prediction of terms that improve NDCG (Robust04 queries).




(a) NDCG regression predictors (Uni +)
(b) NDCG regression predictors (NPs +)
(c) NDCG regression predictors (BPhr +)
(d) NDCG regression predictors (GDep +)
(e) NDCG regression predictors (Cat +)
(f) NDCG regression predictors (Seq2 +)
(g) NDCG regression predictors (Seq3 +)
(h) NDCG regression predictors (MI +)
(i) NDCG regression predictors (logL +)

Figure D.2: Relative importance of word association methods used in combination for the prediction of terms that improve NDCG (GOV2 queries).

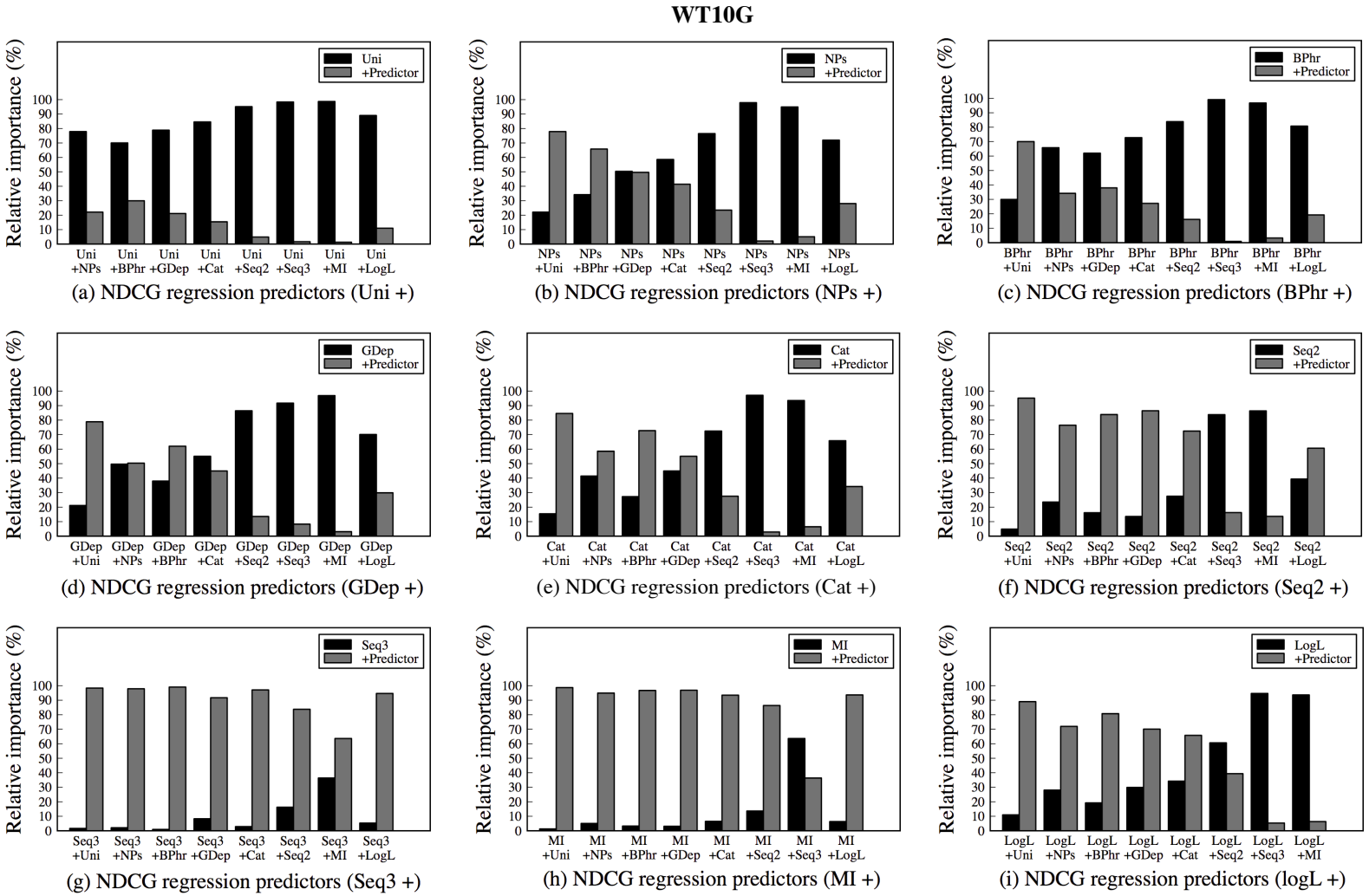



Figure D.3: Relative importance of word association methods used in combination for the prediction of terms that improve NDCG (WT10G queries).

# Appendix E

# Results: Combination of Terms and Methods of Word Association

Plots in this Appendix show the overlap in terms that deliver an increase in percent change NDCG for particular combination types and multiple word association methods. To improve legibility, only data for terms that improve NDCG scores are shown. Results are given for both individual terms and combinations of two or three terms identified by one given method (the base method). Plots on the following pages show the percentage of these terms identified by other methods.

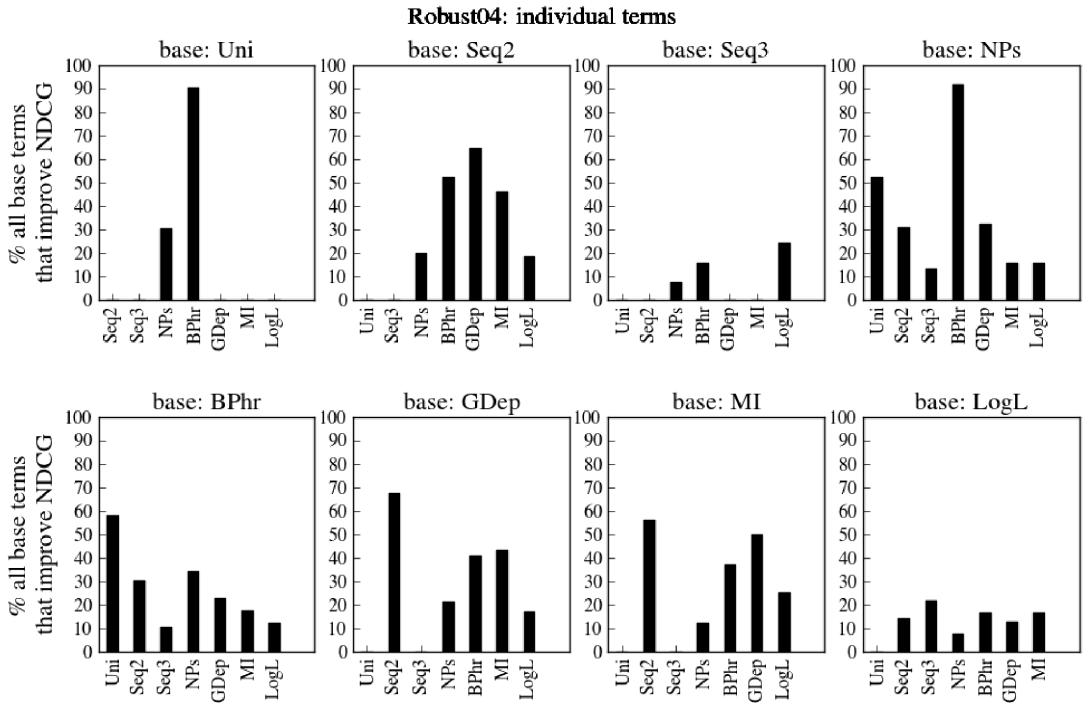



Figure E.1: Method overlap with individual terms for Robust04.

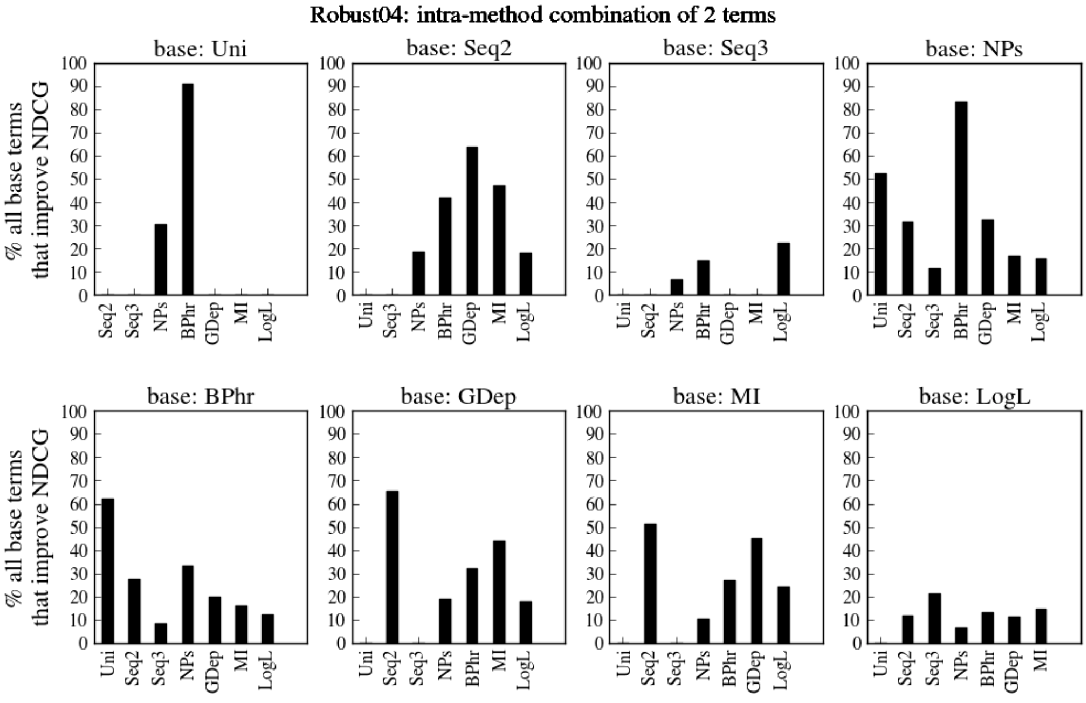



Figure E.2: Method overlap with combinations of two base terms for Robust04.

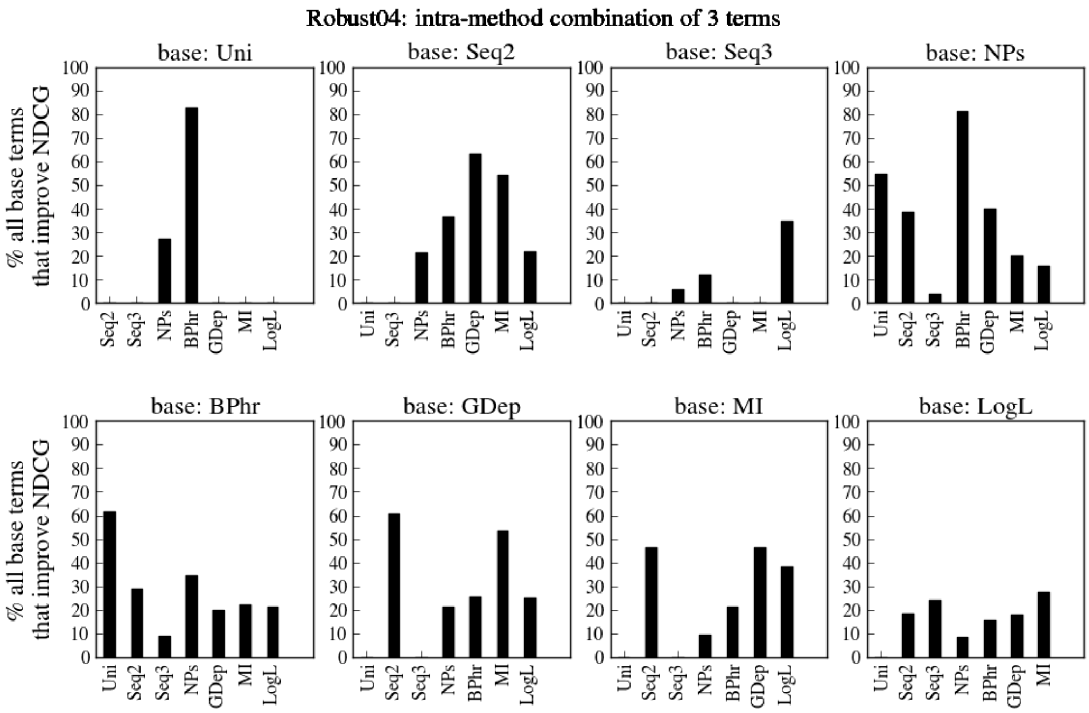



Figure E.3: Method overlap with combinations of three base terms for Robust04.

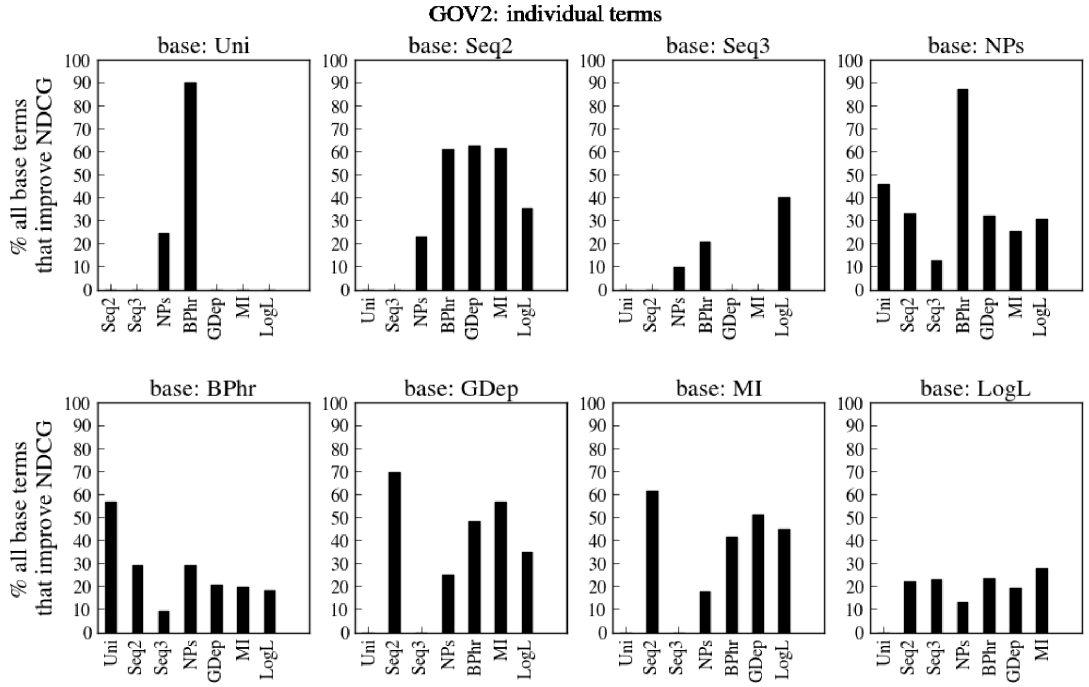



Figure E.4: Method overlap with individual terms for GOV2.

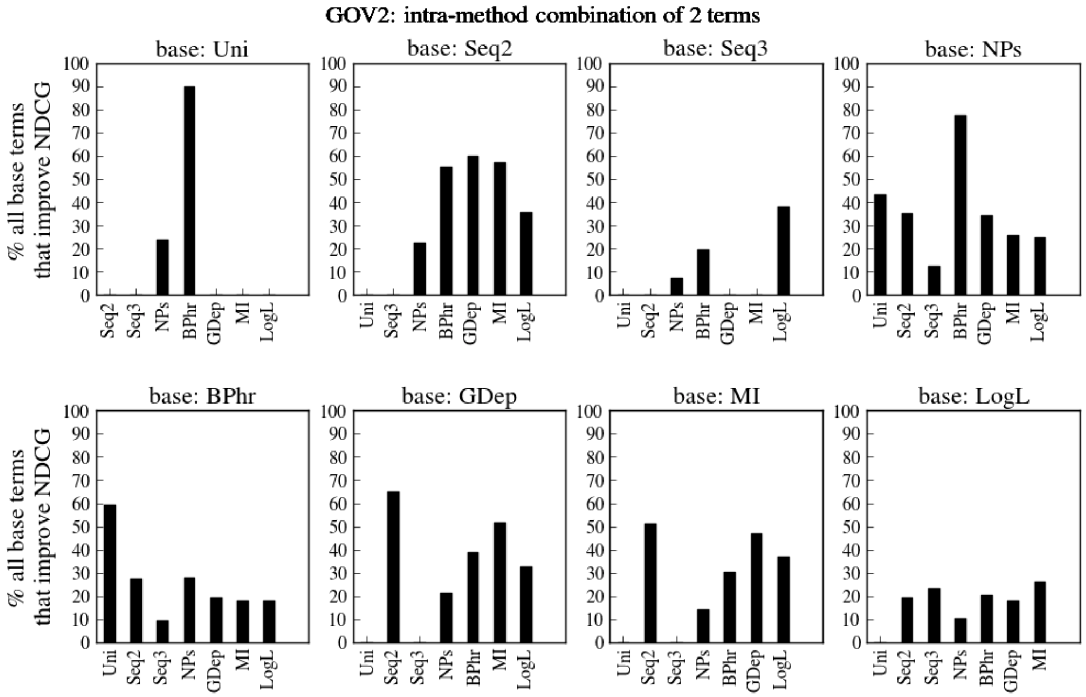



Figure E.5: Method overlap with combinations of two base terms for GOV2.

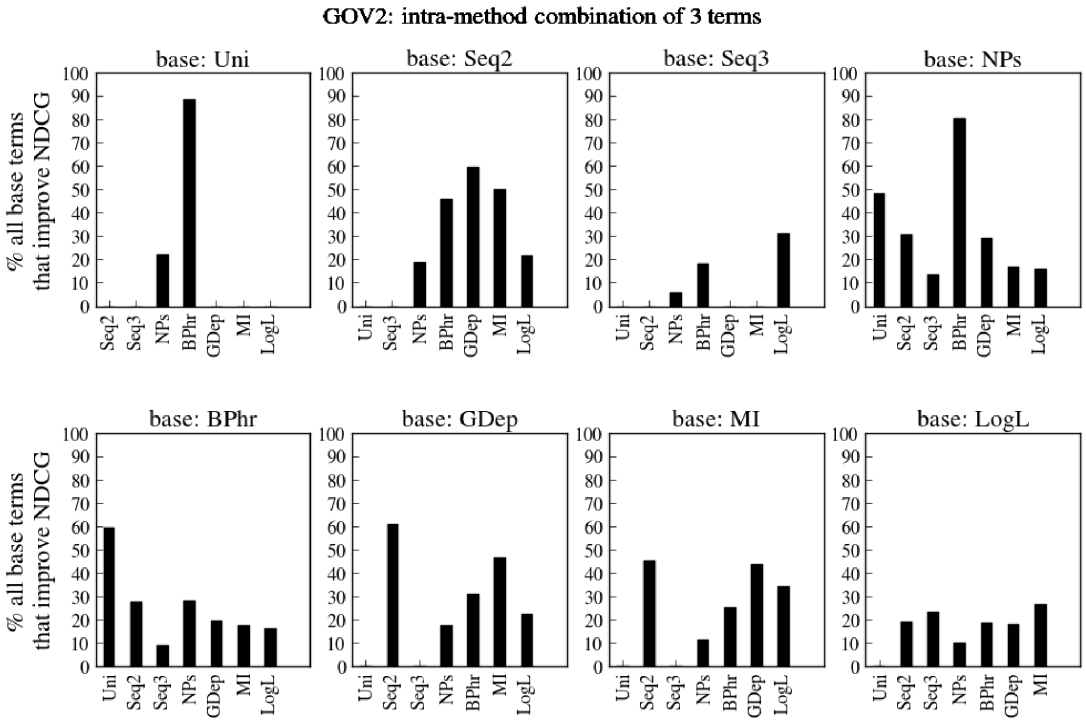



Figure E.6: Method overlap with combinations of three base terms for GOV2.

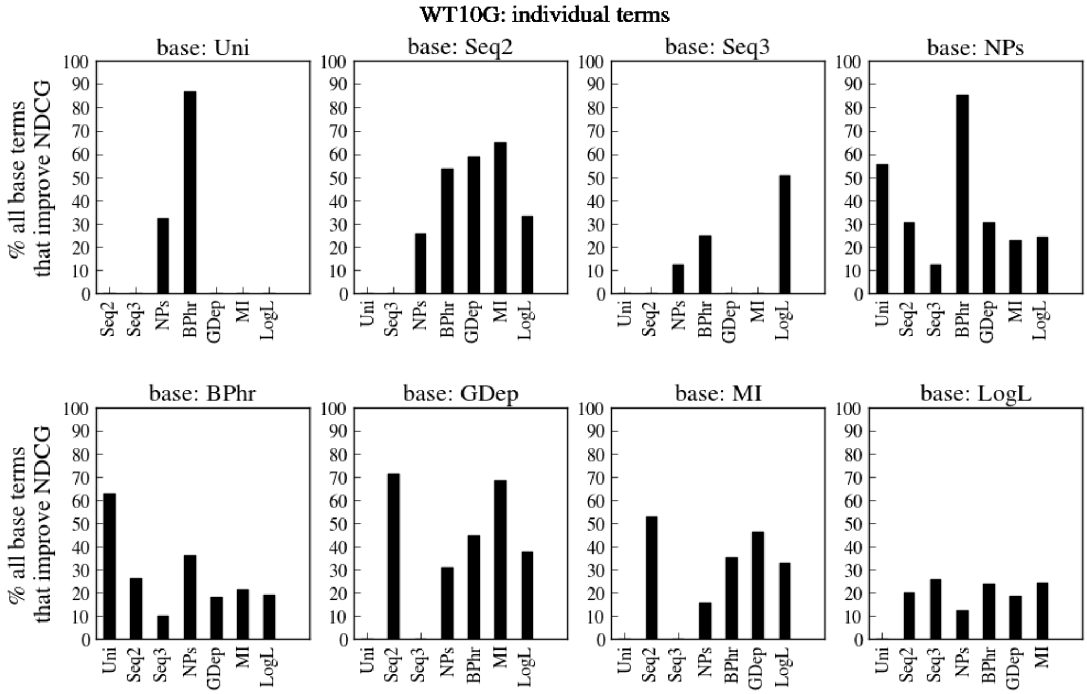



Figure E.7: Method overlap with individual terms for WT10G.

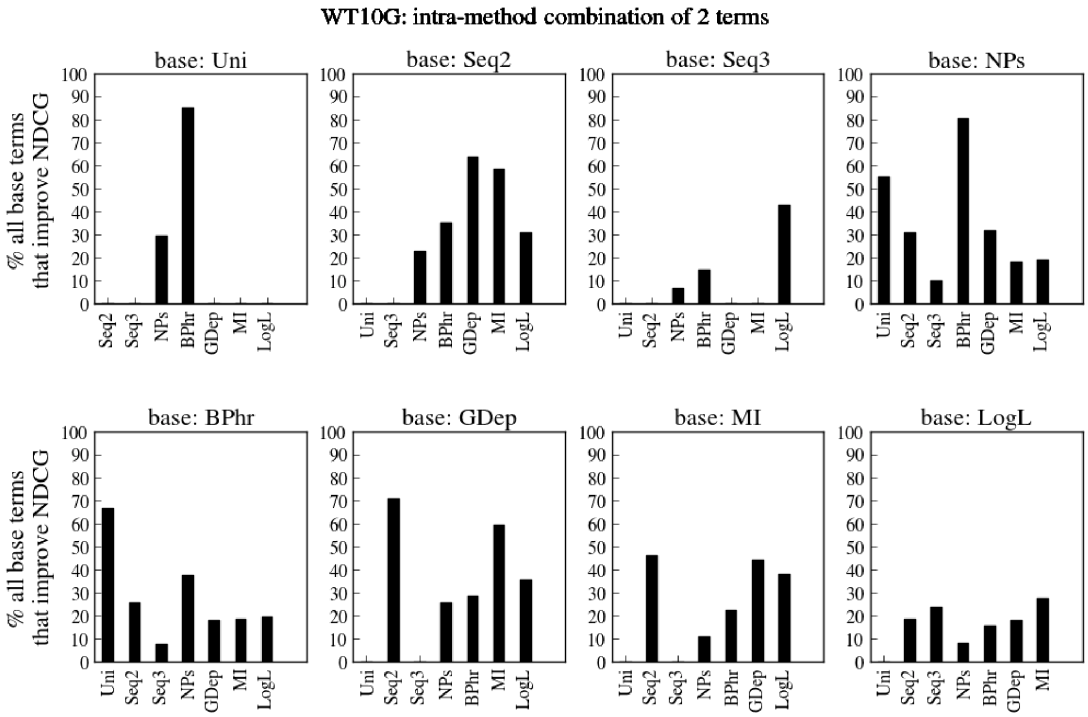



Figure E.8: Method overlap with combinations of two base terms for WT10G.

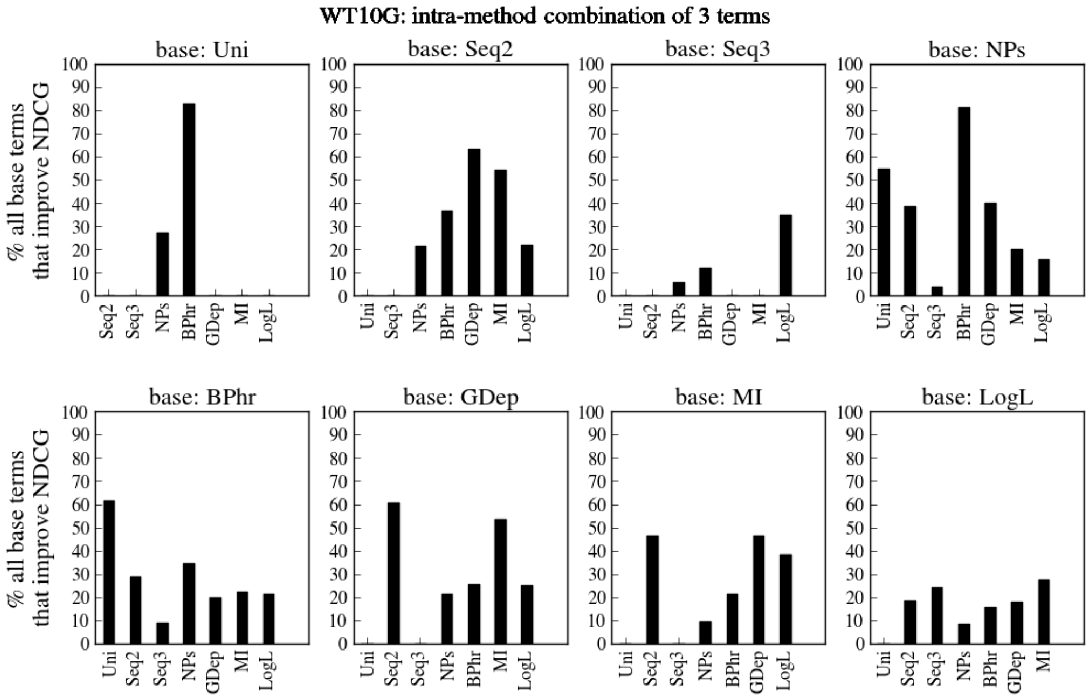



Figure E.9: Method overlap with combinations of three base terms for WT10G.

# Appendix F

# Classification Rules for Grammaticality

| Rule type | Relaxed Constraint | Strict Constraint |
|---|---|---|
| Orphaned word | The query remnant contains a token that has no dependents, its governor is ellided and one of the following is true:<br>• The token is a subject (actor);<br>• The token has no other dependents;<br>• The token is a determiner, conjunctive, possessive or punctuation mark and the immediately following token in the surface order of the coordinated query is not ellided.<br>Excludes '*documents that mercy*', '*damage loss of life*'; permits '*documents that discuss mercy*', '*damage and loss of life*'. | The query remnant contains a token that has no dependents, its governor is ellided, and the remnant text includes either:<br>• A determiner that is not followed by either an adjective or a noun;<br>• a wh-determiner (e.g. '*which*', '*what*) that is immediately followed by punctuation.<br>Excludes '*prevent the of*', '*what , the study of*'; permits '*prevent the demise of*' '*what is happening in genetics , the study of*'. |
| Incomplete quotation | The query remnant contains only either the opening or the closing mark of a pair of quotations. | |
| Invalid end | The last word in the query remnant is a determiner other than '*any*', the conjunctive '*and*' or a punctuation mark not on the whitelist ?!.”'). | |

| | | |
|---|---|---|
| Illegal sequence (2) | The query remnant contains one of the following sequences of two tokens:<br>• A noun immediately followed by an adjective, unless the adjective is immediately followed by a string of adjectives and nouns in which adjectives are optional but at least one noun is required;<br>• A wh-determiner immediately followed by punctuation;<br>• A conjunctive immediately followed by punctuation that is not in the whitelist ,:;'$#<>"<br>• A determiner that is not immediately followed by an adjective, noun, adverb, or open quotation mark;<br>• An opening quotation mark or bracket immediately followed by a closing quotation mark or bracket of the same type;<br>• A punctuation mark immediately followed by another punctuation mark (commas and quotation marks do not cooccur in TREC queries). | The query remnant contains one of the following sequences of two tokens:<br>• An adjective immediately followed by a determiner;<br>• A wh-determiner immediately followed by a determiner or punctuation; |
| Illegal sequence (3+) | The query remnant contains a sequence of part of speech tags that are unacceptable variants of the following, where $*$ indicates one or more words of any part of speech:<br>• {determiner $*$ punctuation/verb $*$ noun}<br>Excludes '*the , the area*', '*the current is the prognosis*'; permits '*the area where Burma , Thailand*', '*the current ineffectiveness and is the prognosis*'. | The query remnant contains a sequence of part of speech tags that are unacceptable variants of the following, where $*$ indicates one or more words of any part of speech:<br>• {determiner $*$ verb $*$ noun}<br>• {wh-determiner, $*$ to $*$ noun}<br>Excludes '*the current is the prognosis*', '*what to the demise*'; permits '*what effort to prevent the demise*'. |

Table F.1: Hand-coded rules to classify sentences as grammatical or ungrammatical, identified as strict or relaxed according to the observed possibility of sentence misclassification.

# Appendix G

# Comparator for Discrimination

NDCG was selected as the IR metric for term comparison on the basis of logistic regressions. This strategy avoided arbitrary selection of a evaluation metric from a set of acceptable alternatives by assuming that terms nominated by users are associated with terms that improve IR effectiveness. The IR metric that most accurately predicted user-nominated terms was selected for future experiments.

Scores for one of several IR metrics were predictors, and user-nominated terms were targets. Association between each IR metric and user-nominated terms was calculated using logistic regression (binary target variable: either a term is included on the list of user nominated terms or it is not). The strength of association was quantified by Nagelkerke's *pseudo* $R^2$ (Nagelkerke, 1991). This metric is analogous to the coefficient of determination $R^2$ typically used to quantify association for linear regression.[1] Pseudo $R^2$ is a similar metric for logistic regression but it cannot be interpreted independently or compared across different target data. Nevertheless, a higher value reliably indicates better predictions for models that predict the same target data. Nagelkerke's pseudo $R^2$ is defined as:

$$pseudo\ R^2 = \frac{1 - (\frac{L_M}{L_0})^{\frac{2}{n}}}{1 - L_0^{\frac{2}{n}}}$$

where $L_0$ is the value of the likelihood function for a model with no predictors, $L_M$ is the likelihood for the model being estimated, and $n$ is the sample size. Likelihood is computed using the $\chi^2$ test. Note that a measure of correlation can be used instead of pseudo $R^2$ to determine a degree of association (Lapata et al., 1999) but it is not clear what a correlation score means with respect to any interpretable criteria. $R^2$ measures

[1] $R^2$ is defined as the squared correlation between predictions and targets.

| Collection | % change IR metric → user judgment: Pseudo $R^2$ | | | | | |
|---|---|---|---|---|---|---|
| | **MAP** | **P@10** | **P@100** | **R-Prec** | **NDCG** | **NDCG15** |
| Robust04 | **0.095** | 0.031 | 0.066 | 0.060 | 0.084 | 0.035 |
| GOV2 | 0.035 | 0.021 | 0.019 | 0.014 | **0.057** | 0.023 |
| WT10G | 0.046 | 0.045 | 0.091 | 0.077 | **0.111** | 0.041 |

Table G.1: NDCG is the IR metric most strongly associated with user-nominated terms.

represent the proportion of total variance explained by a prediction model.

For logistic regression, if the baseline query score was greater than 0 the predictor value associated with a term was the percent change in NDCG when the term was added to a corresponding baseline query. Otherwise the value was 100. The target value for the same term, if it appeared in the list of user-nominated terms, was 1. Otherwise the value was 0. Prediction and target variables were assigned to the same set of terms: all unigrams, sequential bigrams and trigrams, noun phrases, bounded phrases, governor-dependent pairs, and terms identified by mutual information and the log likelihood ratio, as presented in Section 4.5, plus all terms on the final list of user-nominated terms (if not included already). Regressions were performed separately for Robust04, GOV2 and WT10G, and the query formulation is reported in Section 5.1.

The regression results are shown in Table G.1. According to Nagelkerke's pseudo $R^2$, NDCG is most strongly associated with user-nominated terms. MAP performs slightly better than NDCG for Robust04, but is substantially worse for the other two collections. NDCG is assumed to be the most appropriate IR metric for analysis.

# Bibliography


Abney, S. (1995). Dependency grammars and context-free grammars. Manuscript. University of Tübingen. Presented at Meeting of Linguistic Society of America, 5-8 January, New Orleans, LA.

Allan, J. (1995). Relevance feedback with too much data. In *Proceedings of the 18th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '95, pages 337–343, New York, NY, USA. ACM.

Allan, J., Ballesteros, L., Callan, J. P., Croft, W. B., and Lu, Z. (1995). Recent experiments with INQUERY. In *Proceedings of the Fourth Text REtrieval Conference (TREC-4)*, pages 49–63.

Allan, J., Connell, M. E., Croft, W. B., Feng, F.-F., Fisher, D., and Li, X. (2000). INQUERY and TREC-9. In *Proceedings of the Ninth Text REtrieval Conference (TREC-9)*, pages 551–562.

Allan, J., Dalton, J., Foley, J., Manmatha, R., Murthy, V., and Wemhoener, D. (2013). Short text queries for video retrieval: Multimedia event detection at trecvid 2013. In *Proceedings of TRECVID*.

Allan, J. and Raghavan, H. (2002). Using part-of-speech patterns to reduce query ambiguity. In *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '02, pages 307–314, New York, NY, USA. ACM.

Bach, E. (1976). An extension of classical transformational grammar. In *Proceedings of the 1976 Conference on Problems of Linguistic Metatheory*, pages 183–224.

Bach, E. (1979). Control in Montague Grammar. *Linguistics Inquiry*, 10:515–531.

Bai, J., Song, D., Bruza, P., Nie, J.-Y., and Cao, G. (2005). Query expansion using term relationships in language models for information retrieval. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, CIKM '05, pages 688–695.

Balasubramanian, N. and Allan, J. (2009). Syntactic query models for restatement retrieval. In *Proceedings of the 16th International Symposium on String Processing and Information Retrieval*, SPIRE '09, pages 143–155.

Balasubramanian, N., Kumaran, G., and Carvalho, V. R. (2010). Exploring reductions for long web queries. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 571–578.

Barker, K. and Cornacchia, N. (2000). Using noun phrase heads to extract document keyphrases. In *Proceedings of the 13th Biennial Conference of the Canadian Society on Computational Studies of Intelligence: Advances in Artificial Intelligence*, AI '00, pages 40–52.

Baroni, M., Bernardini, S., Ferraresi, A., and Zanchetta, E. (2009). The WaCky wide web: a collection of very large linguistically processed web-crawled corpora. *Language Resources and Evaluation*, 43(3):209–226.

Bauer, L. (2007). *The linguistics student's handbook*. Edinburgh University Press.

Baxendale, P. B. (1958). Machine-made index for technical literature: An experiment. *IBM Journal of Research and Development*, 2(4):354–361.

Belsey, D., Kuh, E., and Welsch, R. (1980). *Regression Diagnostics: Identifying Influential Data and Sources of Collinearity*. New York : John Wiley & Sons.

Bendersky, M. and Croft, W. B. (2008). Discovering key concepts in verbose queries. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 491–498.

Bendersky, M. and Croft, W. B. (2009). Analysis of long queries in a large scale search log. In *Proceedings of the 2009 workshop on Web Search Click Data*, WSCD '09, pages 8–14.

Bendersky, M. and Croft, W. B. (2012). Modeling higher-order term dependencies in information retrieval using query hypergraphs. In *Proceedings of the 35th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 941–950.

Bendersky, M., Metzler, D., and Croft, W. B. (2010). Learning concept importance using a weighted dependence model. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 31–40.

Bendersky, M., Metzler, D., and Croft, W. B. (2011). Parameterized concept weighting in verbose queries. In *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, SIGIR '11, pages 605–614.

Benson, M. (1989). The structure of the collocational dictionary. *International Journal of Lexicography*, 2(1):1–14.

Bergsma, S. and Wang, Q. I. (2007). Learning noun phrase query segmentation. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 819–826.

Bilotti, M. W., Elsas, J. L., Carbonell, J., and Nyberg, E. (2010). Rank learning for factoid question answering with linguistic and semantic constraints. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 459–468.

Blackburn, P. and Bos, J. (2003). Computational semantics. *Theoria*, 18(1):27–45.

Bloomfield, L. (1933). *Language*. Holt, Rinehart and Winston Inc.

Bordag, S. (2007). *Elements of Knowledge-free and Unsupervised Lexical Acquisition*. PhD thesis, University of Leipzig.

Bourigault, D. (1992). Surface grammatical analysis for the extraction of terminological noun phrases. In *Proceedings of the 14th conference on Computational linguistics - Volume 3*, COLING '92, pages 977–981.

Brants, T. (2004). Natural language processing in information retrieval. In *Proceedings of the 14th Meeting of Computational Linguistics in the Netherlands*, CLIN 2003, pages 1–13.

Bruza, P. D. and Song, D. (2002). Inferring query models by computing information flow. In *Proceedings of the 11th ACM international conference on Information and knowledge management*, CIKM '02, pages 260–269.

Buyko, E. and Hahn, U. (2010). Evaluating the impact of alternative dependency graph encodings on solving event extraction tasks. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 982–992.

Cai, D. and van Rijsbergen, C. (2009). Learning semantic relatedness from term discrimination information. *Expert Systems with Applications*, 36(2, Part 1):1860 – 1875.

Cai, K., Bu, J., Chen, C., and Liu, K. (2007a). Exploration of term dependence in sentence retrieval. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, ACL '07, pages 97–100.

Cai, K., Bu, J., Chen, C., and Qiu, G. (2007b). A novel dependency language model for information retrieval. *Journal of Zhejiang University SCIENCE A*, 8(6):871–882.

Cai, K., Chen, C., Bu, J., Qiu, G., and Huang, P. (2007c). A multi-dependency language modeling approach to information retrieval. In *Proceedings of the 2007 international conference on Emerging technologies in knowledge discovery and data mining*, PAKDD'07, pages 484–491.

Cai, K., Chen, C., Liu, K., Bu, J., and Huang, P. (2007d). MRF based approach for sentence retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 795–796.

Cao, G., Nie, J.-Y., and Bai, J. (2005). Integrating word relationships into language models. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 298–305.

Carletta, J. (1996). Assessing agreement on classification tasks: The kappa statistic. *Computational Linguistics*, 22(2):249–254.

Cer, D. C., de Marneffe, M.-C., Jurafsky, D., and Manning, C. (2010). Parsing to Stanford dependencies: Trade-offs between speed and accuracy. In *Proceedings of the seventh international conference on Language Resources and Evaluation (LREC'10)*, LREC '10, pages 1628–1632.

Chafe, W. L. (1968). Review of: Outline of stratificational grammar. *Language*, 44(3):593–603.

Chen, Z. and Fu, B. (2005). A quadratic lower bound for rocchio's similarity-based relevance feedback algorithm. In Wang, L., editor, *Computing and Combinatorics*, volume 3595 of *Lecture Notes in Computer Science*, pages 955–964. Springer Berlin Heidelberg.

Chomsky, N. (1957). *Syntactic Structures*. Mouton, The Hague/Paris.

Chomsky, N. (1965). *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA.

Chomsky, N. (1967). Recent contributions to the theory of innate ideas. *Synthese*, 17(1):2–11.

Chomsky, N. (1970). Remarks on nominalization. In Jacobs, R. and Rosenbaum, P., editors, *Readings in English Transformational Grammar*, pages 184–221. Waltham: Ginn.

Chomsky, N. (1981). *Lectures on Government and Binding: The Pisa Lectures*. Foris Publications.

Chomsky, N. (1995). *The minimalist program*. MIT Press, Cambridge, MA.

Chomsky, N. (2006). *Language and Mind*. Cambridge University Press.

Chrupala, G., Dinu, G., and Roth, B. (2010). Enriched syntax-based meaning representation for answer extraction. In *Proceedings of Query Representation and Understanding, workshop of the 33rd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 13–16.

Chu-Carroll, J., Prager, J., Czuba, K., Ferrucci, D., and Duboue, P. (2006). Semantic search via XML fragments: A high-precision approach to IR. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '06, pages 445–452.

Church, K., Gale, W., Hanks, P., and Hindle, D. (1991). Using statistics in lexical analysis. In *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*, pages 115–164. Erlbaum.

Church, K. W. (2000). Empirical estimates of adaptation: The chance of two noriegas is closer to $P/2$ than $P^2$. In *Proceedings of the 18th Conference on Computational Linguistics - Volume 1*, COLING '00, pages 180–186.

Church, K. W. (2008). Approximate lexicography and web search. *International Journal of Lexicography*, 21(3):325–336.

Church, K. W. and Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Clarke, C. L., Craswell, N., Soboroff, I., and Voorhees, E. M. (2011). Overview of the TREC 2011 web track. In *Proceedings of The Twentieth Text REtrieval Conference*, TREC 2011, pages 1–9.

Collins, M. (1999). *Head-Driven Statistical Models for Natural Language Parsing*. PhD thesis, University of Pennsylvania.

Collins-Thompson, K. and Callan, J. (2005). Query expansion using random walk models. In *Proceedings of the 14th ACM international conference on Information and knowledge management*, CIKM '05, pages 704–711.

Committee, A. L. P. A. (1966). Language and machines: Computers in translation and linguistics. Technical report, National Academy of Sciences, National Research Council.

Cooper, W. S. (1991). Some inconsistencies and misnomers in probabilistic information retrieval. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '91, pages 57–61.

Copestake, A., Lascarides, A., and Flickinger, D. (2001). An algebra for semantic construction in constraint-based grammars. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, pages 140–147.

Cowie, A. (2005). Lexicology. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Crestani, F. (1997). Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6):453–482.

Croft, W. B. (1986). Boolean queries and term dependencies in probabilistic retrieval models. *Journal of the American Society for Information Science*, 37(2):71–77.

Croft, W. B. and Harper, D. (1979). Using probabilistic models of document retrieval without relevance information. *Journal of Documentation*, 35(4):285 – 295.

Croft, W. B., Metzler, D., and Strohman, T. (2010). *Search Engines: Information Retrieval in Practice*. Addison Wesley.

Croft, W. B., Turtle, H. R., and Lewis, D. D. (1991). The use of phrases and structured queries in information retrieval. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '91, pages 32–45.

Cruse, D. (1986). *Lexical Semantics*. Cambridge University Press.

Cui, H., Sun, R., Li, K., Kan, M.-Y., and Chua, T.-S. (2005). Question answering passage retrieval using dependency relations. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 400–407.

Cummins, R. and O'Riordan, C. (2009). Learning in a pairwise term-term proximity framework for information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 251–258.

Dang, E. K. F., Luk, R. W. P., Allan, J., Ho, K. S., Chan, S. C. F., Chung, K. F. L., and Lee, D. L. (2010). A new context-dependent term weight computed by boost and discount using relevance information. *Journal of the American Society for Information Science and Technology*, 61(12):2514–2530.

Dang, H. T., Kelly, D., and Lin, J. (2007). Overview of the trec 2007 question answering track. In *Proceedings of the Sixteenth Text REtrieval Conference (TREC-16)*, pages 105–122.

de Marneffe, M.-C., MacCartney, B., and Manning, C. D. (2006). Generating typed dependency parses from phrase structure parses. In *Proceedings of the fifth international conference on Language Resources and Evaluation*, LREC '06, pages 449–454.

Dillon, M. and Gray, A. S. (1983). FASIT: A fully automatic syntactically based indexing system. *Journal of the American Society for Information Science*, 34(2):99–108.

Dori-Hacohen, S. and Allan, J. (2013). Detecting controversy on the web. In *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, CIKM '13, pages 1845–1848, New York, NY, USA. ACM.

Dryer, M. S. (1997). *Essays on Language Function and Language Type: Dedicated to T. Givon*, chapter Are Grammatical Relations Universal?, pages 115 – 143. John Benjamins, Amsterdam.

Dunning, T. (1993). Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Echihabi, A. and Marcu, D. (2003). A noisy-channel approach to question answering. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - Volume 1*, ACL '03, pages 16–23.

Efron, B. (1983). Estimating the error rate of a prediction rule: Improvement on cross-validation. *Journal of the American Statistical Association*, 78(382):316–331.

Evert, S. (2005). *The Statistics of Word Cooccurrences, Word Pairs and Collocations*. PhD thesis, University of Stuttgart.

Evert, S. (2008). *Corpus Linguistics. An International Handbook*, chapter Corpora and collocations. Mouton de Gruyter, Berlin.

Evert, S. and Krenn, B. (2001). Methods for the qualitative evaluation of lexical association measures. In *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, ACL '01, pages 188–195.

Fagan, J. L. (1987). Experiments in automatic phrase indexing for document retrieval: A comparison of syntactic and non-syntactic methods. Technical report, Cornell University.

Fang, H., Tao, T., and Zhai, C. (2004). A formal study of information retrieval heuristics. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 49–56.

Ferrer i Cancho, R. (2005). The structure of syntactic dependency networks: insights from recent advances in network theory. In Altmann, G., Levickij, V., and Perebyinis, V., editors, *The Problems of Quantitative Linguistics*, pages 60–75. Chernivtsi: Ruta.

Ferrer i Cancho, R. and Solé, R. (2001). The small-world of human language. In *Proceedings of the Royal Society of London B*, volume 268, pages 2261–2265.

Ferrer i Cancho, R., Solé, R. V., and Köhler, R. (2004). Patterns in syntactic dependency networks. *Physical Review E*, 69(5):051915–051923.

Fillmore, C. J., Kay, P., and O'Connor, M. C. (1988). Regularity and idiomaticity in grammatical constructions. *Language*, 64(3):501–538.

Firth, J. R. (1935). The technique of semantics. *Transactions of the Philological Society*, pages 36–72.

Firth, J. R. (1957). Modes of meaning. In *Papers in Linguistics, 1934-51*, pages 190–215. Oxford University Press.

Fleiss, J. (1981). *Statistical methods for rates and proportions, 2nd ed.* New York: John Wiley.

Francis, G. (1993). *Text and technology: In honour of John Sinclair*, pages 137–156. John Benjamins.

Frantzi, K. T. (1997). Incorporating context information for the extraction of terms. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, pages 501–503.

Fraser, N. M., Corbett, G. G., and McGlashan, S., editors (1993). *Heads in Grammatical Theory*, chapter Introduction. Cambridge University Press.

Freund, Y. and Schapire, R. E. (1996). Experiments with a new boosting algorithm. In *Machine Learning, Proceedings of the Thirteenth International Conference*, ICML '96, pages 148–156.

Frohman, B. (1990). Rules of indexing: A critique of mentalism in information retrieval theory. *Journal of Documentation*, 46(2):81–101.

Fujii, A., Iwayama, M., and Kando, N. (2007). Introduction to the special issue on patent processing. *Information Processing & Management*, 43(5):1149–1153.

Gaifman, H. (1965). Dependency systems and phrase-structure systems. *Information and Control*, 8:304–337.

Gao, J., Nie, J.-Y., Wu, G., and Cao, G. (2004). Dependence language model for information retrieval. In *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '04, pages 170–177.

Gao, J., Qi, H., Xia, X., and Nie, J.-Y. (2005). Linear discriminant model for information retrieval. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 290–297.

Gellner, E. (1998). *Language and Solitude: Wittgenstein, Malinowski and the Habsburg Dilemma*. Cambridge University Press.

Giegerich, H. J. (2005). Associative adjectives and the lexicon-syntax interface. *Journal of Linguistics*, 41:571–591.

Giegerich, H. J. (2006). *Englisch in Zeit und Raum - English in Time and Space: Forschungsbericht für Klaus Faiss*, chapter Attribution in English and the distinction between phrases and compounds. Wissenschaftlicher Verlag Trier.

Gledhill, C. (2000). Collocations in science writing. In *Language in Performance Series*, volume 22, pages 7–20. Gunter Narr Verlag, Tübingen.

Goldsmith, J. (2005). Review article: The legacy of Zellig Harris, edited by Bruce Nevin. *Language*, 81(3):719–736.

Graffi, G. (2005). 20th century linguistics: Overview of trends. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Grefenstette, G. (1992). Use of syntactic context to produce term association lists for text retrieval. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '92, pages 89–97.

Grefenstette, G. (1998). The future of linguistics and lexicographers: Will there be lexicographers in the year 3000. In *Proceedings of the eighth EURALEX congress*, EURALEX '98, pages 25–41.

Grice, P. (1989). *Studies in the Way of Words*. Harvard University Press, Cambridge, MA.

Guo, J., Xu, G., Cheng, X., and Li, H. (2009). Named entity recognition in query. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 267–274.

Haegeman, L. (2005). X-bar theory. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Hagen, M., Potthast, M., Stein, B., and Bräutigam, C. (2011). Query segmentation revisited. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 97–106.

Hale, K. (1983). Warlpiri and the grammar of non-configurational languages. *Natural Language and Linguistic Theory*, 1:5–47.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., and Witten, I. H. (2009). The WEKA data mining software: An update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18.

Halliday, M. and Fawcett, R., editors (1987). *New Developments in Systemic Linguistics*, chapter Introduction: The problem of how to make progress in linguistics - and keep your friends. Pinter, London.

Halliday, M. A. K. (1966). Lexis as a linguistic level. In Bazell, C., Catford, J. C., Halliday, M. A. K., and Robins, R., editors, *In Memory of J.R. Firth*, pages 148–62. Longman.

Halpin, H. (2009). *Sense and Reference on the Web*. PhD thesis, University of Edinburgh.

Harper, D. and van Rijsbergen, C. J. (1978). An evaluation of feedback in document retrieval using co-occurrence data. *Journal of Documentation*, 34(3):189–216.

Harris, Z. S. (1968). *Mathematical Structures of Language*. Interscience Publishers.

Hasan, K. S. and Ng, V. (2010). Conundrums in unsupervised keyphrase extraction: making sense of the state-of-the-art. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, COLING '10, pages 365–373.

Haspelmath, M. (2007). Pre-established categories don't exist – consequences for language description and typology. *Linguistic Typology*, 11:119–132.

Hays, D. G. (1964). Dependency theory: A formalism and some observations. Technical report, The RAND Corporation.

Heilman, M. and Smith, N. A. (2010). Tree edit models for recognizing textual entailments, paraphrases, and answers to questions. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 1011–1019.

Hockett, C. (1968). *The State of the Art*. The Hague: Mouton.

Hoenkamp, E., Bruza, P., Song, D., and Huang, Q. (2009). An effective approach to verbose queries using a limited dependencies language model. In *Proceedings of the 2nd International Conference on Theory of Information Retrieval: Advances in Information Retrieval Theory*, ICTIR '09, pages 116–127.

Honeybone, P. (2005). *Key Thinkers in Linguistics and the Philosophy of Language*, chapter J.R. Firth., pages 80–86. Edinburgh University Press, Edinburgh.

Howarth, P. (1996). *Phraseology in English Academic Writing*. Max Niemeyer, Tübingen.

Howarth, P. (1998). Phraseology and second language proficiency. *Applied Linguistics*, 19(1):24–44.

Huang, Y., Sun, L., and Nie, J.-Y. (2010). Query model refinement using word graphs. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1453–1456.

Hudson, R. (2012). An encyclopedia of word grammar and english grammar. Available at `http://tinyurl.com/wg-encyc`, Date accessed 12 Feb 2014.

Hudson, R. A. (1987). Zwicky on heads. *Journal of Linguistics*, 23(1):109–132.

Hui, B. (1988). Applying NLP to IR: Why and how. Technical report, Department of Computer Science, University of Waterloo.

Hunston, S. and Francis, G. (2000). *Pattern Grammar: A Corpus-Driven Approach to the Lexical Grammar of English*. John Benjamins Publishing Company.

Huston, S. (2013). *Indexing Proximity-based Dependencies for Information Retrieval"*. PhD thesis, University of Massachusetts, Amherst.

Huston, S. and Croft, W. B. (2010). Evaluating verbose query processing techniques. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 291–298.

Iwayama, M., Fujii, A., Kando, N., and Takano, A. (2000). NTCIR Patent: A test collection for patent retrieval / classification. In *Proceedings of Patent Retrieval, workshop of the 23rd international ACM SIGIR conference on Research and development in information retrieval*.

Jackendoff, R. (1972). *Semantic Interpretation in Generative Grammar*. MIT Press, Cambridge, MA.

Jackendoff, R. (1997). *The architecture of the language faculty*. MIT Press, Cambridge, MA.

Jacobson, P. (2005). Constituent structure. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Johansson, R. and Nugues, P. (2008). Dependency-based syntactic–semantic analysis with PropBank and NomBank. In *Proceedings of the 12th Conference on Computational Natural Language Learning*, CoNNL 2008, pages 183–187.

Jones, S. and Paynter, G. W. (2001). Human evaluation of kea, an automatic keyphrasing system. Technical report, University of Waikato.

Jones, S. and Paynter, G. W. (2003). An evaluation of document keyphrase sets. *Journal of Digital Information*, 4(1).

Justeson, J. S. and Katz, S. M. (1995). Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural Language Engineering*, 1(01):9–27.

Karlsson, F. (1990). Constraint grammar as a framework for parsing running text. In Karlgren, H., editor, *Papers presented to the 13th International Conference on Computational Linguistics*, volume 3, pages 168–173.

Kilpert, D. (2003). Getting the full picture: a reflection on the work of M.A.K. Halliday. *Language Sciences*, 25:159–209.

Kjellmer, G. (1984). *Corpus Linguistics*, chapter Some thoughts on collocational distinctiveness. Rodopi.

Klein, D. and Manning, C. D. (2003a). Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics - volume 1*, ACL '03, pages 423–430.

Klein, D. and Manning, C. D. (2003b). Fast exact inference with a factored model for natural language parsing. In Becker, S., Thrun, S., and Obermayer, K., editors, *Advances in Neural Information Processing Systems 15*, NIPS 2002, pages 3–10.

Klein, E. and Rovatsos, M. (2011). Temporal vagueness, coordination and communication. In Nouwen, R., Rooij, R., Sauerland, U., and Schmitz, H.-C., editors, *Vagueness in Communication*, volume 6517 of *Lecture Notes in Computer Science*, pages 108–126. Springer Berlin Heidelberg.

Klein, E. and Sag, I. A. (1985). Type-driven translation. *Linguistics and Philosophy*, 8(2):163–201.

Koenig, J.-P. (2005). Syntax-semantics interface. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Kolomiyets, O. and Moens, M.-F. (2011). A survey on question answering technology from an information retrieval perspective. *Information Sciences*, 181(24):5412 – 5434.

Kong, W. and Allan, J. (2013). Extracting query facets from search results. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '13, pages 93–102, New York, NY, USA. ACM.

Krishnamurthy, R. (2005). Collocations. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Krovetz, R. (1997). Homonymy and polysemy in information retrieval. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, ACL '98, pages 72–79.

Krovetz, R. and Croft, W. B. (1992). Lexical ambiguity and information retrieval. *ACM Transactions on Information Systems (TOIS)*, 10(2):115–141.

Krovetz, R. J. (1995). *Word-sense disambiguation for large text databases*. PhD thesis, University of Massachusetts, Amherst.

Kruijff, G.-J. M. (2005). Dependency grammar. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Kumaran, G. and Allan, J. (2006). Simple questions to improve pseudo-relevance feedback results. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '06, pages 661–662, New York, NY, USA. ACM.

Kumaran, G. and Allan, J. (2007). A case for shorter queries, and helping users create them. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 220–227.

Kumaran, G. and Allan, J. (2008). Effective and efficient user interaction for long queries. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 11–18.

Kumaran, G. and Carvalho, V. R. (2009). Reducing long queries using query quality predictors. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 564–571.

Kuno, S. and Oettinger, A. (1962). Multiple-path syntactic analyzer. In *Information Processing 1962, Proceedings of IFIP Congress 62*, pages 306–312.

Kunze, J. (1975). *Abhängigkeitsgrammatik*. Akademie Verlag, Berlin.

Lafferty, J. and Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 111–119.

Lakoff, G. (1987). *Women, Fire, and Dangerous Things: What Categories Reveal About the Mind*. Chicago: CSLI.

Lapata, M., McDonald, S., and Keller, F. (1999). Determinants of adjective-noun plausibility. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, EACL '99, pages 30–36.

LaPolla, R. J. (2006). The how and why of syntactic relations. Paper presented at the Annual Conference of the Australian Linguistics Society.

Lavrenko, V. (2004). *A Generative Theory of Relevance*. PhD thesis, University of Massachusetts, Amherst.

Lavrenko, V. and Croft, W. B. (2001). Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 120–127.

Lease, M. (2007). Natural language processing for information retrieval: The time is ripe (again). In *Proceedings of the ACM first Ph.D. workshop in CIKM (conference on Information and knowledge management)*, PIKM '07, pages 1–8.

Lease, M. (2009). An improved markov random field model for supporting verbose queries. In *Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '09, pages 476–483.

Lease, M. (2010). *Beyond keywords: finding information more accurately and easily using natural language*. PhD thesis, Brown University.

Lease, M., Allan, J., and Croft, W. B. (2009). Regression rank: Learning to meet the opportunity of descriptive queries. In *Proceedings of the 31st European Conference on IR Research on Advances in Information Retrieval*, ECIR '09, pages 90–101.

Lee, C., Lee, G. G., and Jang, M.-G. (2006). Dependency structure language model for information retrieval. *ETRI journal (Electronics and Telecommunications Research Institute)*, 28(3):337–346.

Levi, J. N. (1978). *The syntax and semantics of complex nominals*. Academic Press, San Francisco, CA & London.

Levy, S. (2011). *In the Plex*. Simon and Schuster.

Lewis, D. D. (1992). An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of the 15th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '92, pages 37–50.

Lewis, D. D. and Croft, W. B. (1990). Term clustering of syntactic phrases. In *Proceedings of the 13th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '90, pages 385–404.

Lewis, D. D. and Jones, K. S. (1996). Natural language processing for information retrieval. *Communications of the ACM*, 39(1):92–101.

Lin, D. and Pantel, P. (2001). DIRT - discovery of inference rules from text. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, pages 323–328.

Lindeman, R., Merenda, P., and Gold, R. (1980). *Introduction to Bivariate and Multivariate Analysis*. Scott, Foresman, Glenview IL.

Lioma, C. and Ounis, I. (2008). A syntactically-based query reformulation technique for information retrieval. *Information Processing and Management*, 44(1):143–162.

Liu, H. (2004). MontyLingua: An end-to-end natural language processor with common sense. Available at `web.media.mit.edu/~hugo/montylingua`, Date accessed 13 Feb 2014.

Liu, T.-Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, 3(3):225–331.

Losee Jr., R. M. (1994). Term dependence: Truncating the Bahadur Lazarsfeld expansion. *Information Processing & Management*, 30(2):293–303.

Lund, K. and Burgess, C. (1996). Producing high dimensional semantic spaces from lexical co-occurrence. *Behavior Research Methods, Instruments, and Computers*, 28(2):203–208.

Luo, X., Raghavan, H., Castelli, V., Maskey, S., and Florian, R. (2013). Finding what matters in questions. In *Human Language Technologies: The 2013 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, NAACL-HLT '13, pages 878–887.

Macdonald, C. and Ounis, I. (2010). Global statistics in proximity weighting models. In *Proceedings of Web N-gram workshop of the 33rd annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10.

Maisonnasse, L., Gaussier, E., and Chevallet, J.-P. (2007). Revisiting the dependence language model for information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 695–696.

Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.

Manning, C. D. (2007). Learning language from distributional evidence. Presentation at MIT World Series Workshop: Where does syntax come from? Have we all been wrong? Available at `http://video.mit.edu/watch/machine-learning-of-language-from-distributional-evidence-9291/`, Date accessed 13 Feb 2014.

Manning, C. D., Raghavan, P., and Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Marcotty, M. (1996). Obituaries - Gerard Salton. *IEEE Annals of the History of Computing*, 18(1):67–68.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

May, R. (1985). *Logical form: Its structure and derivation*. MIT Press, Cambridge, MA.

McGill, W. J. (1954). Multivariate information transmission. *Psychometrika*, 19:97–116.

McInnes, B. T. (2004). Extending the log likelihood measure to improve collocation identification. Master's thesis, University of Minnesota.

Medelyan, O. (2007). Computing lexical chains with graph clustering. In *Proceedings of the 45th Annual Meeting of the ACL: Student Research Workshop*, ACL '07, pages 85–90.

Meetham, A. (1963). Preliminary studies for machine generated index vocabularie. *Language and Speech*, 6:22–36.

Mei, Q., Zhang, D., and Zhai, C. (2008). A general optimization framework for smoothing language models on graph structures. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 611–618.

Meij, E., Bron, M., Hollink, L., Huurnink, B., and Rijke, M. (2009). Learning semantic query suggestions. In *Proceedings of the 8th International Semantic Web Conference*, ISWC '09, pages 424–440.

Mel'čuk, I. A. (1981). Meaning-text models: A recent trend in soviet linguistics. *Annual Review of Anthropology*, 10:27–62.

Mel'čuk, I. A. (1988). *Dependency Syntax: Theory and Practice*. State University of New York Press.

Mel'čuk, I. A. (1998). Collocations and lexical functions. In Cowie, A., editor, *Phraseeology, Theory, Analysis and Applications*, pages 23–53. Oxford University Press.

Mel'čuk, I. A. (2003). Levels of dependency in linguistic description: Concepts and problems. In Agel, V., Eichinger, L., Eroms, H.-W., Hellwig, P., Herringer, H. J., and Lobin, H., editors, *Dependency and Valency. An International Handbook of Contemporary Research*, volume 1, pages 188–229. Walter De Gruyter, Berlin–New York.

Metzler, D. (2007). Using gradient descent to optimize language modeling smoothing parameters. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 687–688.

Metzler, D. and Croft, B. W. (2007a). Linear feature-based models for information retrieval. *Information Retrieval*, 10(3):257–274.

Metzler, D. and Croft, W. B. (2004). Combining the language model and inference network approaches to retrieval. *Information Processing and Management: an International Journal - Special issue: Bayesian networks and information retrieval*, 40(5):735–750.

Metzler, D. and Croft, W. B. (2005). A Markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '05, pages 472–479.

Metzler, D. and Croft, W. B. (2007b). Latent concept expansion using Markov random fields. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 311–318.

Metzler, D. P. and Haas, S. W. (1989). The constituent object parser: Syntactic structure matching for information retrieval. In *Proceedings of the 12th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '89, pages 117–126.

Mihalcea, R. and Tarau, P. (2004). TextRank: Bringing order into texts. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, EMNLP 2004, pages 404–411.

Mizzaro, S. (1998). How many relevances in information retrieval? *Interacting With Computers*, 10:305–322.

Montague, R. (1974). *Formal Philosophy*. Yale University Press, New Haven, CT.

Moschitti, A. (2008). Kernel methods, syntax and semantics for relational text categorization. In *Proceeding of the 17th ACM international conference on Information and knowledge management*, CIKM '08, pages 253–262.

Na, S.-H., Kim, J., Kang, I.-S., and Lee, J.-H. (2008). Exploiting proximity feature in bigram language model for information retrieval. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pages 821–822.

Nagelkerke, N. J. D. (1991). A note on a general definition of the coefficient of determination. *Biometrika*, 78(3):691–692.

Nallapati, R. and Allan, J. (2002). Capturing term dependencies using a language model based on sentence trees. In *Proceedings of the eleventh international conference on Information and knowledge management*, CIKM '02, pages 383–390.

Nallapati, R., Allan, J., and Mahadevan, S. (2004). Extraction of key-words from news stories. IR 345, University of Massachusetts.

Needham, R. M. (1965). Applications of the theory of clumps. *Mechanical Translation and Computational Linguistics*, 8(3 and 4):113–127.

Needham, R. M. (1967). Automatic classification in linguistics. *Journal of the Royal Statistical Society*, 17(1):45–54.

Nevin, B. E. (2010). *Chomskyan (R)evolutions*, chapter Noam and Zellig, pages 103–168. John Benjamins Publishing Company.

Nivre, J. (2003). An efficient algorithm for projective dependency parsing. In *Proceedings of the 8th International Workshop on Parsing Technologies*, IWPT 2003, pages 149–160.

Nivre, J. (2005). Dependency grammar and dependency parsing. Technical report, Växjö University: School of Mathematics and Systems Engineering.

Nivre, J., Hall, J., Kübler, S., McDonald, R., Nilsson, J., Riedel, S., and Yuret, D. (2007). The CoNLL 2007 shared task on dependency parsing. In Eisner, J., editor, *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 915–932.

Norvig, P. (2011). On Chomsky and the two cultures of statistical learning. Available at `http://norvig.com/chomsky.html`, Date accessed 23 July 2012.

Nunberg, G., Wasow, T., and Sag, I. A. (1994). Idioms. *Language*, 70(3):491–538.

O'Grady, W. (1998). The syntax of idioms. *Natural Language and Linguistic Theory*, 16(2):279–312.

Oller, D. K. (2008). Noam Chomsky's role in biological theory: A mixed legacy. *Biological Theory*, 3(4):344–350.

Osborne, T. (2005). Beyond the constituent: A dependency grammar analysis of chains. *Folia Linguistica*, 39(3-4):251–297.

Osborne, T. and Groß, T. (2012). Constructions are catenae: Construction grammar meets dependency grammar. *Cognitive Linguistics*, 23(1):165–216.

Osborne, T., Putnam, M., and Groß (2012). Catenae: Introducing a novel unit of syntactic analysis. *Syntax*, 15(4):354–396.

Page, L., Brin, S., Motwani, R., and Winograd, T. (1999). The PageRank citation ranking: Bringing order to the web. Technical report, Stanford University, Stanford, CA.

Palmer, H. (1933). *Second Interim Report on English Collocations: submitted to the Tenth Annual Conference of English Teachers*. Institute for Research in English Teaching.

Park, J. H. and Croft, W. B. (2010). Query term ranking based on dependency parsing of verbose queries. In *Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '10, pages 829–830.

Park, J. H., Croft, W. B., and Smith, D. A. (2011). A quasi-synchronous dependence model for information retrieval. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 17–26.

Partee, B. H. (2001). Montague semantics. In Smelser, N. J. and Baltes, P. B., editors, *International Encyclopedia of the Social and Behavioral Sciences*. Pergamon/ Elsevier Science.

Pasca, M. A. and Harabagiu, S. M. (2001). High performance question/answering. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '01, pages 366–374.

Peat, H. J. and Willett, P. (1991). The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the American Society for Information Science*, 42(5):378–83.

Pecina, P. and Schlesinger, P. (2006). Combining association measures for collocation extraction. In *Proceedings of the COLING/ACL main conference poster sessions*, COLING-ACL '06, pages 651–658.

Pickering, M. and Barry, G. (1993). Dependency categorial grammar and coordination. *Linguistics*, 31(5):855–902.

Pollard, C. and Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar*. University of Chicago Press.

Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '98, pages 275–281.

Prager, J. (2006). Open-domain question–answering. *Foundations and Trends in Information Retrieval*, 1(2):91–231.

Prince, E. F. (1986). On the syntactic marking of presupposed open propositions. In *Proceedings of the 22nd Annual Meeting of the Chicago Linguistic Society*, pages 208–222.

Proffitt, M., editor (2013). *Oxford British and World English Dictionary*. Oxford University Press.

Punyakanok, V., Roth, D., and Yih, W. (2004). Mapping dependencies trees: An application to question answering. In *Presented at the 8th International Symposium on Artificial Intelligence and Mathematics (Special session: Intelligent Text Processing)*.

Raghavan, H. and Allan, J. (2007). An interactive algorithm for asking and incorporating feature feedback into support vector machines. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 79–86, New York, NY, USA. ACM.

Rambow, O. (2010). The simple truth about dependency and phrase structure representations: An opinion piece. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 337–340.

Rasolofo, Y. and Savoy, J. (2003). Term proximity scoring for keyword-based retrieval systems. In *Proceedings of the 25th European conference on IR research*, ECIR'03, pages 207–218.

Rehman, S. (2010). Wittgenstein's language-games, Stoppard's building-blocks and context-learning in a corpus. *Skase Journal of Literary Studies*, 2(1):67–83.

Risvik, K. M., Mikolajewski, T., and Boros, P. (2003). Query segmentation for web search. In *Proceedings of the Twelfth International World Wide Web Conference*, WWW Posters 2003.

Robertson, S. E. (1977). The probability ranking principle in IR. *Journal of Documentation*, 33:294–304.

Robertson, S. E. and Spärck Jones, K. (1988). Relevance weighting of search terms. In Willett, P., editor, *Document Retrieval Systems*, pages 143–160. Taylor Graham Publishing, London, UK.

Robertson, S. E. and Walker, S. (1994). Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '94, pages 232–241.

Robinson, J. T. (1970). Dependency structures and transformational rules. *Language*, 46(2):259–285.

Rocchio, J. (1971). Relevance feedback in information retrieval. In *The SMART Retrieval System - Experiments in Automatic Document Processing*. Prentice-Hall, Inc., NJ, USA.

Ruthven, I. and Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *Knowl. Eng. Rev.*, 18(2):95–145.

Sadock, J. (1991). *Autolexical syntax*. Chicago University Press, Chicago.

Salton, G. (1963). Associative document retrieval techniques using bibliographic information. *Journal of the ACM (JACM)*, 10(4):440–457.

Salton, G. (1964a). Automatic information processing in Western Europe. *Science*, 144(3619):626–632.

Salton, G. (1964b). Automatic phrase matching. Technical Report ISR-8, The National Science Foundation.

Salton, G. (1966). Information storage and retrieval. Scientific Report No. ISR-11 to the National Science Foundation. Technical report, Cornell University.

Salton, G. (1971). *The SMART Retrieval System - Experiments in Automatic Document Processing*. Prentice-Hall, Inc., NJ, USA.

Salton, G. (1989). *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison-Wesley Publishing Company.

Salton, G., Allan, J., and Buckley, C. (1993). Approaches to passage retrieval in full text information systems. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '93, pages 49–58.

Salton, G. and Buckley, C. (1991). Automatic text structuring and retrieval - experiments in automatic encyclopedia searching. In *Proceedings of the 14th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '91, pages 21–30.

Salton, G. and McGill, M. (1983). *Introduction to modern information retrieval*. McGraw-Hill.

Salton, G. and Smith, M. (1990). On the application of syntactic methodologies in automatic text analysis. *Information Processing and Management*, 26(1):73–92.

Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, 18(11):613–620.

Schapire, R. E., Freund, Y., Bartlett, P., and Lee, W. S. (1997). Boosting the margin: A new explanation for the effectiveness of voting methods. In *Machine Learning: Proceedings of the Fourteenth International Conference*, ICML '97, pages 322–330.

Schneider, G. (2007). *Hybrid long-distance functional dependency parsing*. PhD thesis, University of Zurich.

Schone, P. and Jurafsky, D. (2001). Is knowledge-free induction of multiword unit dictionary headwords a solved problem? In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, pages 100–108.

Schwartz, C. (1990). Automatic syntactic analysis of free text. *Journal of the American Society for Information Science*, 41(6):408–417.

Searle, J. R. (1972). Chomsky's revolution in linguistics. *New York Review of Books*, 18(12):16–24.

Sgall, P., Hajičová, E., and Panevová, J. (1986). *The Meaning of the Sentence in Its Semantic and Pragmatic Aspects*. D. Reidel Publishing Company, Dordrecht.

Shen, D., Kruijff, G.-J. M., and Klakow, D. (2005). Exploring syntactic relation patterns for question answering. In *Proceedings of the Second international joint conference on Natural Language Processing*, IJCNLP'05, pages 507–518.

Sheridan, P. and Smeaton, A. F. (1992). The application of morpho-syntactic language processing to effective phrase matching. *Information Processing & Management*, 28(3):349–369.

Shi, L. and Nie, J.-Y. (2010). Using various term dependencies according to their utilities. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1493–1496.

Sinclair, J. (1966). Beginning the study of lexis. In Bazell, C., Catford, J. C., Halliday, M. A. K., and Robins, R., editors, *In Memory of J.R. Firth*. Longman.

Sinclair, J. (1987a). *Language topics: Essays in Honour of Michael Halliday*, chapter Collocation: A progress report, pages 319–331. John Benjamins, Amsterdam.

Sinclair, J. (1991). *Corpus, Concordance, Collocation*. Oxford University Press.

Sinclair, J. M. (1970 (2004)). *English Collocation Studies: The OSTI Report*. Continuum.

Sinclair, J. M., editor (1987b). *Collins Cobuild English language dictionary*, chapter Introduction. Collins, London/Glasgow.

Singhal, A. (2001). Modern information retrieval: A brief overview. In *Bulletin IEEE Computer Society Technical Committee on Data Engineering*, volume 24, pages 35–43.

Sleator, D. D. K. and Temperley, D. (1993). Parsing English with a link grammar. In *Proceedings of the Third International Workshop on Parsing Technologies*, IWPT '93, pages 277–292.

Smadja, F. (1989). Macrocoding the lexicon with co-occurrence knowledge. In Zernik, U., editor, *Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon*, pages 165–190. Lawrence Erlbaum Associates, Hillsdale, NJ.

Smeaton, A., O'Donnell, R., and Kelledy, F. (1995). Indexing structures derived from syntax in TREC-3: System description. In *Proceedings of the Third Text REtrieval Conference (TREC-3)*, pages 55–67.

Smeaton, A. F. (1999). Using NLP or NLP resources for information retrieval tasks. In Strzalkowski, T., editor, *Natural Language Information Retrieval*, pages 99–111. Kluwer Academic Publishers, Dordrecht.

Smeaton, A. F. and van Rijsbergen, C. J. (1988). Experiments on incorporating syntactic processing of user queries into a document retrieval strategy. In *Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '88, pages 31–51.

Smith, R. C. (1999). *The Writings of Harold E. Palmer: An Overview*. Hon-no-Tomosha, Japan.

Song, F. and Croft, W. B. (1999). A general language model for information retrieval. In *Proceedings of the eighth international conference on Information and knowledge management*, CIKM '99, pages 316–321.

Song, R., Taylor, M., Wen, J.-R., Hon, H.-W., and Yu, Y. (2008a). Viewing term proximity from a different perspective. In Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., and White, R., editors, *Advances in Information Retrieval*, volume 4956 of *Lecture Notes in Computer Science*, pages 346–357. Springer Berlin / Heidelberg.

Song, Y.-I., Han, K.-S., Kim, S.-B., Park, S.-Y., and Rim, H.-C. (2008b). A novel retrieval approach reflecting variability of syntactic phrase representation. *Journal of Intelligent Information Systems*, 31(3):265–286.

Spärck Jones, K. (1965). Experiments in semantic classification. *Mechanical Translation and Computational Linguistics*, 8(3 and 4):97–112.

Spärck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28(1):11–21.

Spärck Jones, K. (1999). What is the role of NLP in text retrieval? In Strzalkowski, T., editor, *Natural Language Information Retrieval*, pages 1–24. Kluwer Academic Publishers, Dordrecht.

Spärck Jones, K. and Abbate, J. (2001). Karen Spärck Jones: An oral history conducted in 2001 by Janet Abbate. IEEE History Center, New Brunswick, NJ, USA.

Spärck Jones, K. and Tait, J. (1984). Automatic search term variant generation. *Journal of Documentation*, 40(1):50–66.

Spärck Jones, K., Walker, S., and Robertson, S. E. (2000). A probabilistic model of information retrieval: Development and comparative experiments. *Information Processing and Management*, 36(6):779–808.

Srikanth, M. and Srihari, R. (2002). Biterm language models for document retrieval. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '02, pages 425–426.

Srikanth, M. and Srihari, R. (2003a). Exploiting syntactic structure of queries in a language modeling approach to IR. In *Proceedings of the twelfth international conference on Information and knowledge management*, CIKM '03, pages 476–483.

Srikanth, M. and Srihari, R. (2003b). Incorporating query term dependencies in language models for document retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, SIGIR '03, pages 405–406.

Steedman, M. (1999). Categorial grammar. In Wilson, R. A. and Keil, F., editors, *The MIT Encyclopedia of Cognitive Sciences*, pages 101–104. MIT Press, Cambridge, MA.

Steedman, M. (2000). *The syntactic process*. MIT Press, Cambridge, MA.

Steedman, M. J. (1990). Gapping as Constituent Coordination. *Linguistics and Philosophy*, 13(2):207–263.

Stiles, H. E. (1961). The association factor in information retrieval. *Journal of the ACM (JACM)*, 8(2):271–279.

Stone, M. and Webber, B. (1998). Textual economy through close coupling of syntax and semantics. In *Proceedings of the Ninth International Workshop on Natural Language Generation*, INLG '98, pages 178–187.

Strzalkowski, T. and Carballo, J. P. (1993). Recent developments in natural language text retrieval. In Harman, D., editor, *Proceedings of the Second Text REtrieval Conference (TREC-2)*, pages 123–136.

Strzalkowski, T. and Vauthey, B. (1992). Information retrieval using robust natural language processing. In *Proceedings of the 30th annual meeting on Association for Computational Linguistics*, ACL '92, pages 104–111.

Stubbs, M. (1995). Collocations and semantic profiles: On the cause of the trouble with quantitative studies. *Functions of language*, pages 23–55.

Stubbs, M. (2009). Memorial article: John Sinclair (1933-2007). *Applied Linguistics*, 30(1):115–137.

Surdeanu, M., Ciaramita, M., and Zaragoza, H. (2008). Learning to rank answers on large online QA collections. In *Proceedings of the 46th Annual Meeting for the Association for Computational Linguistics: Human Language Technologies*, ACL-08: HLT, pages 719–727.

Surdeanu, M., Ciaramita, M., and Zaragoza, H. (2011). Learning to rank answers to non-factoid questions from web collections. *Computational Linguistics*, 37(2):351–383.

Tan, B. and Peng, F. (2008). Unsupervised query segmentation using generative language models and wikipedia. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 347–356.

Tan, P.-N., Kumar, V., and Srivastava, J. (2002). Selecting the right interestingness measure for association patterns. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 32–41.

Tao, T. and Zhai, C. (2007). An exploration of proximity measures in information retrieval. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 295–302.

Tesnière, L. (1959). *Eléments de Syntaxe Structurale*. Klincksieck, Paris.

Thompson, H. (1977). Strategy and tactics: A model for language production. In *Papers from the Thirteenth Regional Meeting, Chicago Linguistics Society*, pages 651–668.

Tsarfaty, R. (2010). *Relational-Realizational Parsing*. PhD thesis, University of Amsterdam.

Tsatsaronis, G. and Panagiotopoulou, V. (2009). A generalized vector space model for text retrieval based on semantic relatedness. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, EACL '09, pages 70–78.

Turney, P. (1999). Learning to extract keyphrases from text. Technical report, National Research Council, Institute for Information Technology.

Turney, P. D. and Pantel, P. (2010). From frequency to meaning: vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188.

Turtle, H. (1995). Text retrieval in the legal world. *Artificial Intelligence and Law*, 3:5–54.

Turtle, H. and Croft, W. B. (1991). Evaluation of an inference network-based retrieval model. *ACM Transactions on Information Systems (TOIS)*, 9(3):187–222.

Van de Cruys, T. (2011). Two multivariate generalizations of pointwise mutual information. In *Proceedings of the Workshop on Distributional Semantics and Compositionality*, DiSCo '11, pages 16–20.

van der Wouden, T. (1997). *Negative Contexts: Collocation, Polarity and Multiple Negation*. Routledge.

van Rijsbergen, C. J. (1977). A theoretical basis for the use of co-occurrence data in information retrieval. *Journal of Documentation*, 33(2):106–119.

van Rijsbergen, C. J. (1979a). *Information Retrieval*. Butterworths, London, UK.

van Rijsbergen, C. J. (1979b). Information retrieval: theory and practice. In *Proceedings of the Joint IBM/University of Newcastle upon Tyne Seminar on Data Base Systems*, pages 1–14.

van Rijsbergen, C. J. (1983). A discrimination gain hypothesis. In *Proceedings of the 6th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '83, pages 101–104.

Vechtomova, O. and Karamuftuoglu, M. (2004). Approaches to high accuracy retrieval: Phrase-based search experiments in the hard track. In *Proceedings of the Thirteenth Text REtrieval Conference*, TREC 2004.

Voorhees, E. (2001). Common evaluation measures. In *Proceedings of the Tenth Text REtrieval Conference*, TREC 2001, pages A14 – A23.

Voorhees, E. M. (2005). Overview of the TREC 2005 question answering track. In *Proceedings of the Fourteenth Text REtrieval Conference*, TREC 2005.

Wan, X. and Xiao, J. (2008). Single document keyphrase extraction using neighborhood knowledge. In *Proceedings of the 23rd national conference on Artificial intelligence - Volume 2*, AAAI'08, pages 855–860.

Wang, M., Smith, N. A., and Mitamura, T. (2007). What is the Jeopardy model? a quasi-synchronous grammar for QA. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 22–32.

Wang, Y.-C., Vandendorpe, J., and Evens, M. (1985). Relational thesauri in information retrieval. *Journal of the American Society for Information Science*, 36(1):15–27.

Williams, P. L. and Beer, R. D. (2010). Nonnegative decomposition of multivariate information. *Computing Research Repository (CoRR) - arXiv*, abs/1004.2515.

Winkler, S. (2005). Ellipsis. In Brown, K., editor, *Encyclopedia of Language and Linguistics, 2nd Edition*. Elsevier.

Wittgenstein, L. (1953). *Philosophical Investigations*. Blackwell Publishers.

Wittgenstein, L. (1958). *The Blue and Brown Books; Preliminary Studies for the 'Philosophical Investigations'*. Harper and Row.

Wong, S. K. M., Ziarko, W., and Wong, P. C. N. (1985). Generalized vector spaces model in information retrieval. In *Proceedings of the 8th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '85, pages 18–25.

Wood, M. M. (1981). *A definition of idiom*. Bloomington: Indiana University Linguistics Club.

Wu, H. C., Luk, R. W. P., Wong, K. F., and Kwok, K. L. (2008). Interpreting TF-IDF term weights as making relevance decisions. *ACM Transactions on Information Systems (TOIS)*, 26(3):13:1–13:37.

Xue, X., Huston, S., and Croft, W. B. (2010). Improving verbose queries using subset distribution. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1059–1068.

Yu, C., Buckley, C., Lam, K., and Salton, G. (1983). A generalized term dependence model in information retrieval. Technical report, Cornell University.

Zhai, C. (2008). *Statistical Language Models for Information Retrieval*. Morgan and Claypool Publishers.

Zhai, C. and Lafferty, J. (2001). Model-based feedback in the language modeling approach to information retrieval. In *Proceedings of the tenth ACM international conference on Information and knowledge management*, CIKM '01, pages 403–410.

Zhao, L. and Callan, J. (2010). Term necessity prediction. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 259–268, New York, NY, USA. ACM.

Zhao, S., Wang, H., Li, C., Liu, T., and Guan, Y. (2011). Automatically generating questions from queries for community-based question answering. In *Proceedings of the 5th International Joint Conference on Natural Language Processing*, pages 929–937.

Zhou, G., Cai, L., Zhao, J., and Liu, K. (2011). Phrase-based translation model for question retrieval in community question answer archives. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 653–662.

Zhou, Y. and Croft, W. B. (2007). Query performance prediction in web search environments. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 543–550.

Zukerman, I. and Raskutti, B. (2002). Lexical query paraphrasing for document retrieval. In *Proceedings of the 19th international conference on Computational linguistics - Volume 1*, COLING '02, pages 1–7.

Zwicky, A. and Isard, S. (1963). Some aspects of tree theory. Technical report, MITRE Corporation.

Zwicky, A. M. (1985). Heads. *Journal of Linguistics*, 21:1–29.