Automatic Lexicon Acquisition from Encyclopedia

LO, Ka Kan

A Thesis Submitted in Partial Fulfilment
of the Requirements for the Degree of
Master of Philosophy
in
Systems Engineering and Engineering Management

Thesis/Assessment Committee

Professor Yang Christopher Chuen Chi (Chair)
Professor Lam Wai (Thesis Supervisor)
Professor Cheng Chun Hung (Committee Member)
Professor Wong Man-leung (External Examiner)

Acknowledgment

Many thanks are due here. First, to my advisor, Wai Lam, who offered me a chance to work on the challenging topic in Natural Language Processing, read the numerous revisions of this thesis, and helped clarify many of the original confusions. Without your guidance in working on practical implementation of Natural Language system and the paper submissions, this thesis would not have been possible. I am grateful for your supports and confidence in letting me to work on the chosen topic freely while at the same time, setting the right path so that I can make sense of the whole work. Second, thanks to my defense committee, Professor Cheng and Professor Yang, who gave me a lot of advice in shaping the research project in the progress presentation and inspired me to thank more deeply about the research problems and evaluation methods.

Thanks to the Chinese University of Hong Kong for awarding me studentships, providing me the financial support to complete the project. Thanks to my officemates: Gatien Wong, Gabriel Fung, Tony Woo, Ken Wu, Cecia, KaKa, Cathy Wong, Mei Choi, Gao Wei, Walter To, Keith and Sancho, which provided me a stimulating environment to do the research while at the same time, a relaxing place to balance all the pressures induced during this project.

Additional thanks to the technical supports of the SEEM department who offered great help in establishing PC Cluster environment to run the huge experiments.

Finally, my deepest thanks for the love and support of my family. To my mom and dad, go read the dedication!

Abstract

Research works in natural language processing and computational linguistic focus primarily in the representation and processing issues of language data from corpora. Though these issues have been critical for whatever breakthroughs to be made, there is one and only one issue which is largely unattended for the communities, the automatic lexicon acquisition problem. Recent progresses have demonstrated that using larger corpora and more linguistically-oriented representation is the positive research direction and encouraging results start to emerge consistently in the field. However, there is one unanswered question from the current acquisition work, the semantic of the acquired languages. Most existing works focus on the acquisition of the syntax from corpora, without the semantic information. Though the automatic method generates a vision that the huge amount of unannotated language data can extend the existing NLP work to whatever domains exist, the inadequate assumption underlying the recent work hinders the further adoption of the automatic lexicon acquisition problem to the more general NLP problem such as question answering and information extraction where high-level semantic is their core components.

In this work, we describe and evaluate a novel model for integrating the syntactic and semantic information in the acquisition of lexicon. The Head-driven Phrase Structure Grammar (HPSG) is the major representation to be used in modeling the linguistic information where it contains both syntactic and semantic representations. First, we discuss how the current design of the HPSG can be potentially used in the acquisition algorithm, which has

been seldom mentioned in the fields and formalize the acquisition structure used. Second, we describe the utilization of the link structure data such as Wikipedia for the addition of semantic information to the acquisition process due to its internal linkages between concepts. Finally, we evaluate our proposed acquisition algorithm in English language data and English Resource Grammar (ERG) and show that it can outperform the baseline.

Automatic lexicon acquisition is critical in further advancement in existing natural language applications such as question answering, information extraction where high level syntax and semantic have to be integrated tightly and the semantic has to be defined and filled by efficient representations. In addition, the current work, while the primary focus is not on the linguistic representation, can shed light in the future development and extension of linguistic theory for making better uses of the semantic information from the corpora.

前言

自然語言處理和計算語言學研究工作的焦點主要表現在來自語料庫的語言結構代表方法和語言數據處理問題上. 雖然這些研究課題一直是至關重要,另一個課題－詞庫自動採集問題則是同樣重要但常被研究人員忽略 .最新進展表明,使用更大的語料庫和更複雜的語言結構代表方法去處理上述課題的研究方向是正面和研究成果是令人鼓舞的 . 從目前的研究工作,有一個無法解答的問題－語義. 現有的大部分作品集中於處理語法結構 ,但無語義信息.雖然目前詞庫自動採集問方法有一個想法 ,通過大量的語言數據可處理不同領域的語言資料, 但是忽略語義這個基本假設阻礙了進一步通過了詞庫的自動採集問題去處理更高層次的語言應用 .

在這篇論文中, 我們描述和測試一種把句法和語義結合的詞庫自動採集模型 . Head-driven Phrase Structure Grammar(HPSG)將用於建模包含句法和語義的語言信息 .在論文中, 我們首先討論如何使現行 HPSG 作為詞庫自動採集的基礎.我們描述如何使用維基百科的連接結構數據作為概念的聯繫. 最後 , 我們使用 English Resource Grammar(ERG)測試我們的採集模型,並發現它較基準確 .

詞庫自動採集的成功可進一步提高現有的自然語言問答系統和資訊採摘系統的準確性 .
此外,更好地利用來自語料庫 語義信息能進一步發展和延伸語言學理論.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The lexical acquisition task has been tackled by a wide range of researchers from diverse fields. In computational linguistic, researchers focused on developing models and algorithms to automatically induce the syntactical and semantic pattern of language from text corpora. In psychology, researchers are impressed by how quickly the children within the experimental settings to acquire languages in their first 36 months of life and based on these observations, various human language models are derived using statistical methods. In linguistic, research agenda has been proposed to discover whatever structure of universal grammar that forms the unique language faculty of human beings where this type of grammar can be used to instantiate the final state of grammar a particular person has in a particular linguistic environment.

Though the goal of lexical acquisition is theoretically unambiguous, finding the model that helps an agent (whether machine or human beings) to learn the languages and unambiguous goal can be reflected by the ultimate goals of researchers in different fields. Given various technical foundations of the re-

search field in which this topic can be related, it is necessary for any research endeavors on this topic to define which particular areas the research efforts are trying to tackle so that the work can be integrated in the discovery of the underlying ways why an agent can break the mysterious codes invented by human civilization.

In this thesis, we are interested in developing learning approach in the field to computational linguistic with the theoretical grounding from linguistic. The past decade has been the glorious period in the field of computational linguistic and linguistic. New learning paradigms from machine learning community and the large scale applications of fMRI - functional Magnetic Resonance Imaging, have helped the communities to critically revise many previous generations of model of natural languages and develop new insights into the way the languages work, both in the external entity as in the machine, or in the internal working of the brain. Mass scale deployment of machine learning method to language learning has come to a point where, according to experts in the field, the learning approach has been at its peak in the field of language learning. Language applications have become the dominant driving force in the field and language theories have been left to old school thoughts.

This research work tries to strike a balance between these two dominant forces in the field. Our major consideration in this work is to find out how the more theoretical linguistic framework devised by the theoretical communities can be flourished with the learning framework to automatically expand its syntactic and semantic coverage.

The other contribution of our work is for the first time, in the development of

language learning theories, to include more diverse semantic information into the framework in which the previous research efforts, under the directions of theoretical linguistic and inherited in the computational framework, the language learning is just a mere kind of syntactic matter. Through deploying algorithms to extract the fine-grain semantic information, the more accurate language structure can be induced.

1.1 Motivation

Linguistically sophisticated, lexicalist grammatical framework such as Head-driven phrase structure grammar-HPSG, Lexical functional grammar-LFG and their modeling capabilities have received a great deal of attention from both the theoretical and computational linguistic community for the past decade. Besides providing a more thorough analysis of a vast family of language data, from English, Chinese, German, Spanish, Russian, Greek and many other languages, they have been gradually proven to be relevant to the application level and a number of research laboratories have been working on their own version of this formalism. Applications of the formalisms to the natural language processing task have also started to emerge due to more efficient processing algorithms. Stochastic extensions of the formalism such as parse disambiguation and incorporation of the shallow processing mechanism are also emerging to improve the robustness of the grammatical framework. There are some large scale implementations of the grammar in the academic and research community.

The lexicalist framework spreads the linguistic information from a large num-

ber of derivation rules to the lexical items, leaving only a small number of phrasal constraints. For example, in a typical HPSG lexicon item, around 300 graph nodes are used to model the linguistic information. As the information is more concentrated on the lexical item, any missing item or incomplete item in processing a language fragment will immediately result in analysis failure. Though the significance of the lexical information stored on the lexical item is well understood, few analysis methodologies and systems that implement some variants of the lexical framework take care of these issues seriously, making the lexicalist framework apparently far more brittle comparing with other shallow processing mechanism in which stochastic extension has been applied.

At the other extreme, we have witnessed the achievements of unsupervised natural language learning for the past decades. This work, instead of using linguistically rich framework, focuses on deriving basic grammatical information such as grammatical dominance constraints from text corpora. However, the information contained in this type of framework is insufficient for real application usages.

This thesis is developed based on the hypothesis that the more general unsupervised lexical acquisition method can be adapted to model the acquisition process of the lexicalist framework, and thus provides a marriage of two traditions of language modeling. From the perspective of lexicalist tradition, the lexical items can be learnt rather than hand coded by grammarians or lexicographers. In learning communities, it can demonstrate the advantages of using additional features as in the grammatical framework to improve the learning performance.

1.2 New paradigm in language learning

There have been a lot of existing approaches in utilizing corpora in automatic induction of language structures. These structures may include phrase structure tree, feature structures and whatever representations applied in the linguistic framework under consideration. Based on these representations, different learning proposals are suggested such as grammar searching and clustering, distributional analysis, and so on. It is fair to say that the current representations and learning proposals have been quite abundant.

However, many of these approaches, no matter the type of representations and learning proposals they have, are based on one assumption which is generally faulty in language modeling and is further investigated in this work: The corpus is made up of merely a stream of sentences.

Further elaboration has to be made on this statement: For ordinary people, it is natural to say that passages, paragraphs are built by sentences where we can normally count the number of sentences within a paragraph, by considering the number of full stops for example. When applying to automatic lexical acquisition, it is again natural to treat the corpus as a mere list of sentences. The current approaches are based on the assumption that the lexical acquisition function can model the language data by fitting the learning function into these sentences.

However, getting deeper into the computational linguistic research would discover that sentences would hardly be an ideal basic unit for modeling languages. In question answering (QA) task, it has been widely known that the in-domain natural language question answering system such as the geo-

graphic system by MIT and the LUNAR system has achieved apparent success in the past. However, when these question answering systems are made to expand their original scope to become a more general-domain system, none of this system succeeds. Even today, general-domain question answering system is still a heat and enduring research topic where the research efforts span a range of contemporary NLP topics such as parsing, role labeling, question typing, name-entity recognition. The historical QA system, and even the today's system, still rely on the sentence based retrieval method for answer extraction. However, system of higher performance also uses discourse and domain information such as considering the information within the whole paragraph where the targeted sentences are located for better answer extraction. More contemporary system [36] uses the inter-relationships between concepts and texts to obtain a more accurate semantic to deduce the answers where the linkages between the texts are considered as many small cluster of discourses and the more accurate answers are extracted due to the more confined set of concepts and relations.

Someone may argue that the QA and lexical induction task may be completely irrelevant task given that they are of different origin and goal. However, careful reconsideration of these two separate research efforts may help to discover that the nature of the major operations used in these two tasks are the same: structure search; For question answering, algorithms are derived to search (to match) a particular answer nugget from sentences based on syntactic and semantic information such as grammatical category, selectional restriction information, question types; For lexical induction, methods are suggested to search (to match) a particular surface word order into a

hidden linguistic structure such as those used in HPSG.

In this work, we argue that the combination of the semantic information, similar to those in the QA research, is beneficial to the lexical acquisition task. However, critics may argue whether this type of semantic information can be found easily and how the information can be applied to the learning model.

1.3 Semantic Relations

The previous section raises a question about the availability of the existing semantic information for learning process. However, recent advance of the online encyclopedia provides an invaluable resources for this new type of approach.

Basically, the online encyclopedia, if considering the raw texts only, consists of a mere stream of texts and it is no difference comparing with the corpora used in lexical acquisition and many other NLP tasks where the corpora to be used include TreeBank, BNC, newswire stories and so on. Like the hyperlinked text, the online encyclopedia is built with links. The core number of links are around 20 millions, where these links represent a huge network of inter-relationships between concepts, entities, relations, and so on. Comparing with normal directory browsing where each item is a link to other web pages, the links are built "inside" the sentences. It means that the links are bracketed within the sentences. These links, comparing with ordinary web links, provide an intermediate linkage to other sentences or paragraphs that also mentions the same concept. Developing suitable algorithms can help us

to explore this network of concepts, the textual realization of the concepts, and the surface syntactic properties of the texts.

This entire new proposal [37], which has been previously published, is a novel idea in the semantic research as the previous work focuses on local compositional semantic based on the logical formula. However, this high level, abstract idea has to be formalized based on the rigorous logical approach. To model this inter-networks, the semantic relations are introduced to act as a medium to connect the concepts, textual realization and surface texts from the online encyclopedia.

However, many people have long arguing about the validity of using encyclopedia as a source corpus of NLP tasks where the writing styles of the encyclopedia may be different of normal language writing genre. There are a few points to be made for the statement. The first part is related to writing style. Comparing with encyclopedia, dictionary has been written in an even more odd writing styles where only the definitions of the words are given. WordNet-type resource is also of the dictionary style. However, many works such as lexical chains and information extraction have relied on these type of resources for modeling and it seems that given that type of unreasonable writing style comparing with normal genre, sensible work can be produced. Second, the online encyclopedia is written in a less formal way where in the traditional encyclopedia, experts and professionals write in a more confined context. This suggests that the writing genre of the online encyclopedia is more similar to the normal writing genre. Third, existing work also depends heavily on the corpus of a particular genre. For example, we can seldom find that a grammar trained in the news corpus can be applied to a non-news

corpus with even satisfactory performance.

Thus, we argue that the online encyclopedia can be a proper candidate in the NLP task and in this case, the lexical induction task.

1.4 Contribution of this thesis

Comparing with previous attempts in lexical acquisition task, and in particular in deep lexicalist framework, a number of novel techniques are used to enhance the acquisition performance.

- Strong focus in the type hierarchy system in acquisition: Previous efforts [3, 58] usually involve a non-incremental type information learning in which the type symbol plays the major role in determining the lexical type of a lexical item. However, no previous work investigates the possibilities of using the fine-grain information stored in a type to further improve the learning performance. In this thesis, the type system and the internal information are taken to be the first-class citizen in acquisition. Though the usage of confidence score measures, the acquisition algorithm can now acquire more accurate lexicon based on the fine-grain linguistic information in a type. The utilization of the internal information makes the overall acquisition process more robust.

- Use of semantic information: Another special technique used in this work is the usage of further semantic information in acquisition. Pre-

vious attempts [25, 1] concentrate on acquisition of syntax of lexicalist framework. The reason why previous works do not extend the acquisition algorithm with semantic is that it is hard to find the right corpus and to model the semantic information so that they can be integrated with the acquisition algorithm properly. Using Wikipedia corpus in this work provides a much well-founded semantic connection between different concepts, entities, and relations. At the same, through using the simple predicate logic and the links built in the Wikipedia, we can connect this information with the subcategorization frame structure in the HPSG to further enhance the acquisition procedure.

- Our proposed model can currently acquire the linguistic information on the unknown lexicon for an accuracy of 0.52. All the different type of words: noun, verb, adverb and adjective can be acquired. As not many previous work on lexicalist framework focuses on the acquisition problem and given the inherent challenge of the task of lexicon acquisition, we believe the envelop of the state-of-the-art performance of lexical acquisition has been pushed forward.

- Besides the open-class words that constitutes around 90% of the total number of lexicons, the algorithms are also applied on the acquisition of closed-class words such as stopwords, pronoun and articles. As the open-class words are much fewer in number but much diverse in grammatical properties, the overall accuracy of the acquisition of closed-class

word is about 0.321. As current attempts are focused on the acquisition of open-class word, however, from the perspectives of computational linguistic, closed-class words appear more often in complex syntactical properties such as agreement, unbounded dependencies, we believe that any experiments and models that shed lights on the acquisition properties of these words should benefit the more general research in linguistic.

- Using Wikipedia as the base corpus to acquire not only provides a more coherent set of documents and sentences for acquisition, the lexicon learnt is also of far more practical uses than other corpus. Very often, high-level ontologies and knowledge base rely heavily on the information and structure of encyclopedia. Exploiting the structure of the encyclopedia as a base of acquisition would demonstrate the tight relations between the lexical realization of concepts and the structure of concepts which would further benefit the development of more advanced knowledge and information system.

Besides the lexical acquisition task, the techniques and models developed in this work have been applied in other related work. In the QA system [36], the feasibilities of using the semantic information based on the Wikipedia are explored and the extra inter-relationships between the concepts and texts within this corpora are of tremendously uses in high-level semantic tasks such as question answering. This led us to further formalize the concept of semantic relations, originally used in the crude form in question answering,

for Machine Reading [37]. These semantic relations, which are basically a huge network of concepts, entities and relations, are further extended and used in some domains where high-level semantic retrieval is critical in the success of the field [38].

Chapter 2

Related Work

The background material of this project spans in a wide range of different research discipline, including theoretical linguistic and computational linguistic research.

2.1 Theoretical Linguistics

2.1.1 Overview

The major linguistic component used in this research work is the deep lexicalist linguistic framework [28, 7, 46]. Comparing with other shallow framework such as context free grammar [13, 14], lexicalist framework stresses on the distribution of the linguistic information from the grammar tree to the lexical items. The complex grammar rules, which often exist in the context free grammar, can then be generalized into a number of highly abstract construction rules. A number of framework exists in the deep lexicalist research area, including Generalized Phrase Structure Grammar (GPSG) [28, 29, 27, 48, 26],

Lexical Functional Grammar (LFG) [7, 5, 6, 4], Categorical Construction Grammar (CCG) [33], and Head Driven Phrase Grammar (HPSG) [46, 39, 32] and so on.

Generalized Phrase Structure Grammar [28] is the earliest ancestor of other grammatical framework as many of the linguistic constructs proposed in GPSG are later adapted to fit into other frameworks. Basically, this grammatical framework used feature structures as the base to represent the linguistic information on the lexicons and the construction rules. Feature structures, the attribute value matrix, are composed of a graph structure with the nodes representing a particular class of linguistic information and the edges representing attributes. Different frameworks use different set of nodes and edges to model the linguistic behaviour.

Just like the feature structures used in knowledge representation in AI research, the major operations on these features include subsumption and unification [45, 51]. Subsumption is the operation to compare the enclosure properties of the two feature structures. Unification is the major operation in combining the information from two feature structures. [50, 53, 52]

Many works have been focused in the development of these frameworks. In this work, we focus on Head Driven Phrase Structure Grammar for the development of the lexical acquisition approach. There are a number of reasons for choosing the HPSG framework:

- Research framework: Comparing with other framework, HPSG has been used to tackle much more linguistic phenomena than the other frameworks such as lexical semantics [16, 18], pragmatics and dis-

course [31, 34, 20, 21], statistical parsing [2, 40, 42, 44], lexical induction. These works have been a strong foundation in supporting the current proposed model.

- Research perspective: HPSG, comparing with other framework, is still very active in both the linguistic and computational linguistic [43, 41, 56] communities. Comparing with other framework where the domain may be either on theoretical or computational side, HPSG receives a great deal of attentions from both sides.

- Evaluation: A major obstacle in evaluating the linguistic research is the evaluation metric being used. Very often, frameworks are proposed but there is no gold standard to compare with. Recently, large scale grammatical work in English Resource Grammar [24, 19, 17] has been a critical resource to evaluate the learnt model and provides a more plausible way to evaluate the proposed learnt model.

2.1.2 Analysis

Current lexicalist framework such as HPSG has been a strong research community, spanning the influence to both the academic and the real industrial applications [22, 60]. However, there are a number of inherited deficiencies in this framework. First, it is the lexicon problem. Like Context Free Grammar [13], the lexicon items have to be filled with grammatical informa-

tion. In context free grammar, it is the part-of-speech symbols that dominate a particular word and with these symbols, grammar trees are constructed. Thus, it is necessary to know the lexical information, part-of-speech symbol, as in the context free grammar, to make the grammatical framework work. Same phenomena exists in lexicalist framework, including HPSG [46], where the lexicons are represented by feature structures and complex lexical information including part-of-speech is stored. Current work shows no trend in acquiring or learning this information automatically. Instead, the large scale English Resource Grammar [24, 19] projects work in the reverse direction, by hand-building several ten of thousands of lexical items for English language. To further make the research framework more applicable to the research tasks, it is necessary to develop models to acquire this information automatically.

In contrast with the CFG where the lexical information is a set of part-of-speech symbols, lexical framework uses a more complex feature structures where hierarchies and memberships are common phenomena, there are more structures and information to be exploited in the acquisition task and the current work demonstrates the feasibility of acquiring this lexical information automatically.

2.2 Computational Linguistics - General Learning

Two major foci are currently pursuing in the research community to tackle the problems of acquisition of linguistic information: Syntactic and semantic information.

For the syntactical information, existing approaches focus on the acquisition of the categorical information such as part-of-speech, subcategorization information such as frame and grammatical relations [10, 59]. The major techniques used involve formulating the problem as a classical machine learning problem where different linguistic information is represented as a list of features and various classifiers are trained and tested on these features. These features, which are usually the hidden syntactical information labels, are extracted from the partial parse results in the constituency or dependency formats or partial structural representation of sentences such as the grammatical relations from statistical parser.

In this formulation, the testing sentences are parsed using a shallow parser to obtain the dependency relationships between the words in a sentence. These relations correspond to the head-complement structure in which the tasks of acquisition try to recover. The resulting sets of grammatical relations are fed into a classifier, against a set of lexical information tag. Various training algorithms are applied on these relations.

There is one critical problem that makes this approach more general and it is the general problem of classification task - the sparsity of data. This approach requires heavy statistical pattern of different type of grammati-

cal relations. While many of the grammatical relations can be found in the training data set, imbalanced distribution of the relations such as excessive amount of head-modifier relations over the head complement relations would make the trained model biased towards a kind of relations and affect the overall performance of the model.

Besides subcategorization information, another commonly attacked problems of acquisition of syntactical information is the part-of-speech and many groups [12, 57, 8, 9] have attacked these problems from different angles including different machine learning model such as classification, sequence role labeling and so on. However, part-of-speech information is only a tiny bit of linguistic information that exists in the lexicon. More comprehensive lexical acquisition involves much more than the surface categorical labels of words and sentences. Grammatical relations, subcategorization information, semantic role labeling are a wide array of tasks that cannot be modeled by the part-of-speech label alone. However, this part of work constitutes one of the major attempts in automatically acquiring linguistic information and thus from the standpoints of this thesis, it should be included.

Another topic of interest to the lexical acquisition task is the semantic information. Comparing with syntactical information like part-of-speech, grammatical relations where the information to be acquired can be easily conceptualized to terms like NP, VP, PP, S or in the case of grammatical relations, subj, comp, mod, adjunct and so on. Semantic information has long been criticized for failure to be conceptualized. Instead of attacking the foundational philosophical problems, lexical acquisition and NLP has instead relied on some pre-built ontology such as FrameNet and the more general first-order

logic framework to be the representations of the semantic information [30, 47]. Approach in acquiring semantic information is to formulate the problem as word-to-meaning mapping problem in which the intuition is originated from cognitive science and artificial intelligence. In this formulation, the acquisition task is thought of a matching process between the surface words order of the sentences and the corresponding logical form. In Siskind's work [54], the mapping problems can be thought of as a set of refinement process to map the word symbol set to the conceptual symbol set and from the word symbol to the conceptual expression set. The criterion in matching is based on a number of rules, with the design principles based on the various difficulties and constraints such as multi-word utterance, referential uncertainty and etc. faced by the language learner in acquiring the native language skills.

In the later work of Thompson and Mooney [55], the problem of word-to-meaning mapping is more structurally defined. Instead of treating a sentence as a concatenation of words, the sentences are parsed into a tree structure with argument role relations and conceptual dependency representations. These representations are considered as potential interpretation of sentences. The task of the acquisition problem is then defined as an optimization problem in which the size of the lexicons spanned by the sentences is to be minimum. In other words, the task is to find a set of minimum lexicons with the highest generalization power to cover all the sentence in a corpus. This approach, however, considers the word as the carrier of semantic information only.

2.3 Computational Linguistics - HPSG Lexical Acquisition

A number of works have attempted to tackle the problem of processing unknown lexical items in lexicalist framework [23, 3]. Early attempts such as the incremental lexical acquisition [58] have introduced notions of learnable generalizable and specializable linguistic information so that through parsing, the linguistic information can be modified and refined.

The basic approach is to have the model to parse a large amount of sentences and through the generalization and specialization procedures to refine the linguistic information stored in a particular lexical entry. The sentences are not necessary related as what the model stressed was to have a corpus to train the model.

However, human intervention has to be made to decide which linguistic information can be modified. In addition, there are a lot of tunable parameters such as the learning rate and updating rate that receive little analysis while these factors are crucial in deriving accurate lexical items. This work discusses the very early attempts in which the lexicalist framework can be acquired through generalizing and specializing the feature structures of a lexical item. Besides the limitation mentioned before, few technical works and comprehensive experimental results are discussed in their works. In addition, their focuses were on Germen languages and in particular the biological domain, though this domain constitutes a significant fraction of general language uses. Sticking to this domain introduces extra difficulties in evaluating the acquisition approach as no general lexicon exists for this domain.

In addition, the approach mentions nothing about the utilization of the semantic information which has been one of the major building blocks of HPSG where the syntactical and semantical information is closely related to each other with structure sharing. It is interesting to see whether the semantic information, as in the design principle of HPSG, can be used to shed light on the effect of this information in the lexical acquisition task.

In another lexical acquisition work [25], part-of-speech tagger is used to provide hints to the parsing framework to determine the actual new words to process. In this approach, the part-of-speech tagger is first run on the corpus to extract the primary part-of-speech information for each word. Using the words, part-of-speech information and the lexical entries in the English Resource Grammar, a classifier is trained on these information to determine the feature structures of the words in the test set.

The part-of-speech tag used in the tagger is the general set from the Brown corpus where four major categories are further divided into subcategories. The four major categories resemble the four of the six categories currently used in HPSG grammar and thus some categories can never be benefited from the usages of the part-of-speech tagger. HPSG is not a framework of part-of-speech but of more comprehensive grammatical information such as semantic and subcategorization information and linkages between this information is achieved through co-indexing. Tasking the subcategorization frame which constitutes subjects, complements and specifiers, they are related to different role fillers through part-of-speech information and in addition, many other information such as inverted, passive and so on. Thus, part-of-speech information is far from sufficient in acquiring a full representation of HPSG.

By using the English resource grammar, the search space can be constrained compared with the previous approach. However, no attempts have been made to investigate the impact of introducing new words to the existing lexical entries. Thus the lexical entries learnt are filled with incremental amount of information, without tuning the lexical items to a particular domain.

In bootstrapping of lexical resources [1], different types of natural language processing techniques such as tagging, chunking, and dependency parsing are applied to induce the lexical items based on the English Resource Grammar [17]. Various methods based on morphology, syntax and ontology are used to induce the unknown words. Basically, their approaches try to extract whatever relevant linguistic information that can be acquired from the many different methods and merged them together in the unified HPSG formats.

It is not difficult to discover that newer attempts in acquiring high-precision HPSG language structure rely more on the machine learning approach and in particular, optimization. Generalization and specification relies on optimizing the underlying feature representation to cover a larger data set. Part-of-speech taggers are used to try to maximize the likelihood of the co-occurrence of the part-of-speech information with the targeted feature structures. We believe that language acquisition can be modeled as a task of optimization - to optimize the structure of languages based on the general language usages.

2.4 Learning approach

co The approach of modeling the acquisition task as learning separate grammatical labels from a large corpus ignores the fact that various lexical infor-

mation such as part-of-speech, subcategorization, argument role interacting with the others to define the actual lexical information of a word. The more precise modeling of languages can only be possible if the various kind of lexical information can be acquired under a unified framework in which the interaction of different lexical information can be modeled and explained.

In traditional corpus-based approach, researchers are more interested in exploiting the statistical distribution of different words within the whole corpus [11, 15, 49, 35]. The proximity of different sentences, different paragraphs are largely ignored in the acquisition task. Potential exploitation of the proximity may involve attaching weights to sentences and applying weight propagation algorithm to spread the weights among sentences. But sentences, paragraphs are grouped together not by random. Rather it is their respective semantic contents connecting these sentences or statements together to achieve the communication goal of natural languages.

The potential of using the semantic information can be understood as connecting or linking sentences to acquire the respective lexical information from the corpus. But the corpus choice is another concern. Traditional corpus, that groups all the experimentally interesting texts together, is not an ideal way to learn the semantic as the search space can be infinite and the results learnt can never be accurate. Also, the content of the texts are seldom highly correlated. Emerging corpus such as encyclopedia in which links are tagged by human editor provides a more semantically connected text. In addition, the texts within a page is usually edited by the same author and the same group of authors and thus different writing styles of the texts can be modeled within pages.

We propose an acquisition model in which different lexical information of words, including syntactic and semantic, are acquired based on the lexicalist grammatical framework. The particular grammatical framework chosen is HPSG due to its large empirical coverage of language data. The major corpus utilized during acquiring process is from the encyclopedia in which the tagged texts are modeled as the semantic linkage between different word sense, sentence meaning, and paragraph themes.

Chapter 3

Background

As our learning approach is based on the typed feature system, a few backgrounds about this system are explained, followed by a review of a current implementation of the HPSG, English Resource Grammar, and the adaptation of the typed feature system for scoring function.

HPSG (Head-Driven Phrase Structure Grammar) [1], which belongs to the family of lexicalist framework in linguistic theory, is built on the constraint satisfaction. The primitive objects in this theory are the feature structures, which are used to formulate the basic modeling primitives in HPSG, including words, phrases, clauses and grammatical constraints or construction constraints. In this section, we explain the foundational concepts of this theory.

[1] In this work, the grammar - the type figures and lexical types, are adapted from [32]. Interested readers please refer to the grammar book for complete reference of all the type hierarchies, phrasal and clausal constraints.

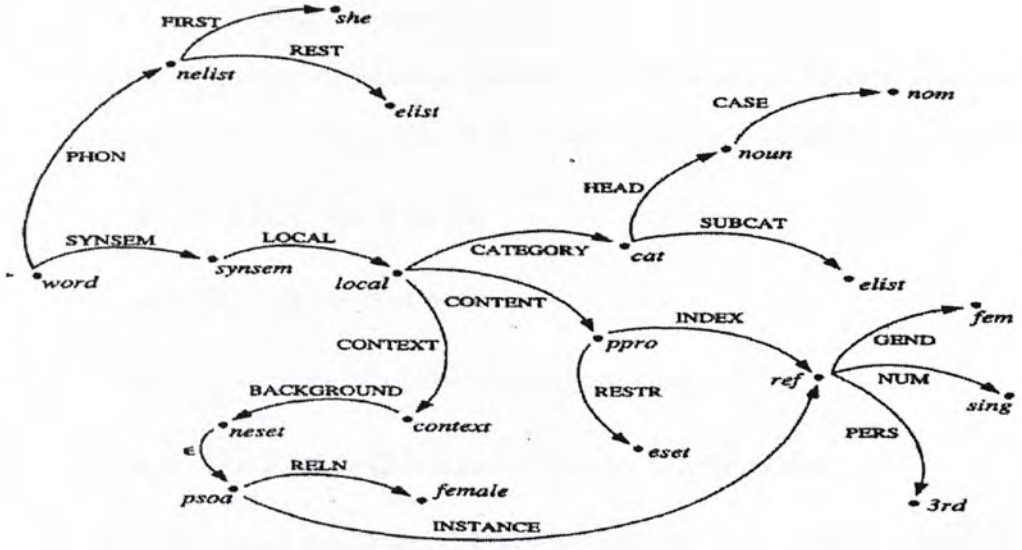


Figure 3.1: Typical feature structure

3.1 Modeling primitives

3.1.1 Feature Structure

The basic modeling object to be used is the feature structures. A feature structure can be thought of a graph structure where the nodes represent a value storage location and the edges represent the attribute of a particular value storage location. A typical feature structure is sketched in Figure 3.1. In HPSG, the values and features of the feature structures are designed from grammarians who concerns about the linguistic abstraction of using the particular model to explain different syntactic and semantic variations across different languages.

The mathematic formalization of the linguistic objects is given below: One of the common definitions of the typed feature system is given as follows:

Definition 1 (Typed feature structures)

A typed feature structure is defined on a finite set of Feature $Feat$ and a type hierarchy $< Type, \sqsubseteq >$. It is a tuple $< Q, r, \delta, \theta >$, where:

- Q is a finite set of nodes,
- $r \in Q$ (r is the root node)
- $\theta : Q \rightarrow Type$ is a partial typing function
- $\delta : Q \times Feat \rightarrow Q$ is a partial feature value function

Two additional terms are defined for different type nodes, depending on whether it is a phrasal node or lexical node.

Definition 2 (Phrasal node, Lexical node)

- $Q_{lexical}$ is a lexical node if the graph spanned by this node is a lexical item
- Q_{phrase} is a phrasal node if the graph spanned by this node is the result of applying phrasal constraint to the daughter constituents.

In the lexical node, the root node belongs to one of the subtypes of "sign", with the feature appropriate for this type spanning from the root node. The lexical information of the lexical entry is spread in the feature graph. The ideal lexical acquisition task should take into account of every single piece of linguistic information such as PHON, SYNSEM and LOCAL and so on for acquisition. However, this may make the problem intractable given the current processing power; the task is made more computational feasible by using a more well defined lexical type system. The type system we are using

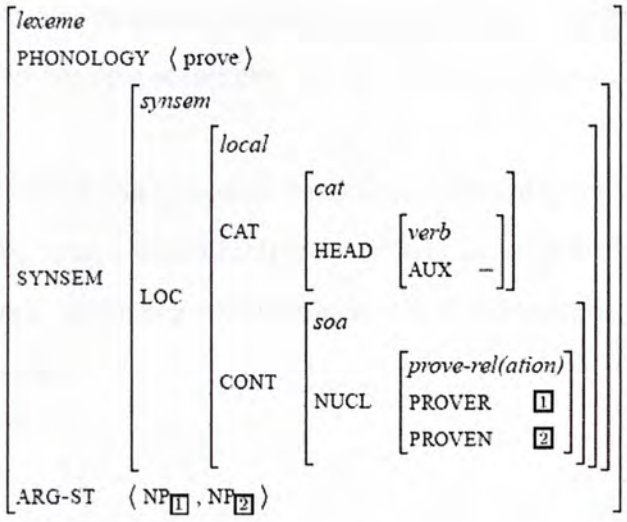


Figure 3.2: Typical feature structure

is the English Resource Grammar. In this implementation of HPSG, every lexicon belongs to one of the lexical types, which are the further subtypes of "sign".

To make the representation less clumsy, it is usually assumed that the attributes are omitted in the feature structure diagram. As well-formed feature structures state precisely what type of attributes are allowed in a particular value, the actual attributes permitted by a particular values can thus be deduced from the feature structures.

3.1.2 Word

Lexicon is the major backbone of the lexicalist framework. HPSG grammar has its own inventory of words and type system to model the different lin-

guistic information exhibited by different lexical items. The following feature structure is an example of common words, which is written in attribute-value matrix form.

The different attributes represent the different type of linguistic information stored on this word. Basically, there are three categories of information in this framework. Phonology, Syntactic-Semantic information, and the argument structures.

Phonology

As the full-feature grammatical framework, it is necessary to model any type of linguistic information on a typical lexical item. The five classes of information include phonology, morphology, and syntax, semantic and pragmatic. As the current thesis and most of the HPSG concerns the syntactic and semantic information, the phonology part is largely ignored in the current research paradigm.

SYNSEM

SYNSEM constitutes most of the information stored in a lexical entry, as suggested in the American-style structuralist linguistic paradigm. This attribute captures the major allowable syntactic and semantic information stored in a word.

The design of this part, and any co-indexing occurring in this module, is considered as a feature structure geometry as designed by the grammarians. As the current thesis concerns more on the acquisition problem of the framework rather than the plausibility of the linguistic constraints, which is an ongoing

TYPE	FEATURES/TYPE OF VALUE	IST
sign	[PHONOLOGY *list(form)* SYNSEM *synsem* CONTEXT *conx-obj*]	*feat-struc*
phrase	...	*sign*
lex-sign	[ARG-ST *list(synsem)*]	*sign*
lexeme		*lex-sign*
word		*lex-sign*
synsem	[LOCAL *local* SLASH *set(local)* WH *set(scope-obj)* BCKGRND *set(fact)*]	*feat-struc*
local	[CATEGORY *category* CONTENT *sem-object* STORE *set(scope-obj)*]	*feat-struc*
category	[HEAD *part-of-speech* SUBJ *list(synsem)* COMPS *list(synsem)* SPR *list(synsem)*]	*feat-struc*
...	...	...

Figure 3.3: Typical feature structure

work in the theoretical linguistic research, we assume the feature structure geometry as in the current state-of-the-art HPSG design. The tables as in Figure 3.3 shows the design of the feature geometry of some attributes and its type of values.

Syntactic information

The major syntactic information stored in the SYNSEM includes the local information, which include the major category and subcategorization infor-

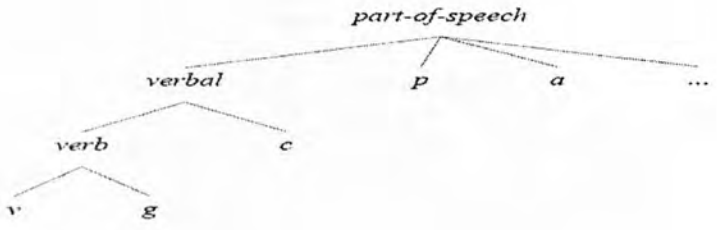


Figure 3.4: Categorical information

mation. The LOCAL attribute models the information to remain largely unattached in the other linguistic phenomena such as unbounded dependencies and clausal constructions. The SLASH attribute is related to unbounded dependencies where the linguistic information is accumulated up the parse graph in case unbounded information exists.

Another attribute is the WH attributes, which describes the interrogative, exclamative, topical and other phenomena in languages.

The major category includes the commonly used part-of-speech tag as in other grammatical framework. Only five major categories exist in this framework including noun, verb, adjective, adverb, and determiner. However, based on the value of the category features, a number of features are valid for a particular of part-of-speech as shown in Figure 3.4, Figure 3.5. For example, the noun value has extra attributes to represent the co-indexing information with the semantic information. The verb value has information such as tense, verb form and value for passive forms.

The subcategorization represents a list of SYNSEM objects in which this

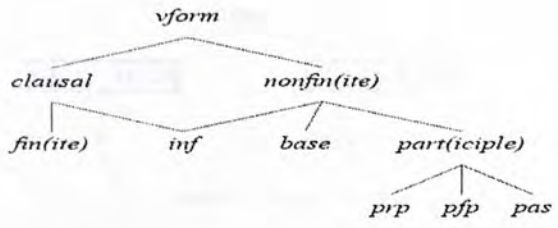


Figure 3.5: Verbal information

lexical entry can combine with to form a phrase. The three major group of subcategorization information include subject, complementizer and specifier. The subcategorization information is modeled as a list of SYNSEM objects. The order on the list represents the obliqueness of the SYNSEM information to be formulated in the phrase. This arrangement facilitates the modeling of the case, role assignment, semantic selection and head-valence agreement of linguistic phenomena.

Semantic information

The semantic information, captured by the CONT attributes, represent the major relational information of a particular word. This attribute, which is designed based on the principles of Charles Fillmore's semantic role modeling, is used to capture which particular words in the phrases can fill the role of the relations. Consider the PROVE relation as above, two roles are available for this relation, including the PROVER and PROVEE. However, unlike the general case theory which models the relation as the selection of semantic

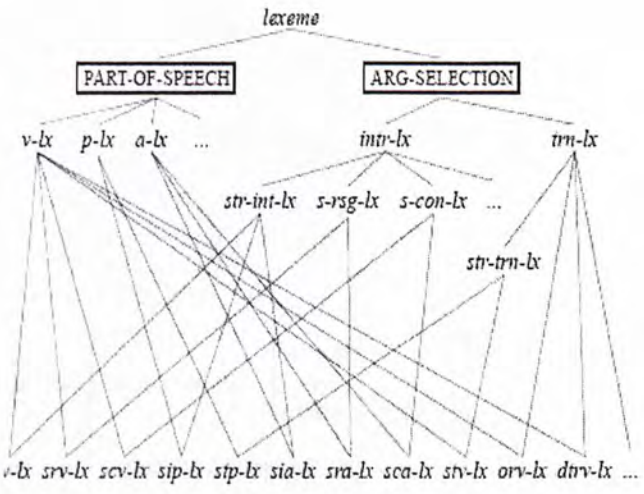


Figure 3.6: Argument Selection Hierarchy

information only, HPSG combines the semantic and syntactic information in modeling the CONT attribute. The particular role in the relation is not selected only on the semantic information but also the syntactic category of the words.

Argument Selection

The argument selection part models the number of argument in which this word can subcategorize with. The three major class of subcategorization information include the intransitive, transitive, and ditransitive, i.e. subcategorizes with one, two or three arguments. This information is co-indexed with the CONT information in the semantic information of the lexical entry. The diagrams shown in Figure 3.6 and Figure 3.7 show a rough hierarchy of the argument selection information.

a. $v\text{-}lx \Rightarrow \left[\text{SS|LOC|CAT} \begin{bmatrix}\text{HEAD} & v \\ \text{SPR} & \langle\,\rangle \\ \text{SUBJ} & \langle \text{XP} \rangle\end{bmatrix}\right]$

b. $p\text{-}lx \Rightarrow \left[\text{SS|LOC|CAT|HEAD} \quad p\right]$

c. $a\text{-}lx \Rightarrow \left[\text{SS|LOC|CAT|HEAD} \quad a\right]$

d. $trn\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \langle \text{NP}, \text{NP}, \ldots \rangle\right]$

e. $str\text{-}intr\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \langle \text{NP} \rangle\right]$

f. $s\text{-}rsg\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \langle [\text{LOC } \boxed{1}], [\text{SUBJ } \langle [\text{LOC } \boxed{1}] \rangle] \rangle\right]$

g. $s\text{-}ctrl\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \langle \text{NP}_i, [\text{SUBJ } \langle \text{NP}_i \rangle] \rangle\right]$

h. $str\text{-}trn\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \langle \text{NP}, \text{NP} \rangle\right]$

i. $otv\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \left\langle \text{NP}, [\text{LOC } \boxed{1}], [\text{SUBJ } \langle [\text{LOC } \boxed{1}] \rangle] \right\rangle\right]$

j. $dtrv\text{-}lx \Rightarrow \left[\text{ARG-ST} \quad \langle \text{NP}, \text{NP}, \text{NP} \rangle\right]$

Figure 3.7: Argument Selection information

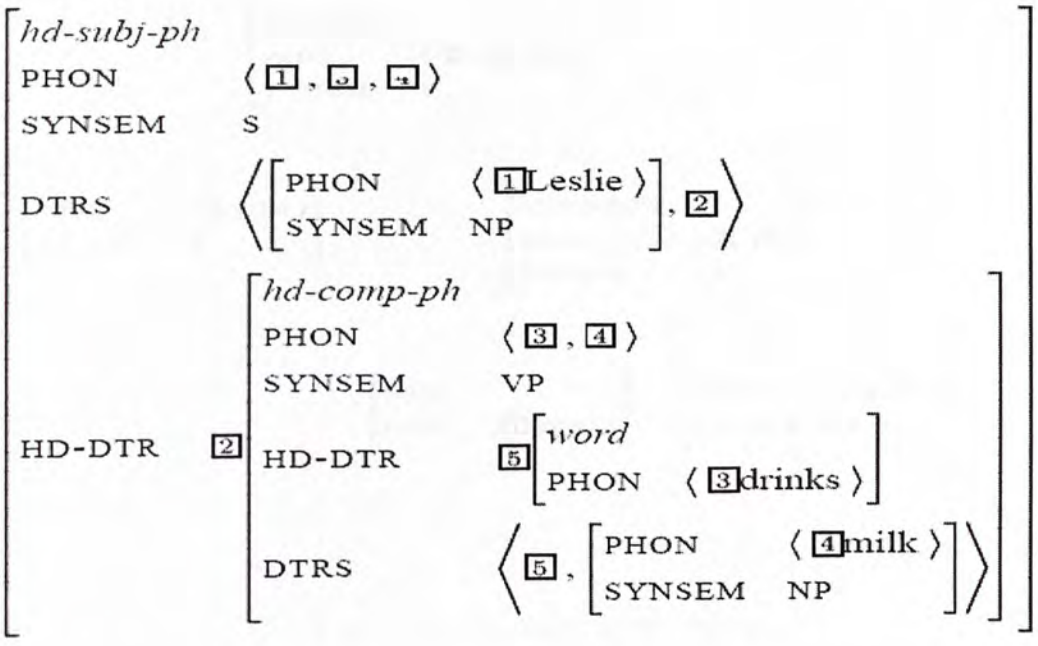


Figure 3.8: HPSG Parse of sentence

3.1.3 Phrase

As described in the previous section, the phrase is also modeled as the feature structure, thus it is not hard to find that the feature geometry of the phrase is largely the same as the words. However, the function of phrase is to group the word together to a hierarchy. Features for joining different words exist in the phrase. The following example shows a HPSG description of a sentence with the phrase information. The HPSG parse of a sentence into a phrase structure is given in Figure 3.8:

To represent the structures in a form commonly used by grammarians in other framework such as CFG, the feature structure can be represented as a

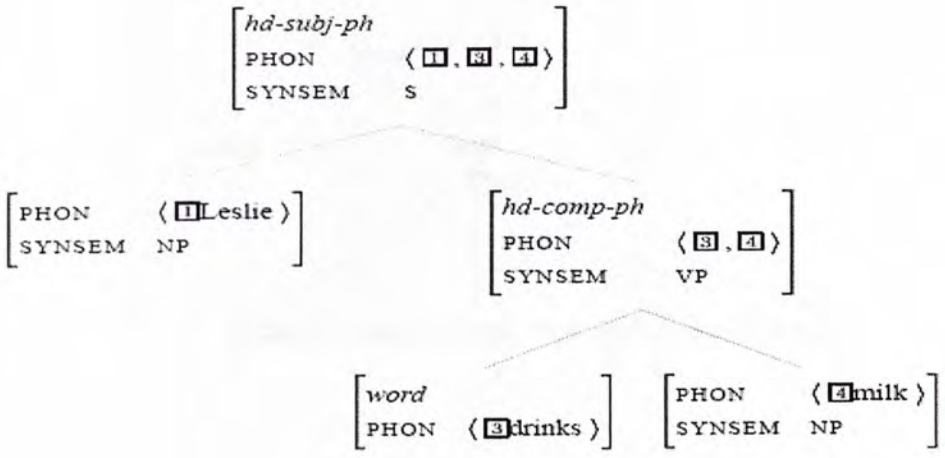


Figure 3.9: Typical feature structure

tree format as shown in Figure 3.9: What remains in the description of the phrase section is the phrase type. The phrase type is explained by which subcategorization information is to be filled by the other words in the sentences. A number of phrase types is sketched below in Figure 3.10: The allowable information under a phrase is sketched: However, given the phrase types, we need some ways to decide whether the terms can be combined to form a particular phrase. This decision is formulated by the grammatical constraints and the phrase constraints. The constraints of the different phrase type are listed in Figure 3.12.

3.1.4 Clause

Finally, there is clause information. Comparing with the previous generation of the framework, the maximal phrase type is now cross-classified with

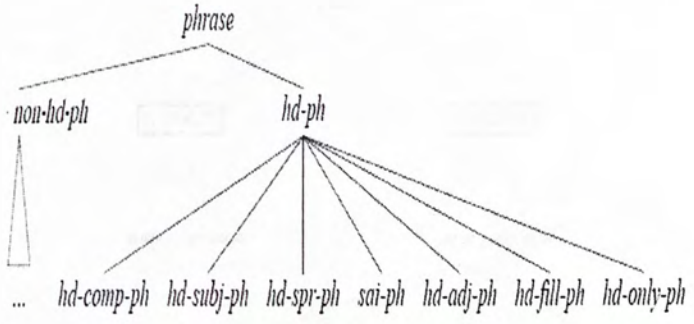


Figure 3.10: Phrase Type

TYPE	FEATURES/TYPE OF VALUE	IST
sign	[PHONOLOGY *list(speech-sound)* SYNSEM *canon-ss* CONTEXT *conx-obj*]	*feat-struc*
phrase	[DTRS *nelist(sign)*]	*sign*
hd-ph	[HD-DTR *sign*]	*phrase*

Figure 3.11: Phrasal linguistic information

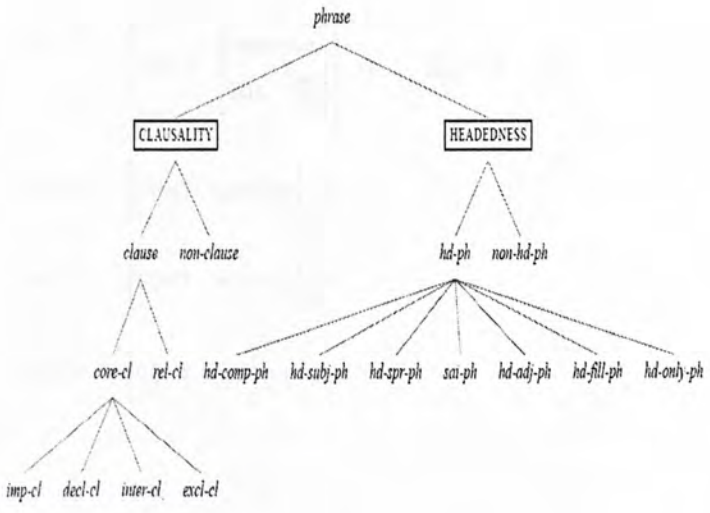


Figure 3.12: Phrasal Constraints

two different information. One is the headedness information, i.e. the non-maximal phrase type and the other is the clause information. The addition of the clause information provides a clear structure to differentiate the information in the phrase not from the X-bar category in Figure 3.12 and Figure 3.13 show some of the clause structures and the relevant constraints.

To summarize this section, the diagram in Figure 3.14 shows the phrases and clauses that will be related in this thesis.

3.2 Wikipedia Resource

The Wikipedia source is a new type of text materials in which the texts have been linked together by the Wikipedians. In this section, we briefly investigate the nature of this corpus and the type of information - semantic

decl-cl: $\left[\text{CONT} \begin{bmatrix} \textit{austinian} \\ \text{SOA} \ /\boxed{1} \end{bmatrix}\right] \rightarrow \ldots \mathbf{H}\left[\text{CONT} \ /\boxed{1}\right]\ldots$

inter-cl: $\left[\text{CONT} \ \textit{question}\right] \rightarrow \ldots$

imp-cl: $\left[\text{CONT} \ \textit{outcome}\right] \rightarrow \ldots$

excl-cl: $\left[\text{CONT} \ \textit{fact}\right] \rightarrow \ldots$

Figure 3.13: Phrasal Constraints

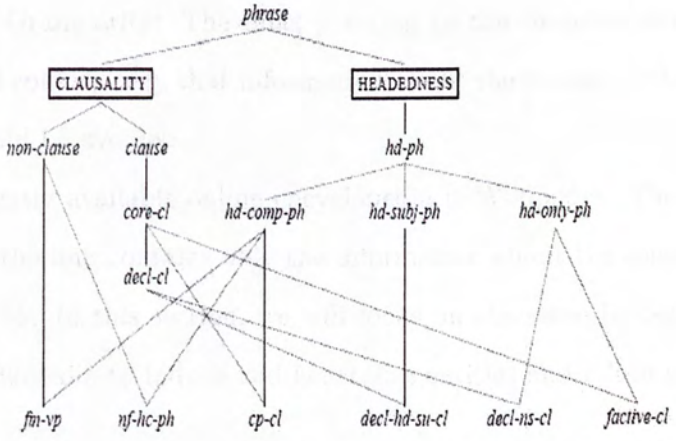


Figure 3.14: Phrasal Constraints

relations, that we can extract all the learning model.

3.2.1 Encyclopedia Text

Encyclopedia texts provide hints and suggestions to the relationships between entities and relations to the world. They act as a foundation in which further information between entities and relations can be generalized. This type of texts has the following properties:

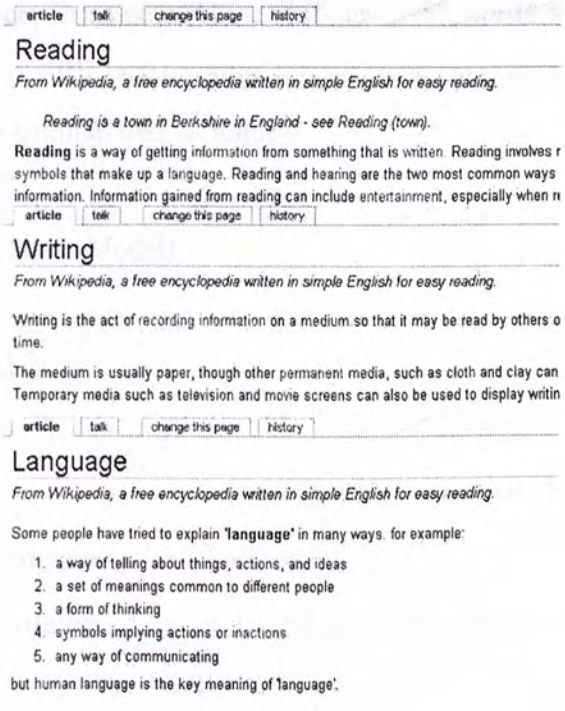
1. Partial Sense Linkage: The texts are partially tagged with relationships. Given a sentence within the text corpus, some words within the sentences are linked to the senses in which the words represent. The linked sense may be other essays or paragraphs containing more detailed semantic description of the given word.

2. Linkage Granularity: The links pointing to the targeted senses from a word should contain only that information about the senses, irrelevant information should be avoided.

One recently available online encyclopedia is Wikipedia. The senses are tagged and the link contains only the information about the sense as shown in Figure 3.15. In this section, we will focus on the sense linkage inherited from the Wikipedia to induce and generalize entities and relations.

3.3 Semantic Relations

The sense model describes the macroscopic relations between different senses, pages, and paragraphs within the corpus. It also provides the mathematical


article talk change this page history

Reading

From Wikipedia, a free encyclopedia written in simple English for easy reading.

Reading is a town in Berkshire in England - see Reading (town).

Reading is a way of getting information from something that is written. Reading involves r
symbols that make up a language. Reading and hearing are the two most common ways
information. Information gained from reading can include entertainment, especially when re

article talk change this page history

Writing

From Wikipedia, a free encyclopedia written in simple English for easy reading.

Writing is the act of recording information on a medium so that it may be read by others o
time.

The medium is usually paper, though other permanent media, such as cloth and clay can
Temporary media such as television and movie screens can also be used to display writin

article talk change this page history

Language

From Wikipedia, a free encyclopedia written in simple English for easy reading.

Some people have tried to explain **'language'** in many ways. for example:

1. a way of telling about things, actions, and ideas
2. a set of meanings common to different people
3. a form of thinking
4. symbols implying actions or inactions
5. any way of communicating

but human language is the key meaning of 'language'.

Figure 3.15: Sample entries extracted from Wikipedia, showing some of the link structures between "reading", "writing" and "language".

foundation in describing the mechanism of entities and relations induction. In the encyclopedia, each page describes a particular concept in which the words represent. Each particular page can be thought of representing the sense of the word. In any pages, there are a number of paragraphs, describing a particular aspect of the senses. In each paragraph, there are a number of sentences where in some of these sentences, the words within these sentences are tagged with relationships to the other encyclopedia page. These relations can be summarized as follows:

Definition 3.1: (Sense Model)

Given a sense p_i:

$$\begin{aligned} p_i &= <\{paragraph_{ij}\}, \{insense_n\}> \\ paragraph_{ij} &= \{sentence_{ijk}\} \\ sentence_{ijk} &= <\{word_{ijkm}\}, \{relation_p\}> \end{aligned}$$

Where:

<>:	The notation for list of tuples
:	The notation for a set of elements
$paragraph_{ij}$:	The paragraphs within the sense p_i
$sentence_{ijk}$:	The sentences within the $paragraph_{ij}$
$word_{ijkm}$:	Words within the sentences.
$relation_p$:	Relations extracted from the sentence $sentence_{ijk}$.

Detailed description on this attribute
will be given in next section.

$insense_n$: The set of senses pointing from
other senses to this sense.

For a word $word_{ijkm}$:

$word_{ijkm} = p_m$, if a link exists, pointing from the current sense p_i to the targeted sense p_m. And,

$word_{ijkm} = 0$, otherwise.

Example

Consider the example in Figure 3.16, there are three paragraphs within the sense "reading" and ten sense links within these paragraphs.

Let p_1 be the sense of the page "reading".

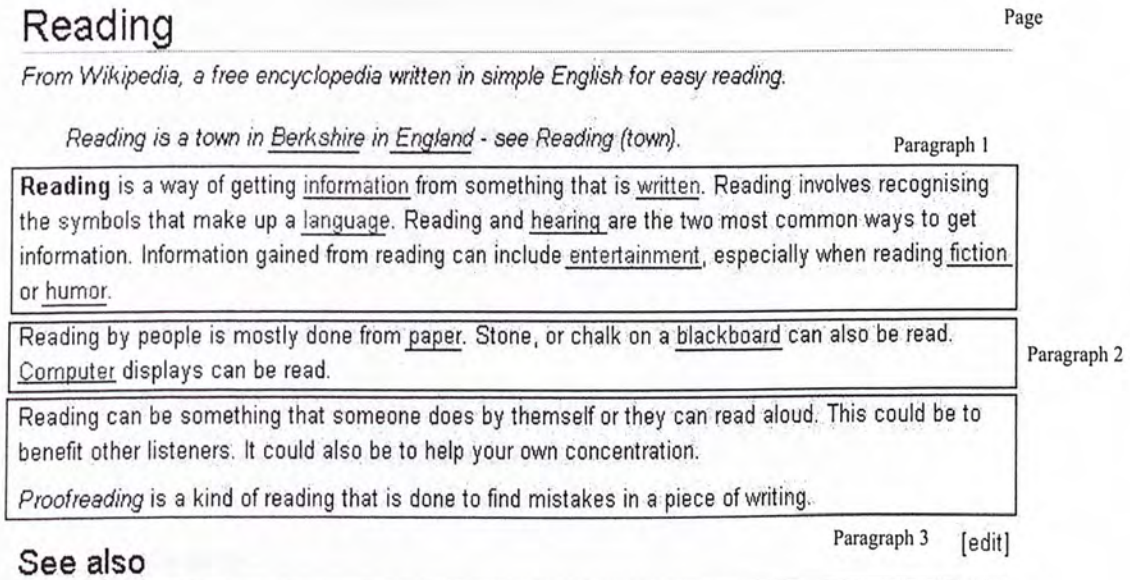


Figure 3.16: Sense model of the entries of Encyclopedia, showing the relationships between page, paragraphs, sentences, and words

$$sentence_{111} =< \{p_{information}, p_{written}\}, \{relation_1\} >$$

$$sentence_{112} =< \{p_{symbols}, p_{language}\},$$

$$sentence_{113} =< \{p_{hearing}\}$$

$$sentence_{114} =< \{p_{entertainment}, p_{fiction}\}$$

$$sentence_{121} =< \{p_{paper}, p_{blackboard}\}$$

$$sentence_{122} = \{p_{computer}\}$$

For the paragraph:

$paragraph_{11} =< \{sentense_{111}, ..., sentence_{114}\}, \{relation_p\} >$

$paragraph_{12} =< \{sentence_{121}, sentence_{122}\}, \{relation_p\} >$

For the page:

$p_1 =< \{paragraph_{11}, paragraph_{12}, paragraph_{13}, \{insense_n\} >$

Chapter 4

Learning Framework - Syntactic and Semantic

The background presented in Chapter 2 on the constraint-based formalism, introduction of the HPSG framework and the representation of semantic structures in Chapter 3 constitute the foundation of the description of the learing framework of the syntactic and semantic structures of lexicons, as described in this chapter. Before explaining the technical details of the model, two objectives are described here so as to act as a guide for the development of the approach and make the acquisition problem more focused.

- Adding linguistic information for unknown words: Unknown words are defined as the lexical items without any explicith linguistic information. This happens as the lexicon cannot cover all the available lexical items, due to the limited coverage of the lexicon or new words being coined in new context. Adding information to new words involves attaching correct linguistic information to lexical items so that the information

remains valid for the later processing of sentences containing these new words and any acquisition method should fill in this unknown information automatically.

- Updating existing linguistic information for words: Existing linguistic information is updated so as to recover the previous error generated and propagated in the previous acquisition process. This step is critical as in the early stage of the acquisition, the lexical item is underspecified, i.e. containing not enough information to determine its syntactic properties. Updating the information can make the acquired lexical items cover a larger portion of the training data set so as to improve the overall accuracy.

These two objectives of lexical acquisition demand novel methods in managing and collecting the linguistic information from both known words and unknown words.

This chapter would proceed as follows: The type feature scoring function is first defined to capture the hierarchical structure of the lexicon in the lexicalist framework. These scores would be used to further define the confidence score function that measures whether the linguistic information on a lexicon needs to be updated in a larger context. Algorithms are then proposed to generalize and specialize the linguistic information based on the score and the semantic information. The remaining sections of this chapter describe the ways the semantic information is collected from the Wikipedia

corpus for a better learning strategy.

4.1 Type feature scoring function

The acquisition model proposed in this thesis makes heavy uses of the hierarchical organization of the feature structures which are the major modeling primitives in HPSG framework. The hierarchical organization provides a means to generalize and specialize the lexical information in the lexical items. Before defining the acquisition algorithm, it is crucial to define a numerical magnitude to capture these relations and this is the **type feature scoring function**.

The **type feature scoring function** is a measure to capture the hierarchical relations between lexical items and lexical types and its usages are further explained in the next section on confidence score of lexical entry. In this section, we define how the numerical quantities are associated with the type system and how the magnitudes are defined.

In the deep lexicalist framework, all the lexicons are defined in a hierarchical organization where more specific items are at the bottom of the hierarchy while less specific items are at the top. The type system defines which items are at the top by the amount of information it contains in the type. In the type system, types are subsumed by other types if the other types contain less specific information, i.e. they reside in a upper position in the type hierarchy. A typical type hierarchy is shown in Figure 4.1, showing the relative subsumption order of the types.

We define the **type feature scoring function** as follows:

Definition 4.1: Type feature scoring function

A Type feature scoring function $TF_{score}(T)$ where $T \in Type$ and $Type$ is the set of available type in the system, is defined as:

$TF_{score}(T) = x \in$ real number set if $\forall t \sqsubset T = \emptyset$

$TF_{score}(T) = \sum TF_{score}(T_1)$ where $T_1 \sqsubset T$

where the symbol $\sqsubset$ in $A \sqsubset B$ means that type A subsumes B in this type system.

In other words, the base type that has no other type subsuming it receives a type feature score of any real number. As the magnitude of this number is not of particular interest to this model. We define this number to have a value of 1. The other types where they are subsumed from the other types receive a score which is the sum of all the parent types in the type system. As in HPSG and other type system, it is possible to have multiple subsumption from more than one type. The subtype hierarchy as in Figure 4.1 shows an example of multiple subsumption.

$T_D \sqsubseteq T_B$

$T_D \sqsubseteq T_C$

$T_B, T_C \sqsubseteq T_A$

In this example, type A is the base type where type B and type C subsume from it. Thus, type B and C receive a score equal to that of A. In the lower hierarchy, type D is subsumed from both type B and C and thus it

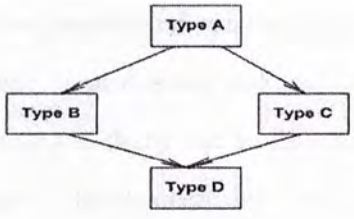


Figure 4.1: The Type definition with multiple subtypes

receives a score which is the sume of type B and type C.

In the typical HPSG grammar, the type is usually subsumed by more than 3 types such as those in the construction rules and the lexicons. In addition, the grammar itself contains about six to nine layer of hierarchical organization with multiple subsumption relations. The **type feature scoring function** as defined above is applied to the HPSG grammar for calculation of the confidence score.

4.2 Confidence score of lexical entry

As defined in the second objective of the acquisition problem, the existing lingistic information on the words is updated so as to cover a larger portion of training data set. To decide whether the information needs to be updated, the confidence score is defined to guide the decision.

The **confidence score** measures whether the currently stored linguistic information in a lexical entry needs to be updated in the coming process. Unlike the previous attempt in the acquisition of lexicalist framework, we use this measure to increase the robustness of the update process of the linguistic information.

The definition of unknown words and existing words are also not as strict as the previous treatment of processing unknown information in lexicalist framework. Instead of using a sharp cut to differentiate between unknown words and existing words in the lexicon, we design our approach based on how confident a lexical entry to be. A relatively higher confidence score means that the lexical item should not be updated so frequently as it has already covered a large set of data while the items with relatively lower score can be further modified so as to increase the likelihood of a particular lexical type for the item.

After processing a sentence with the hypothesized lexical type, results can be obtained in whether the sentences can satisfy all the constraints and in the case where no complete parse is obtained, how many words can the partial parse cover? Based on the values of the confidence scores of the words in the sentence, this approach will decide which words in the sentence should be further processed to make the candidate entries getting a higher confidence score, acquiring more accurate lexical information and covering a larger data set.

The confidence score is defined as follows:

Definition 4.2: Confidence score of an entry

Let $c(w)$ be the current number of sentences processed involving this lexical entry.

Let $c(total)$ be the current number of sentences processed:

Confidence: $p(w) = \frac{c(w)}{c(total)}$

A few comments have to made in the current formulation. First, variation exists in different lexical items in processing the sentences. Some words, such as the closed-class words are the entries with larger variations while the variation of some open-class words are less fluctuated. The existence of variation means that the lexical items can be easily picked for update and make the acquisition process feasible. Second, the confidence score is independent of context. This formulation simplifies the acquisition problem to be more manageable.

4.3 Specialization and Generalization

The type system is organized in a hierarchical fashion where different lexical items can reside in different positions in the hierarchy so as to maximize the coverage of the data set. Since the less specific item carries less information, any lexical item that is too specific in the current failure parse of a particular sentence may possibly be valid in the next parse if the types of this item can be generalized, i.e. containing less information. On the other hand, if the current lexical item contains a high confidence score and can cover a large portion of data set, they can be made more specialized to see whether further information can be added to the item. This dynamic of generalization and specialization captures the major approach in the current acquisition algorithm.

After comparing the confidence score of different words in a sentence, a selected word based on the lower relative magnitude of the score is then applied to the updated process to decide whether the current lexical information is

sufficient for processing the current sentences. If the information is sufficient, this lexical entry is comparatively accurate in processing the current sentences and the approach has more confidence in using these words for further processing. In other words, the current information on this lexical entry can be reused in later processing of these words. The current successfully processed lexical entry is retained and is further specialized for the later processing of the sentence. This is the **Specialization** phase.

Specialization

For $T_i \in Type$,

$Spec = \{\forall T_x \sqsupset T_i \bigwedge \forall T_j \sqsubset T_x \bigwedge i \neq j\}$

The above step collects a set of types for specializing a lexical entry.
Very often, the lexical information on the current entry is insufficient or inaccurate to process the current sentences as the words may exhibit different lexical properties in different situations or there is previous acquisition error, which results in failing to satisfying the phrasal or clausal constraints. In this case, the lexical information has to be generalized so that a less specified set of the lexical information should be applied to discover the right set of lexical information applicable to this entry. This is the **Generalization** phase.

Generalization

For $T_i \in Type$,

$Gen = \{\forall T_x \sqsupset \forall T_y \sqsupset T_i \bigwedge \forall T_z \sqsubset T_x \bigwedge z \neq y\}$

4.3.1 Further Processing

In case of failure to satisfy all the phrasal constraints, or in other words, parse failure, the node dominating the constituents with a total largest confidence score is excluded from processing as they have a relatively high confidence for further processing. Excluding these fragments and the previously processed lexical entry, the one with the lower confidence score is selected for processing.

Processing

Let $Q_{lex} \in Q$ be the phrasal node containing a set of lexical nodes dominating by the phrasal node:
$\max \sum TF_{score}(T)\ \forall T : q_{lex} \rightarrow T$

The dominant node with the maximum score is chosen. This node will be kept constant in the current processing as it has covered a set of constituent with a total highest confidence score. A lexical entry with the lowest confidence score outside this node is chosen for generalizing and specializing.

4.3.2 Algorithm Outline

Observed that the initial stage of the algorithm requires a variation of confidence score for the algorithm to start, in the case where the lexicon contains only the "unknown words", the algorithm will cease to start, some existing

examples of lexicons are needed to start the process. In the current implementation of the HPSG in English Resource Grammar, the lexicon itself is acted as the "training" examples for the algorithm. The algorithm works as follows: A lexical entry is selected and processed. The selected lexical entries, combining with other existing lexical entries, are put to the parser with HPSG grammar. The result of the parse is then analyzed. If the parse fails, the lexical item with the lowest confidence score is extracted and analyzed. If the parse succeeds, the lexical item is specialized and fed to the parser again. If the parse fails, the generalized lexical item is then fed to the parser again. In feeding the lexical item, the confidence scores of other items are also changing as well and after a particular generalization and specialization phase, the new entries with the lowest confidence score are then generalized or specialized. The updated entry is then used in further processing sentences. The core of the algorithm is described in Figure 4.2.

After processing the sentence sets and through the process of generalization and specialization, a set of lexicon with stable magnitudes of confidence score is generated. And this lexicon set is the acquisition result of the model.

4.3.3 Algorithm Analysis

The whole algorithm can be considered as finding a local optimum set of lexicons in the local context. When a sentence is processed, the algorithm tries to optimize the processing of the current sentence based on the confidence score before processing other sentences. The approach tries to extract the most of the information from the current instance before processing the next

Algorithm: (Lexical Acquisition)

Assume the corpus is composed of
a stream of sentences:

Let *corpus* be the corpus
composing of a list of sentence:

$$corpus = < sentence_i > \text{ for } i = 1, 2, ..., n$$

Let *cons* be the set of constraints
of the grammar, in this case, it is
the ERG grammar

For a sentence consists of m words, the
sentence can be represented as:

$$sentence_i = < p(w_j) > \text{ for } j = 1, 2, ..., m$$

where $p(w_j)$ is the confidence score of the
words defined as above.

For $\forall sentence \in corpus$,
While convergence rate < threshold,
$w_{select} = min_{p(w)} < p(w_j) >$

Select the current state of the word w
If w is the maximal sort,
If $cons \sqcup sent \neq \perp$
$p(w_j) = p(w_j) + 1$
$Spec(w_j)$
Else
$Gen(w_j)$
End if
Else if w is of the non-maximal sort,
Select those sorts with the maximum
type feature score
Build up maximal sort
If $cons \sqcup sent \neq \perp$
$p(w_j) = p(w_j) + 1$
$Spec(w_j)$ by 1 layer
End if
End if
End loop
End loop

Figure 4.2: Algorithm description of the lexical acquisition process

sentence.

Compared with finding the global optimum set of the lexicons, local optimum in local context can prevent the brittleness of the lexical information in moving across domains and fit in one of the intuition of the theoretical linguistic work, "natural languages can only work in context or in partial situation".

4.4 Semantic Information

Besides previously described methods in acquiring the lexical information, as described in the previous chapter, the novelty of the current approach is the integration of syntactic and semantic information in the acquisition of lexicalist framework. The next three sections describe our approach in integrating these two types of information together in the acquisition model. In this section, we describe the extraction of the basic semantic representation from the corpus through the three phases.

- Extraction: To gather a set of semantic relation from the corpus.

- Induction: To further expand the current semantic relations to other contexts.

- Generalization: To generalize the learnt relations.

4.4.1 Extraction

The first stage involves extracting the static relations from the corpus. The relation is static as it only contains the local information of the relation, without linking to other entities and relations within the corpus.

We adopt the feature structure notation to describe the triple due to the underlying extracting mechanism from linguistic framework. However, any notation that shares the following mentioned properties can also be used.

Definition 4.3: (Extraction Structure Model)

Given a sentence $sentence_{ijk} \in paragraph_{ij}$,

the relations are extracted with $relation_p$ as shown below:

$$
\begin{aligned}
&<< \\
&\quad relation; \\
&\quad argrole_1 : entity_1; word_1 : words; \\
&\quad argrole_2 : entity_2; word_2 : words; \\
&\quad argrole_3 : entity_3; word_3 : words; \\
&>>
\end{aligned}
$$

In this representation, the *relation* is the word for the lexical realization of the relations. Very often, this *relation* words are verbs and adjectives where they relate one or more entities and relations. The relations contain a number of roles or argument roles as described in linguistic literature or the argument of the predicate logic in first-order logic. Each role contains a lexical realization of the words and the words represent an entity.

The relations are designed with three argument roles as in the current HPSG framework, the lexical entry contains three subcategorization slots.
The relations mentioned are extracted from the parsing result of the HPSG framework. The relation tuple is resided in the SEM content of the feature structure of the parsed sentence. The grammar to be used consists of the basic phrasal and clausal rules and the lexicons which are being acquired.

Example

To clarify the concept, we present a simple extraction example using a simple sentence from the corpus.
Consider the entry in Figure 3.16 which is from the sentence:

"Reading is a way of getting information from something that is written. Reading involves recognising ..."

After processing this sentence with the grammar, the resulting static relations extracted are shown below:

$<< is; argrole_1 : w_{reading}; argrole_2 : t_1, argrole_3 : t_2 >>$
$t_1 :<< get; argrole_1 : w_{information} >>$
$t_2 :<< is; argrole_1 : something; argrole_2 : w_{written} >>$

$<< Recognize; argrole_1 : w_{reading}; argrole_2 : t_3 >>$
$t_3 :<< make; argrole_1 : w_{language} >>$

$<< Get; argrole_1 : w_{hearing}; argrole_2 : w_{information} >>$

$<< include; argrole_1 : t_4; argrole_2 : w_{entertainment} >>$

$t_4 :<< Gain; argrole_1 : w_{information}; argrole_2 : w_{reading} >>$

These extracted relations represent the local relationships between the entities and relations within the sentences. In the extracted relations, there are some parameters such as t_x that represent the case in which a relation fills the role of other relations. These relations are then further used to constrain the space of the syntactical lexical acquisition.

4.4.2 Induction

After extracting the static relations from sentences, the next step is to link different entities and different relations from the extraction phase.
The most intuitive approach is to find all the relations and entities with the same name and linked them together. However, this approach does not work. Like normal texts, the texts within the encyclopedia also contain incoherent relations in which the same set of entities exists in a mutual exclusive set of relations. For example, it is common to have 3 entities named e_1, e_2, e_3 existing in *relation* and the negation of this relation. This is due to the fact that the encyclopedia texts are edited by different authors and at different instance of time and space. Thus, they may not use the completely consistent set of words and concepts to describe things. The link structure of the encyclopedia provides an invaluable evidence to resolve this issue. Instead of taking a global view of relations and entities, we take a partial view of entities and relations in which the relations between two closely linked pages are induced first.

By executing the induction algorithm as shown in Figure 4.3 on every

Algorithm 1: (Induction Phase)

Let $relation = \emptyset$

Given $p_i = <\{paragraph_{ij}\}, \{insense_n\}>$

Let $relation_{self}$ = {Relation Extracted from p_i }

For $\forall insense \in insense_n$,
  For $\forall word_{\alpha\beta\gamma\delta} \in sentence_{\alpha\beta\gamma}$
  and $sentence_{\alpha\beta\gamma} \in paragraph$
  and $paragraph \in insense$
  and $word_{\alpha\beta\gamma\delta} = p_i$
    Generate $relation_{new}$ from $sentence_{\alpha\beta\gamma}$
    If $relation_{new}$ is consistent with $x \in relation_{self}$
      $relation = relation \bigcup relation_{new}$
      set $relation_{new}.argrole_x = p_i$
    end if
  end loop
end loop

Figure 4.3: Algorithm description of the induction phase

document in the encyclopedia texts, each page p_i will contain an extra set: *relation*, containing consistent relations from the other documents to this entry. The step of Induction will be executed based on the extraction mechanism as shown in the previous section. The newly generated relations will be compared against the set of relation induced for the current document by the level coherency. This step is necessary as the texts within different entries may contain inconsistent information and relations. This filtering step is performed to minimize the level of inconsistencies within the set of relations. The appropriate argument role of the newly generated relation will be

set to point to the current entry. The resulting group - *relation*, contains a coherent set of entities and relations as induced from the encyclopedia.

The level of consistency is measured by whether the newly generated relation $relation_{new}$ contradicts with any of the relations in the current entry $relation_{self}$. The current algorithm uses a simple matching method to filter relations. For example, if the current entry contains the text "Reading is mostly done from paper ..." is checked against the incoming entry containing the texts "Reading is not done from paper ...". The incoming relation is filtered out. More sensible methods should be based on how different the entities in the argument role between the $relation_{new}$ and $relation_{self}$ of the same type of relation. In our current framework, we adopt this simple checking method.

The result of this step will be a set of relations, *relation*, with each element containing an appropriate argument role pointing to the current entry.

Example

Following the figure as shown in Figure 3.16 and consider the entry from "school" containing the sentences "For example: writing, reading, and calculating numbers (maths). Many schools also teach arts such as music and art."

The sentence is first translated into the relation tuples, $relation_{new}$ and then matched against the entry "reading". As the entry is consistent, it is added to the *relation*. The appropriate argument role of $relation_{new}$ will point to the entry "reading".

4.4.3 Generalization

The induction phase groups different entities and relations into relationships as modeled by the *relation* set. However, further generalizations on these extracted relations are needed.

The generalization phase, as shown in Figure 4.4 can be thought of deducing the cross-cutting properties in the induced entities and relations. The generalized items act as the basis for further general reading tasks.

Categorization Function:

The role of the categorization function is to deduce how similar two entities are within the possible world of encyclopedia. After executing the generalization phase, each sense will have three new classes containing the information of how they are related to other senses. These relations can be of the three argument roles or in the relation. A value for each tuple is calculated based on the following formula:

$$val = \sum \left(\frac{tuplenum_x}{totalnum_x} \right)$$

where $tuplenum_x$ is the number of occurrence of $role_x$ within a particular sense; and $totalnum_x$ is the number of occurrence of $role_x$ within the corpus.

These values measure how representative a pointing entry is with respect to p_i. A more representative pointing entry means that within the corpus, or within the structure of the world spanned by the corpus, the pointing entry

Algorithm 2: (Generalization Phase)

From the induced group of a particular sense:
$relation_{new}$ of p_i

$role_1 = \emptyset, role_2 = \emptyset, role_3 = \emptyset$
$rel_{set} = \emptyset$

For $\forall r \in relation_{new}$,
if $p_i \in \{argrole_x\}$where $x \in \{1, 2, 3\}$ then
$role_x = role_x \bigcup r$
end if
if $p_i \in \{r\}$ then
if r is not duplicate then
$rel_{set} = rel_{set} \bigcup r$
end if
end if
end loop

For $role_x$ where $x \in \{1, 2, 3\}$
$val = \left(\frac{tuplenum_y}{totalnum_y}\right) + \left(\frac{tuplenum_z}{totalnum_z}\right) + \left(\frac{tuplenum_{rel}}{totalnum_{rel}}\right)$
Define class c_x where c_x contains
the tuple $< R, role_y, role_z, val >$
where $x \neq y$ and $x \neq z$
end loop

For $r \in rel_{set}$
$val = \left(\frac{tuplenum_x}{totalnum_x}\right) + \left(\frac{tuplenum_y}{totalnum_y}\right) + \left(\frac{tuplenum_z}{totalnum_z}\right)$
Define class c_r where c_r contains
the tuple $< role_x, role_y, role_z >$
end loop

Figure 4.4: Algorithm description of the Generalization Model

and p_i are more frequently related with respect to other entries with a less value and will be used in reading more general texts.

4.5 Extension with new text documents

In encountering more general text, in which, unlike encyclopedia, no sense linkage exists. All the information a particular algorithm has is the surface order of the words within the sentences and the paragraph information.
Using the similar strategy of the induction phase, the basic unit of processing in handling the general text is a paragraph and the notation of the paragraphs, sentences and words are defined as below, except the words now have no linkage information. Also, the sense p_i is undefined as the paragraph now contains a lot of different senses, some may be conflicting.

The extension algorithm as shown in Figure 4.5 tries to expand the current encyclopedia knowledge base with the newly processed text. Theoretically, by processing more texts, the base will become larger and can cover more relations and entities.

4.6 Integrating the syntactic and semantic acquisition framework

In contrast with the previous attempt, the major novelty of this approach is to exploit the cross-cutting semantic information across the semantically and contextually related documents for the acquisition. The addition of the semantic information is to further improve the confidence of the semantic

Algorithm 3: (Extension with new texts)

Assume $paragraph_a$ is being processed:
For $\forall sentence_{ab} \in paragraph_a$
    Extract the set of $\{relation\}$ from $sentence_{ab}$
    For $r \in relation$
        For $\forall$ known entities within r,
            Check whether r and
            the relations in the *relation* collected in generalization
            phase are consistent.

            If relations are consistent,
            add this relation to the appropriate $relation_{set}$
        end loop

        For unknown entities,
            Form new relation $relation_{add}$ containing these entities.
            If $relation_{add}$ contains some entities
            existing in the corpus
                Build up a new link from this relation
                to the target entities.
            end if
        end loop

    end loop
end loop

Figure 4.5: Algorithm description of the Extension Phase

role and in particular, in acquiring the subcategorization information. The

Algorithm 4: (Integrating syntactic and semantic information in lexical acquisition)

Consider a sentence $sent_i$ from document d in the corpus
And assume *relation* is the set of relations in d
after the expansion from the three phases.
For $w_j \in sent_i$,
If w_j contains subcategorization in the current lexical types
Match the w_j with the *relation* for matched relations
If match is found
Increment the confidence score of the entries by the number of matched relations.
End if
Match the number of roles in w_j with the relation in *relation*
Decrease the confidence score if the number of role does not matches
Increase the score if the number of role matches
End if

Figure 4.6: Algorithm description of integrating syntactic and semantic information in lexical acquisition

routine exploits two types of information to further increase the confidence score of the current type for a particular lexical entries: The number of roles and the filler of the roles. Since these types of information are extracted from the document network as occurred in the corpus and they represent high-level inter-relationships between entities and relations. Exploiting this information makes the current acquisition model to capture the high level semantic information and be integrated with the SEM semantic features of the HPSG framework.

Chapter 5

Evaluation

We evaluate our approach based on the accuracy of the acquisition process. We measure the accuracy of the lexical entry learnt based on type precision and type recall against the hand-built lexicon from the English Resource Grammar.

5.1 Evaluation Metric - English Resource Grammar

Unlike other shallow lexical acquisition tasks, few resources are available for the direct evaluation of the grammar and lexicon learnt from deep lexicalist framework such as GPSG, LFG, HPSG, and many others. This introduces additional difficulties in evaluating the performance of a particular acquisition algorithm for deep lexicalist framework. However, recent work in manually building a HPSG grammar from scratch provides an invaluable resource for the evaluation.

5.1.1 English Resource Grammar

The introduction of the VerbMobil project initiates the need of the construction of a large scale broad coverage deep lexicalist framework for the design of a spoken machine translation system where the HPSG forms the backbone of all the language processing routines. The grammar generated - English Resource Grammar (ERG) is now available freely [1]. Other projects from different research groups are extending the efforts to other languages such as Spanish, French, Japanese, German and many others.

The experiments in this thesis are targeted at the English Resource Grammar. The ERG is a precise broad coverage grammar in HPSG. The grammar version used in our experiments consists of 12,000 lexical items and when combined with the lexical rules, compile to 25,000 distinct word forms. For the construction rules, the grammar now contains 85 construction rules. The tables shown in Figure 5.1 depict some examples of the base rules used in the grammar. The grammar, no matter it is the rules or the lexicons, are inherited from the basic grammar rules where some of these examples are shown in Figure 5.1. The construction rules, which are used in parsing the sentences, are created from the syntax table. This syntax table represents some of the core rules in the grammatical theory such as headed phrase in Figure 5.2, non-head phrase, unary and binary phrase in Figure 5.3 and in Figure 5.4, clause in Figure 5.5, head complement phrase in Figure 5.6.

Finally, the lexicons are built from the basic rule, lexical rules and the final lexicon entries. Figure 5.7 shows some of the examples.

[1] This resource can be obtained freely from http://lingo.stanford.edu

```
sign := sign-min AND
( SYNSEM synsem
( LOCAL.CONT.HOOK.INDEX @index,
-SIND @index ),
ARGS *list*,
INFLECTD bool,
GENRE genre,
DIALECT dialect,
POSSCL bool,
IDIOM bool,
STEM *list* ).

phrase-or-lexrule := sign AND
( SYNSEM canonical-synsem
( LOCAL.CONT.HOOK *hook,
PHON.ONSET *onset ),
C-CONT mrs-min ( HOOK *hook ),
ARGS.FIRST.SYNSEM.PHON.ONSET *onset,
RNAME string ).

word-or-lexrule-min := sign-min.

word-or-lexrule := word-or-lexrule-min AND sign
( ALTS alts-min ).

word := word-or-lexrule
( POSSCL -,
SYNSEM.PUNCT.PNCTPR ppair ).
```

Figure 5.1: English Resource Grammar Basic Rule

Word class	Lexical Types	Lexical Items
Noun	28	7100
Verb	40	1400
Adjective	21	1300
Adverb	25	700

Table 5.1: English Resource Grammar - Lexical type distribution

```
headed-phrase := phrase AND
( SYNSEM.LOCAL ( CAT ( HEAD head AND @head,
HC-LEX @hclex ),
AGR @agr,
CONT.MSG.PSV @psv,
CONJ @conj ),
HD-DTR.SYNSEM.LOCAL local AND
( CAT ( HEAD @head,
HC-LEX @hclex ),
AGR @agr,
CONT.MSG.PSV @psv,
CONJ @conj ) ).
```

Figure 5.2: English Resource Grammar Construction Rule - Head Phrase

From the set of the total of 25,000 distinct word forms, we select a set of 10,500 distinct word forms for the final evaluation where the selected word forms are of higher occurrence in the Wikipedia corpus.
The breakdown of the different types of word class in the grammar is shown in Figure 5.1.

5.2 Experiments

5.2.1 Tasks

Two tasks are designed to evaluate the proposed algorithms in acquiring the closed-class and open-class words.

Baseline

Since we are evaluating the accuracies of applying our algorithms to attach lexical class label to the candidate words, the baseline model is built by

```
basic-unary-phrase := phrase AND
( SYNSEM.LOCAL.CONT ( RELS *diff-list* AND
( LIST @first,
LAST @last ),
HCONS *diff-list* AND
( LIST @scfirst,
LAST @sclast ) ),
C-CONT ( RELS ( LIST @first,
LAST @middle ),
HCONS ( LIST @scfirst,
LAST @scmiddle ) ),
ARGS < sign AND ( SYNSEM.LOCAL local AND
( CONT ( RELS *diff-list* AND
( LIST @middle,
LAST @last ),
HCONS *diff-list* AND
( LIST @scmiddle,
LAST @sclast ) ) ),
IDIOM @idiom,
DIALECT @dialect,
GENRE @genre ) >,
IDIOM @idiom,
DIALECT @dialect,
GENRE @genre ).
```

Figure 5.3: English Resource Grammar Construction Rule - Unary phrase

```
basic-binary-phrase := phrase AND
( SYNSEM ( LOCAL.CONT ( RELS *diff-list* AND
( LIST @first,
LAST @last ),
HCONS *diff-list* AND
( LIST @scfirst,
LAST @sclast ) ) ),
C-CONT ( RELS *diff-list* AND
( LIST @first,
LAST @middle1 ),
HCONS *diff-list* AND
( LIST @scfirst,
LAST @scmiddle1 ) ),
ARGS < sign AND ( SYNSEM ( LOCAL local AND
( CONT ( RELS *diff-list* AND
( LIST @middle1,
LAST @middle2 ),
HCONS *diff-list* AND
( LIST @scmiddle1,
LAST @scmiddle2 ) ) ),
POSSCL -,
IDIOM @idiom,
DIALECT @dialect,
GENRE @genre ),
sign AND ( SYNSEM ( LOCAL local AND
( CONT ( RELS *diff-list* AND
( LIST @middle2,
LAST @last ),
HCONS *diff-list* AND
( LIST @scmiddle2,
LAST @sclast ) ) ) ),
POSSCL @posscl,
IDIOM @idiom,
DIALECT @dialect,
GENRE @genre ) >,
POSSCL @posscl,
IDIOM @idiom,
DIALECT @dialect,
GENRE @genre ).
```


Figure 5.4: English Resource Grammar Construction Rule - Binary Phrase

```
clause := phrasal AND
( SYNSEM.LOCAL ( CAT ( HEAD v-or-g-or-dadv,
VAL.COMPS < > ),
CONJ cnil ) ).
```

Figure 5.5: English Resource Grammar Construction Rule - Clause

randomly selecting the lexical types from the whole label type set and attach these generated lexical types to the candidate words for evaluation. The average accuracy of the generated lexicon items on open-class word, including all types, is 0.364.

Closed-class word

The first task is to evaluate the performance of our method in acquiring the closed-class words such as pronoun, auxiliary verb. This is not a common learning task as the previous literature and the current consensus usually treats these words as language constants which do not vary from time to time. However, this is challenging as many of these words, so-called "stop words" from the information retrieval community, are some of the most commonly occurring words. These words do function in the construction of linguistic statements. Investigating the learning process of these words can help us to investigate how the lexical properties change or evolve. As common texts usually contain too many of these closed-class words and thus make them difficult to acquire, we test our approach on some of the simple English corpus as in a simple version of Wikipedia.

```
basic-complementizer-rule := binary-phrase AND non-headed-phrase AND
binary-rule-left-to-right AND
( SYNSEM ( LOCAL ( CAT ( HEAD comp AND( VFORM @vform, AUX @aux, INV @inv,
PRD @prd, TAM @tam, CASE @case,
POSS -, KEYS @keys, MOD @mod ),
VAL ( SUBJ @subj, SPR @spr, SPEC @spec,
COMPS @comps ),
MC @mc,
POSTHD @ph ),
CONT.MSG @msg,
CONJ @conj,
AGR @agr ),
NONLOC @nonloc,
PUNCT.PNCTPR @pnctpr ),
ARGS < sign AND
( SYNSEM ( LOCAL ( CAT ( HEAD lexcomp AND
( VFORM @vform, AUX @aux, INV @inv,
PRD @prd, TAM @tam, CASE @case,
KEYS @keys, MOD @mod ),
VAL ( SUBJ @subj, SPR @spr, SPEC @spec,
COMPS < @comp1 . @comps > ),
MC @mc,
POSTHD @ph ),
CONT ( HOOK @hook,
MSG @msg ),
CONJ @conj AND cnil,
AGR @agr ),
NONLOC @nonloc,
PUNCT ( RPUNCT comma-or-rbc-or-pair-or-no-punct,
PNCTPR ppair ) ) ),
sign AND ( SYNSEM @comp1 AND ( PUNCT.PNCTPR @pnctpr ) ) >,
C-CONT ( HOOK @hook,
RELS.LIST < message AND @msg, ... >,
HCONS <! !> ) ).
```

Figure 5.6: English Resource Grammar Construction Rule - Complementizer

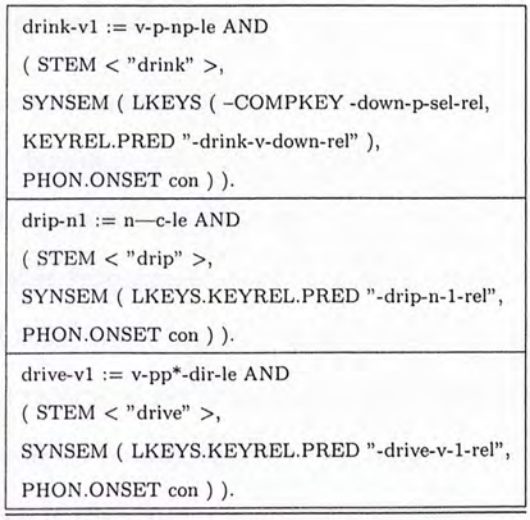


Figure 5.7: English Resource Grammar Construction Rule - Lexicon

Open-class word

The second task is to evaluate the algorithm in acquiring new words. In this task, the closed-class words and some of the open-class words are selected as "training" lexicon examples for the acquisition.

Since the English Resource Grammar cannot cover every word in the corpus we use in acquisition, only words existed in the Grammar are evaluated in the current experiments. For the 10,500 words in the grammar, one-tenth of the 10,500 words are selected for targets in acquisition while the remaining words are assumed to be constant. Thus, the algorithm is evaluated by the performance in acquiring the around 1,000 new words which do not exist in the grammar initially.

5.2.2 Evaluation Measures

Special consideration is made in designing the evaluation metric. First, comparing with other language tasks such as part-of-speech or pure subcategorization where information is not cascaded in the form of hierarchy, the design of the deep lexicalist framework uses the inheritance in every construct of the grammar and thus, evaluating the learnt lexicon which focuses on precise accuracies such as recall and precision is not appropriate for the current task. Second, as in other deep lexicalist frameworks, the information is spread from the root type, down to the lexical type or construction types where the upper class contains less specific information. As the upper class contains the more general information and the process of generalizing and specializing is one of the features employed by the proposed method to tackle the problem of lexical acquisition. It is appropriate to evaluate the accuracies of the upper class information as well so as to evaluate the performance of the generalization and specialization process.

Thus, two measures are used to evaluate the accuracies of the lexicons.

Type precision, type recall, type F-score

A number of factors are investigated in the lexical acquisition task. These include the accuracy of the items being learnt and the convergence rate of the learning of the lexical items.

We follow the precision definition from [1] as follows:

- **type precision**: The proportion of correct hypothesized lexical entries.
- **type recall**: The proportion of gold-standard lexical entries.

- **type F-score**: The harmonic mean of the type precision and type recall.

L-x type precision, L-x type recall, L-x type F-score

However, we would also be interested in how the type of the lexical item is being fixed in the type hierarchy. This is an interesting phenomena to be observed as this can get insight in how the type of the lexical items are gradually formed based on the corpus. To facilitate the evaluation process, we extend the type accuracy into super-type accuracy as follows:

- ***L-x* type precision**: The supertype precision at the xth supertypes.
- ***L-x* type recall**: The supertype recall at the xth supertypes.
- ***L-x* type F-score**: The harmonic mean of the precision and recall at the xth supertype.

Finally, it is interesting to observe how many iterations are needed before the lexical items remain constant at a particular lexical types. We define the convergence rate as follows:

Definition: Convergence rate

The number of successful parses involving the words before the words remain in the maximal type and the L-1 supertype.

5.2.3 Methodologies

To evaluate the effectiveness of the model, a baseline model is built by generating lexical type for each candidate word randomly. As our method focuses

on evaluating the capabilities to acquire the lexicons from the corpus with full contextual information such as encyclopedia, the random lexical types would reveal how the algorithm would perform in acquiring from random langauge data and the language data in context.

5.2.4 Corpus Preparation

Closed-class word

The experiment on the closed-class word is performed on the simple version of the Wikipedia corpus.[2] This corpus contains 201,101 distinct terms and the corpus contains 3,230,796 terms, spreading in about 13,000 documents. Some of the terms with higher occurrence are shown in Figure 5.2.

Open-class word

The experiment on the open-class words is performed on the corpus by varying the size from 10,000 to 50,000. Of the english version of encyclopedia, there are around 1.3 million documents with the lexicon size of around 1.2 billion words in total. In the current experiments, we select a subset of the documents for evaluation.

Instead of randomly selecting the documents from the corpus, the subset of documents are selected based on the link information. First, a document is selected from the corpus. This document should be rich in links. Some of the targeted documents are those in the portal. Second, we trace the linked documents and include them in the subset. This process is repeated in the

[2]This corpus can be obtained from http://download.wikimedia.org

Term	Term count
is	39,345
the	99,494
of	71,555
with	14,999
that	13,052
to	31,777
in	39,477
and	46,966
a	40,295
for	11,566
it	11,790
or	10227
are	14,638
was	12,444
by	10,022

Table 5.2: Simple Wikipedia - Commonly occurring words

outbound link of the subset document until the document size reaches a particular limit for the experiment.

As our evaluation is focused on the contextual information, as stored in the link in the Wikipedia structure, we believe that the contextual documents prepared by the described document preparation method should be more appropriate with the whole corpus.

5.2.5 Results

Closed-class word

The list of closed-class word to be evaluated is listed in Table 5.3. The acquisition accuracy of these words is shown in Table 5.4. The average accuracy of the acquisition of closed-class word is 0.321. The evaluation measure used is the one mentioned in the Section 5.2.2.

Open-class word

The result for type F-score for all the different major lexical classes (noun, verb, adjective and adverb) are shown in Table 5.8. The proposed learning methods attain a F-score of 0.521. Figures 5.9, 5.10, 5.11 and 5.12 show the type F-score of the acquisition of other major lexical classes including noun, verb, adjective, and adverb. Of these findings, the adjective class is relatively easier to be induced from the corpus, with a type F-score of 0.735 while the verb group is harder to learn.

Comparing with part-of-speech tag, the lexicalist framework such as HPSG uses a hierarchy of lexical types to model the cross-cutting properties of the

Word	Word count	Word	Word count	Word	Word count	Word	Word count
is	28476	the	73262	of	51803	with	13113
The	14407	that	10058	to	23951	in	28739
and	35751	a	30478	are	10957	was	9747
for	8421	or	7852	as	7013	by	7287
from	6241	on	6514	it	6347	not	5292
be	5205	have	4269	they	4319	It	4668
In	4337	can	4233	an	4806	also	4487
he	3117	at	3496	part	3294	which	3142
has	3461	who	3225	his	3064	other	3264
one	3429	was	3081	so	1315	if	1245
where	1306	will	1377	after	1530	usually	1037
he	3117	at	3496				

Table 5.3: Closed-class word list

Run	Accuracy
Run 1	0.283
Run 2	0.370
Run 3	0.391
Run 4	0.239
Average	0.321

Table 5.4: Closed-class word list

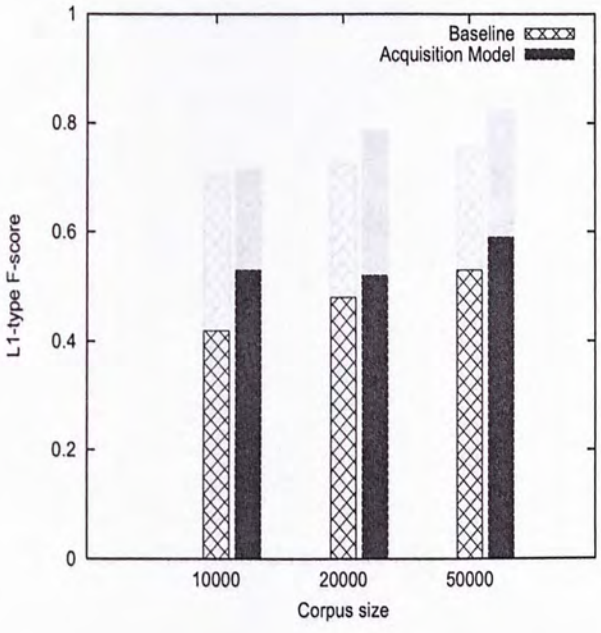


Figure 5.8: Acquisition Result for all types

lexical items. This hierarchy of types represent the variation in the amount of linguistic information stored in a type, with the deeper type containing more specific information. As the proposed algorithm makes use of the relationships between different types in the hierarchy, it is valuable to observe how the information is accumulated in the acquisition process and how the inconsistent information is gradually filtered out.

We evaluate our method by calculating the L-1 **type F-score** and L-2 **type F-score**. Figure 5.8 shows the overall performance of the L-1 type F-score. As the type is less specific, the accuracy is higher than the type F-score. Comparing the L-1 type F-score and the type F-score, the difference between the two sets of values is not significant, indicating that as the type becomes more specific, not much information is added that is significant enough to change

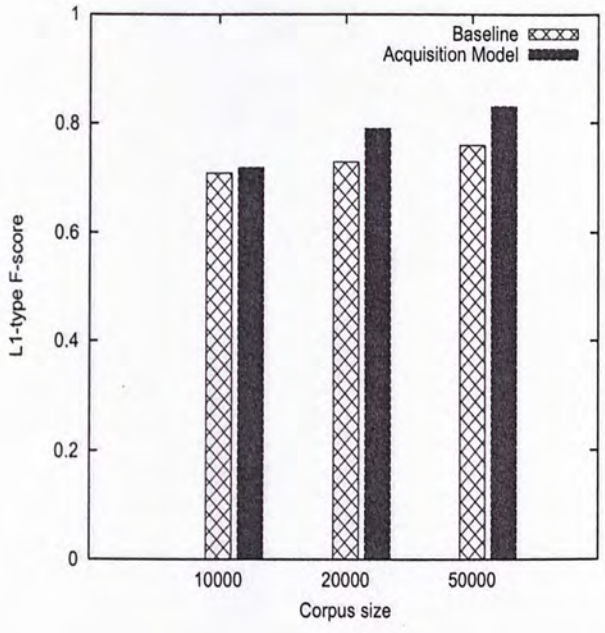


Figure 5.9: Acquisition Result for Adjective

the child type. We suspect that this may relate to the original design of the lexical type hierarchy and the "amount" of information embedded in the different types occurring in the hierarchy. Table 5.5 shows the experimental results of the acquisition model from two randomly selected documents from the Wikipedia. Table 5.6, 5.9, 5.12 and 5.15 show the acquisition results of the different type of lexical class (Adjectives, Noun, Adverb, Verb) on different size of the corpora (10000,20000,50000). Table 5.7, 5.10, 5.13, 5.16 show the precision of the respective experiments while Table 5.8, 5.11, 5.14, 5.17 show the recall of the respective experiments. Each of the experiment is obtained by randomly selecting an article from the Wikipedia, tracing all the links originated from the selected articles to build a corpus for acquisition. Since the generated corpus size always produces a large corpus that are

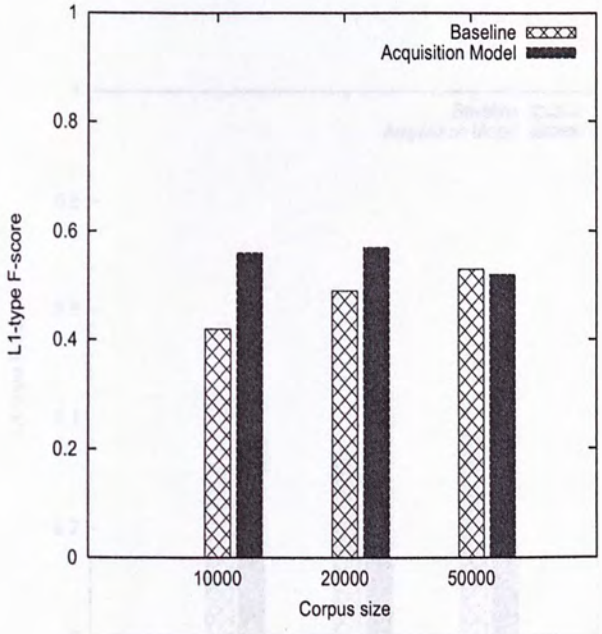


Figure 5.10: Acquisition Result for Noun

computationally intractable, we further restrict the corpus size by randomly picking a specific portion of the links to expand the corpus and a number of runs (Run 1 to 4) are generated. The figure represents the results of L-1 type F-score.

5.3 Result Analysis

There are a number of trends that can be discovered from the results. First, the proposed algorithm shows a steady trend of increasing performance as the corpus size increases. As the size of the corpus increases, the word occurrence of the selected lexical items would be higher and thus, it is expected

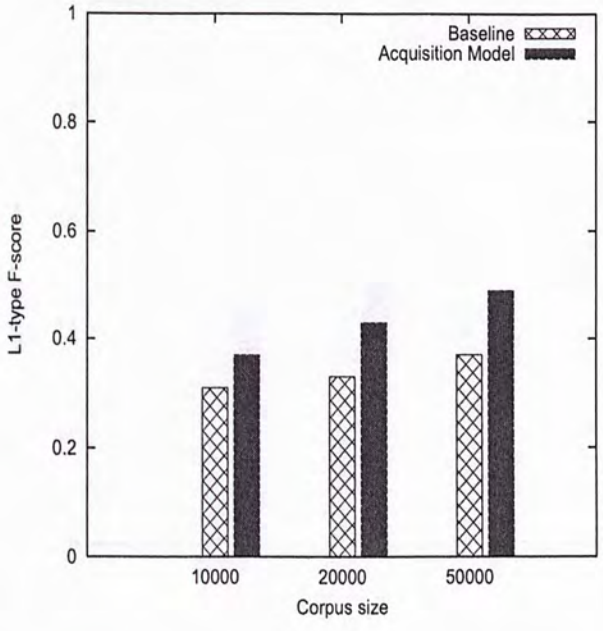


Figure 5.11: Acquisition Result for Adverb

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.488	0.498	0.537	0.659	0.709	0.747	0.514	0.509	0.534	0.379	0.364	0.39	0.394	0.408	0.425
Run 2	0.490	0.500	0.5308	0.620	0.625	0.658	0.484	0.508	0.509	0.37	0.406	0.419	0.448	0.456	0.495
Run 3	0.487	0.492	0.505	0.681	0.675	0.697	0.464	0.475	0.495	0.389	0.389	0.404	0.474	0.488	0.508
Run 4	0.502	0.503	0.512	0.641	0.645	0.667	0.426	0.443	0.471	0.391	0.401	0.416	0.465	0.474	0.477
Average	0.492	0.498	0.521	0.650	0.664	0.692	0.472	0.484	0.502	0.382	0.390	0.408	0.445	0.457	0.476

Table 5.5: Acquisition result for 4 data sets

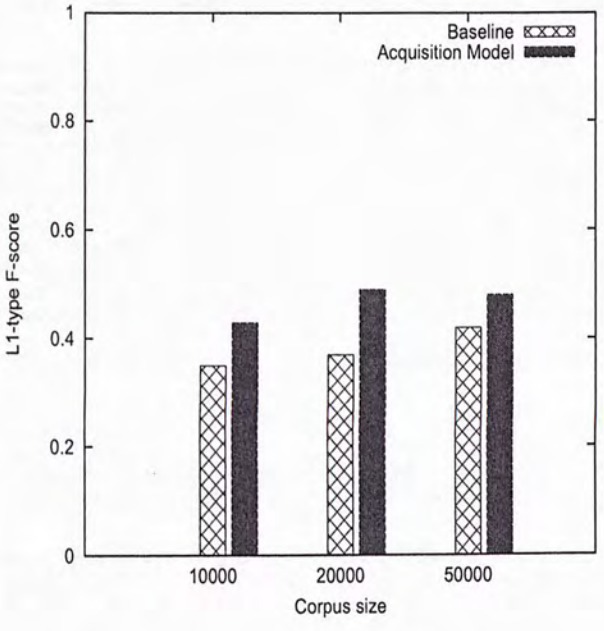


Figure 5.12: Acquisition Result for Verb

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.532	0.526	0.598	0.643	0.766	0.788	0.564	0.477	0.511	0.411	0.389	0.422	0.491	0.423	0.456
Run 2	0.512	0.534	0.577	0.562	0.634	0.655	0.512	0.545	0.557	0.342	0.412	0.433	0.433	0.488	0.499
Run 3	0.488	0.51	0.563	0.657	0.688	0.733	0.412	0.467	0.49	0.41	0.32	0.398	0.39	0.423	0.437
Run 4	0.501	0.512	0.523	0.71	0.723	0.789	0.543	0.544	0.533	0.389	0.39	0.388	0.399	0.389	0.423
Run 5	0.405	0.41	0.423	0.722	0.734	0.77	0.539	0.512	0.578	0.344	0.311	0.309	0.256	0.319	0.309
Average	0.488	0.498	0.537	0.659	0.709	0.747	0.514	0.509	0.534	0.379	0.364	0.39	0.394	0.409	0.425

Table 5.6: Acquisition result for the document - Jose Reis

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.642	0.741	0.566	0.658	0.746	0.674	0.649	0.722	0.472	0.697	0.595	0.485	0.603	0.654	0.748
Run 2	0.561	0.546	0.582	0.688	0.749	0.493	0.592	0.674	0.595	0.615	0.586	0.739	0.507	0.577	0.555
Run 3	0.715	0.572	0.726	0.61	0.579	0.607	0.533	0.478	0.578	0.556	0.725	0.723	0.59	0.579	0.627
Run 4	0.589	0.689	0.724	0.721	0.627	0.723	0.464	0.469	0.647	0.609	0.634	0.482	0.598	0.692	0.609
Run 5	0.703	0.657	0.731	0.679	0.599	0.665	0.705	0.564	0.638	0.546	0.692	0.515	0.684	0.632	0.489
Average	0.642	0.641	0.666	0.671	0.66	0.632	0.589	0.581	0.586	0.605	0.647	0.589	0.597	0.627	0.606

Table 5.7: Acquisition result for the document (Precision) - Jose Reis

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.454	0.408	0.634	0.629	0.787	0.949	0.499	0.356	0.556	0.291	0.289	0.374	0.414	0.313	0.328
Run 2	0.471	0.522	0.572	0.475	0.55	0.975	0.451	0.458	0.524	0.237	0.318	0.306	0.378	0.423	0.453
Run 3	0.37	0.46	0.46	0.712	0.848	0.925	0.336	0.456	0.425	0.325	0.205	0.275	0.291	0.333	0.335
Run 4	0.436	0.407	0.409	0.7	0.853	0.868	0.654	0.648	0.453	0.286	0.282	0.325	0.299	0.271	0.324
Run 5	0.284	0.298	0.298	0.77	0.946	0.915	0.436	0.469	0.528	0.251	0.201	0.221	0.157	0.213	0.226
Average	0.403	0.419	0.475	0.657	0.797	0.926	0.475	0.477	0.497	0.278	0.259	0.3	0.308	0.311	0.333

Table 5.8: Acquisition result for the document (Recall) - Jose Reis

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.506	0.512	0.544	0.655	0.689	0.712	0.453	0.489	0.491	0.341	0.441	0.432	0.488	0.433	0.491
Run 2	0.412	0.422	0.432	0.544	0.59	0.589	0.433	0.456	0.412	0.312	0.344	0.378	0.41	0.476	0.501
Run 3	0.499	0.501	0.519	0.633	0.672	0.691	0.51	0.498	0.523	0.386	0.398	0.388	0.478	0.488	0.493
Run 4	0.517	0.533	0.581	0.612	0.544	0.623	0.509	0.529	0.531	0.412	0.423	0.444	0.412	0.411	0.499
Run 5	0.514	0.533	0.573	0.655	0.63	0.674	0.513	0.567	0.589	0.399	0.423	0.455	0.452	0.472	0.491
Average	0.490	0.500	0.530	0.620	0.625	0.658	0.484	0.508	0.509	0.37	0.406	0.420	0.448	0.456	0.495

Table 5.9: Acquisition result for the document - Woodlands MRT Station

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.677	0.734	0.462	0.617	0.605	0.662	0.457	0.729	0.461	0.611	0.475	0.68	0.718	0.717	0.496
Run 2	0.495	0.69	0.622	0.577	0.516	0.572	0.525	0.648	0.707	0.643	0.58	0.688	0.492	0.632	0.509
Run 3	0.7	0.559	0.493	0.712	0.726	0.647	0.624	0.733	0.626	0.635	0.593	0.651	0.565	0.562	0.618
Run 4	0.611	0.607	0.558	0.483	0.734	0.624	0.605	0.509	0.522	0.562	0.702	0.652	0.5	0.744	0.534
Run 5	0.558	0.694	0.643	0.601	0.656	0.619	0.498	0.53	0.601	0.674	0.715	0.745	0.575	0.53	0.557
Average	0.608	0.657	0.556	0.598	0.647	0.625	0.542	0.63	0.583	0.625	0.613	0.683	0.57	0.637	0.543

Table 5.10: Acquisition result for the document (Precision) - Woodlands MRT Station

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.404	0.393	0.661	0.698	0.801	0.77	0.449	0.368	0.526	0.237	0.411	0.317	0.37	0.31	0.486
Run 2	0.353	0.304	0.331	0.515	0.689	0.607	0.368	0.352	0.291	0.206	0.244	0.261	0.351	0.382	0.494
Run 3	0.388	0.454	0.548	0.57	0.625	0.741	0.431	0.377	0.449	0.277	0.299	0.276	0.414	0.431	0.41
Run 4	0.448	0.475	0.606	0.835	0.432	0.622	0.439	0.55	0.541	0.325	0.303	0.337	0.351	0.284	0.468
Run 5	0.476	0.433	0.517	0.72	0.606	0.74	0.529	0.609	0.577	0.283	0.3	0.328	0.372	0.425	0.439
Average	0.414	0.412	0.533	0.667	0.631	0.696	0.443	0.451	0.477	0.266	0.312	0.304	0.372	0.366	0.459

Table 5.11: Acquisition result for the document (Recall) - Woodlands MRT Station

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.491	0.493	0.51	0.691	0.643	0.712	0.467	0.478	0.51	0.412	0.415	0.432	0.509	0.498	0.512
Run 2	0.483	0.489	0.498	0.674	0.688	0.687	0.501	0.5	0.517	0.391	0.387	0.415	0.508	0.509	0.521
Run 3	0.487	0.483	0.504	0.697	0.683	0.701	0.412	0.431	0.462	0.377	0.365	0.398	0.451	0.489	0.534
Run 4	0.502	0.517	0.528	0.682	0.691	0.699	0.464	0.481	0.495	0.398	0.391	0.387	0.489	0.491	0.503
Run 5	0.472	0.477	0.484	0.663	0.672	0.685	0.478	0.483	0.491	0.367	0.388	0.39	0.412	0.452	0.469
Average	0.487	0.492	0.505	0.681	0.675	0.697	0.464	0.475	0.495	0.389	0.389	0.404	0.474	0.488	0.508

Table 5.12: Acquisition result for the document - Thermopylae

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.618	0.658	0.7	0.632	0.531	0.653	0.61	0.641	0.532	0.728	0.538	0.638	0.467	0.675	0.715
Run 2	0.632	0.553	0.715	0.638	0.683	0.657	0.739	0.573	0.671	0.688	0.693	0.513	0.556	0.601	0.463
Run 3	0.738	0.682	0.496	0.743	0.584	0.656	0.634	0.666	0.634	0.722	0.554	0.651	0.646	0.519	0.534
Run 4	0.75	0.484	0.722	0.718	0.749	0.707	0.707	0.572	0.628	0.644	0.516	0.692	0.45	0.666	0.705
Run 5	0.738	0.598	0.451	0.732	0.733	0.657	0.616	0.649	0.541	0.588	0.452	0.743	0.484	0.521	0.526
Average	0.695	0.595	0.617	0.693	0.656	0.666	0.661	0.62	0.601	0.674	0.551	0.647	0.521	0.596	0.589

Table 5.13: Acquisition result for the document (Precision) - Termopylae

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.407	0.394	0.401	0.762	0.815	0.783	0.378	0.381	0.49	0.287	0.338	0.327	0.559	0.395	0.399
Run 2	0.391	0.438	0.382	0.715	0.693	0.72	0.379	0.443	0.42	0.273	0.268	0.348	0.468	0.442	0.595
Run 3	0.363	0.374	0.512	0.656	0.821	0.752	0.305	0.319	0.363	0.255	0.272	0.287	0.346	0.462	0.534
Run 4	0.377	0.554	0.416	0.65	0.641	0.691	0.345	0.415	0.408	0.288	0.315	0.269	0.535	0.389	0.391
Run 5	0.347	0.397	0.523	0.606	0.621	0.716	0.391	0.385	0.449	0.267	0.34	0.264	0.359	0.399	0.423
Average	0.377	0.431	0.447	0.678	0.718	0.732	0.36	0.389	0.426	0.274	0.307	0.299	0.453	0.417	0.468

Table 5.14: Acquisition result for the document (Recall) - Termopylae

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.516	0.523	0.545	0.674	0.688	0.691	0.451	0.478	0.491	0.381	0.366	0.385	0.491	0.499	0.484
Run 2	0.491	0.488	0.496	0.641	0.634	0.649	0.466	0.452	0.478	0.391	0.402	0.415	0.486	0.495	0.49
Run 3	0.486	0.488	0.496	0.593	0.599	0.643	0.327	0.387	0.416	0.401	0.423	0.417	0.433	0.438	0.462
Run 4	0.503	0.497	0.517	0.634	0.652	0.675	0.431	0.398	0.461	0.387	0.399	0.413	0.455	0.467	0.471
Run 5	0.516	0.518	0.508	0.664	0.651	0.679	0.453	0.501	0.509	0.395	0.416	0.452	0.461	0.473	0.478
Average	0.502	0.503	0.512	0.641	0.645	0.667	0.426	0.443	0.471	0.391	0.401	0.416	0.465	0.474	0.477

Table 5.15: Acquisition result for the document - Petrous portion of the internal carotid artery

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.573	0.747	0.501	0.566	0.66	0.608	0.452	0.629	0.457	0.51	0.531	0.648	0.736	0.722	0.627
Run 2	0.468	0.503	0.559	0.68	0.482	0.605	0.476	0.663	0.633	0.473	0.698	0.514	0.534	0.529	0.518
Run 3	0.588	0.652	0.515	0.639	0.468	0.725	0.497	0.47	0.604	0.504	0.53	0.685	0.702	0.517	0.657
Run 4	0.579	0.534	0.71	0.688	0.743	0.543	0.491	0.656	0.726	0.514	0.603	0.49	0.599	0.683	0.557
Run 5	0.737	0.585	0.622	0.626	0.603	0.597	0.674	0.623	0.451	0.728	0.703	0.686	0.68	0.47	0.593
Average	0.589	0.604	0.582	0.64	0.591	0.615	0.518	0.608	0.574	0.546	0.613	0.604	0.65	0.584	0.591

Table 5.16: Acquisition result for the document (Precision) - Petrous portion of the internal carotid artery

		ALL			Adj			Noun			Adv			Verb	
Corpus size	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000	10000	20000	50000
Run 1	0.47	0.402	0.597	0.833	0.719	0.801	0.45	0.385	0.531	0.304	0.279	0.274	0.368	0.381	0.394
Run 2	0.517	0.474	0.446	0.606	0.925	0.7	0.456	0.343	0.384	0.333	0.282	0.348	0.446	0.465	0.465
Run 3	0.414	0.39	0.478	0.553	0.831	0.578	0.244	0.329	0.317	0.333	0.352	0.3	0.313	0.38	0.356
Run 4	0.445	0.465	0.406	0.588	0.581	0.892	0.384	0.286	0.338	0.31	0.298	0.357	0.367	0.355	0.408
Run 5	0.397	0.465	0.429	0.706	0.707	0.787	0.341	0.419	0.584	0.271	0.295	0.337	0.349	0.476	0.4
Average	0.448	0.439	0.471	0.657	0.753	0.751	0.375	0.352	0.431	0.31	0.301	0.323	0.369	0.411	0.405

Table 5.17: Acquisition result for the document (Recall) - Petrous portion of the internal carotid artery

that the final learnt lexicon should converge and should approach the ultimate lexicon.

It can be discovered that the general trend of the acquisition accuracy increases steadily as the corpus size increases from 10000 to 50000 articles. There are a number of reasons contributing to this trend. First, as the corpora sizes increase, there is the more likelihood for a word to exist more times throughout the corpus, and thus more chances for the acquisition algorithm to generalize from these increasing number instances of word throughout the corpus. As in the algorithm, the more times the lexical information can be appended to the lexicon during the acquisition, the more likely it can be optimized to obtain a representation that can be fit into larger amount of data. Second, as the corpus is generated by following the different links to other articles where the articles are in the similar context with the sentences and words in the original articles. Thus, the increase in corpus size can provide more instance of the occurrence of the vocabulary inside the context. As

our algorithm works on both syntax and semantics side, the increase in the instances of words within context can help to further improve the accuracy of the semantic features which contributes to the overall accuracy.

As the corpus we use in the acquisition experiment is from the encyclopedia where the major content and the semantic relations are describing entities and relations, which are basically linguistically realized as nouns and verbs, and thus, we can discover better performance for the acquisition of nouns and verbs and these type of lexical items appear more times in the corpus and they are related to each others in a structural way in the encyclopedia. The adjective class shows the least improvement in the result. This is not surprising as the adjective exhibits more abstract grammatical properties from noun and verb, where a noun can be classified by case roles, gender, and many other features where a verb can be classified by subcategorization roles, case roles. In addition, the adjectives are some form of second-class citizen in the encyclopedia corpus and thus, the acquisition strategy proposed here would be adapted for more accurate adjective acquisition.

For the adverb class, it shows a steady trend in increasing performance. However, more investigations are needed for better representation of the result in adverb as this class of lexical item constitutes a smaller proportion of lexical items in the grammar.

As normal in the acquisition experiments, nouns and verbs are easier to acquire as they represent predominately materialized entities and relations while adjectives, adverbs and other function words are harder to acquire as they concern about abstract properties and relations of words. More interesting is that the nature of the encyclopedia documents is about fact, entities,

relations and it is easy to discover that they can be acquired better and attain a more stable performance comparing with adjective and adverb. However, we do think that working on noun and verb is the right step in designing the acquisition experiment as the adjectives and adverbs can be seen as by-product of other word class where the nature of adjectives and adverbs are generalized from different instances of entities and relations.

Considering the different runs for each experiments, it can be found that the performance can vary greatly by 0.1. In generating different runs, we randomly select a specific portion of documents to do the experiments. The accuracy depends greatly on which portion is selected. If the portion is more relevant to the original article, there is a high chance that the additional existence of lexical item will correlate better with the original article. On the contrary, the portion would be irrelevant and inconsistency may exist between the lexical item on different pages. It suggests that the composition of the background corpus can affect the acuqisition accuracy given a relatively small size of corpus. However, if the number of article increases, it can be foreseen that as the context is larger, the accuracy flunctuation would be less. This verifies the fact that language is context-dependent.

In extracting the corpora to generate different runs, we extract only a specific portion of texts for acquisition. The major motivation is originally to restrict the corpus size to make the task computational feasible. We simply select a subset of the links for the task. The difference in the different runs reflects the how the further extracted texts are related to the original article and thus it indirectly represents the coherence of the word structure in different context.

Chapter 6

Conclusions

Deep lexicalist framework provides a more structural way to represent and model natural languages. Through the investigation of the type system and the uses of extra semantic information, the lexicon acquisition of this framework can be realized and the extra lexicon is of tremendous uses in the theoretical study of the framework and the practical applications of deep grammar to natural language processing systems.

Many open questions remain. One particular question is whether the semantic information, which has been overlooked in the current language research, can take a stronger role in the future theoretical modeling of languages. Previous representation of languages focus in deriving various elegant but brittle predicate-argument logic to do the compositional semantics and this semantic information is at most, at auxiliary to language representation framework. The semantic information is at most, local to an utterance. Using the link-type resource can bridge the gap between the different side of the representational framework. If this question can be resolved, the natural

language research would be more fruitful and finds itself to be more relevant in semantic task.

Bibliography

[1] Timothy Baldwin. Bootstrapping deep lexical resources: Resources for courses. In *Proceedings of the ACL-SIGLEX Workshop on Deep Lexical Acquisition*, pages 77–86, 2005.

[2] Colin Bannard, Timothy Baldwin, and Alex Lascarides. A statistical approach to the semantics of verb-particles. In *Proceedings of the ACL 2003 workshop on Multiword expressions: analysis, acquisition and treatment - Volume 18*, pages 65–72, Sapporo, Japan, 2003.

[3] Petra Barg and Markus Walther. Processing unknown words in HPSG. In Christian Boitet and Pete Whitelock, editors, *Proceedings of the Thirty-Sixth Annual Meeting of the Association for Computational Linguistics and Seventeenth International Conference on Computational Linguistics*, pages 91–95, San Francisco, California, 1998. Morgan Kaufmann Publishers.

[4] Joan Bresnan. Monotonicity and the theory of relation changes in LFG. In *Journal of Language Research 26:4*, pages 637–652, 1990.

[5] Joan Bresnan. Lexicality and argument structure. In *Syntax and Semantics: Proceedings of a conference, Paris, Oct 1995.*, 1995.

[6] Joan Bresnan. *Optimal Syntax.* Oxford University Press, 1998.

[7] Joan Bresnan. *Lexical-Functional Syntax.* Oxford: Blackwell, 2001.

[8] Eric Brill. Automatic grammar induction and parsing free text: A transformation-based approach. In *Proceedings of 31st Annual Meet-*

ing of the Association for Computational Linguistics, pages 259–265, 1993.

[9] Eric Brill. Some advances in transformation-based part of speech tagging. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 722–727, 1994.

[10] Ted Briscoe and John Carroll. Automatic extraction of subcategorization from corpora. In *Proceedings of the 5th Applied Natural Language Processing Conference(ANLP-97), Washington, DC, USA.*, pages 356–363, 1997.

[11] Peter F. Brown, Vincent J. Della Pietra, Peter V. deSouza, Jenifer C. Lai, and Robert L. Mercer. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479, 1992.

[12] Eugene Charniak. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence*, pages 598–603, Menlo Park, Calfornia, 1997. AAAI Press.

[13] Noam Chomsky. *Syntactic Structures.* The Hague: Mouton, 1957.

[14] Noam Chomsky. *Aspects of the theory of Syntax.* Cambridge: The MIT Press, 1965.

[15] Michael John Collins. A new statistical parser based on bigram lexical dependencies. In Arivind Joshi and Martha Palmer, editors, *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 184–191, San Francisco, 1996. Morgan Kaufmann Publishers.

[16] A. Copestake. Semantic transfer in verbmobil. In *Verbmobil-Report, 93.*, 1995.

[17] Ann Copestake and Dan Flickinger. An open source grammar development environment and broad-coverage English grammar using HPSG. In

Proceedings of the 2nd International Conference on Language Resources and Evaluation, pages 591–600, Athens, Greece, 2000.

[18] Ann Copestake, Dan Flickinger, Rob Malouf, Susanne Riehemann, and Ivan Sag. Translation using minimal recursion semantics. In *Proceedings of the 6th. International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-95)*, Leuven, Belgium, July 1995.

[19] Ann Copestake, Dan Flickinger, Ivan A. Sag, and Carl J. Pollard. Minimal recursion semantics: An introduction. In *Research on Language and Computation, Volume 3, Number 4*, pages 281–332, 2005.

[20] Ann Copestake and Alex Lascarides. Integrating symbolic and statistical representations: The lexicon pragmatics interface. In *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics*, pages 136–143, Somerset, New Jersey, 1997. Association for Computational Linguistics.

[21] Ann Copestake and Alex Lascarides. Resolving underspecified values with discourse information. In *Workshop on models of underspecification and the representation of meaning*, Bad Teinach, Germany, 1998.

[22] Martin C. Emele, Michael Dorna, Anke Ludeling, Heike Zinsmeister, and Christian Rohrer. Semantic-based transfer. In *In Proceedings of 16th International Conference on Computational Linguistics*, pages 316–321, 1996.

[23] Gregor Erbach. Syntactic processing of unknown words. In Ph. Jorrand and V. Sgurev, editors, *Journal of Artificial Intelligence IV - methodology, systems,applications*, pages 371–382. North-Holland, Amsterdam, 1990.

[24] Dan Flickinger. On building a more efficient grammar by exploiting types. In *Natural Language Engineering, Volume 6, Issue 1*, pages 15–28, Stanford, 2000. CSLI Publications.

[25] Frederik Fouvry. Lexicon acquisition with a large-coverage unification-based grammar. In *Proceedings of 10th Conference of The European Chapter. Conference Companion*, pages 87–90, Budapest, Hungary, April 2003. Association for Computational Linguistic.

[26] Jean Mark Gawron, Jonathan King, John Lamping, Egon Loebner, Anne Paulson, Geoffrey K. Pullum, Ivan A. Sag, and Thomas Wasow. Processing english with a generalized phrase structure grammar. In *Proceedings of the 20th Annual Meeting of the Association for Computational Linguistics*, pages 74–81, Menlo Park, California, 1982. Association for Computational Linguistics.

[27] Gerald Gazdar, Ewan Klein, Geoffrey K. Pullum, and Ivan A. Sag. Coordination and unbounded dependencies. developments in generalized phrase structure grammar. In *Stanford University Working Papers in Linguistics*, Bloomington, Indiana, 1982. Indiana University Linguistics Club.

[28] Gerald Gazdar, Ewan Klein, Geoffrey K. Pullum, and Ivan A. Sag. *Generalized Phrase Structure Grammar.* Oxford: Blackwell, and Cambridge, Ma.: Harvard University Press, 1985.

[29] Gerald Gazdar and Geoffrey K. Pullum. Generalized phrase structure grammar. In *Generalized Phrase Structure Grammar: A Theoretical Synopsis*, Bloomington, Indiana, 1982.

[30] Daniel Gildea and Dan Jurafsky. Automatic labeling of semantic roles. In *Computational Linguistics 28(3)*, pages 245–288, 2002.

[31] Jonathan Ginzburg, Matthew Purver, and Ivan A. Sag. Integrating conversational move types in the grammar of conversation. In *Proceedings of Bidialog, the 5th Workshop on the Semantics and Pragmatics of Dialogue*, pages 25–42, 2001.

[32] Jonathan Ginzburg and Ivan A. Sag. *Interrogative investigations. The form, meaning, and use of English interrogatives.* CSLI Publications, Stanford, CA 94305-4115, 2001.

[33] Adele E. Goldberg. *Constructions: A construction grammar approach to argument structure.* University of Chicago Press, Chicago, 1995.

[34] Alex Lascarides and Ann Copestake. The pragmatics of word meaning. In Mandy Simons and Teresa Galloway, editors, *Proceedings from Semantics and Linguistic Theory V*, pages 204–221, Ithaca, New York, 1995. Cornell University.

[35] Lluís Màrquez. Machine learning and natural language processing. Technical report, Departament de Llenguatges i Sistemes Informatics (LSI), Universitat Politecnica de Catalunya (UPC), Barcelona, Spain, 2000.

[36] Ka Kan Lo and Wai Lam. Using semantic relations with world knowledge for question answering. In *The Fifteenth Text REtrieval Conference (TREC 2006) Proceedings*, 2006.

[37] Ka Kan Lo and Wai Lam. Building knowledge base for reading from encyclopedia. In *AAAI 2007 Spring Symposium Series Machine Reading*, pages 73–78, Menlo Park, Calfornia, 2007. AAAI Press.

[38] Ka Kan Lo and Wai Lam. Enhance legal retrieval applications with an automatically induced knowledge base. In *ICAIL 2007 Workshop on Supporting Search and Sensemaking for Electronically Stored Information in Discovery Proceedings (DESI Workshop)*, pages 9–16, 2007.

[39] Christopher D. Manning and Ivan A. Sag. Argument structure, valence, and binding. In *Nordic journal of linguistics 21(2)*, pages 107–144, 1998.

[40] Takuya Matsuzaki, Yusuke Miyao, and Jun ichi Tsujii. Efficient hpsg parsing with supertagging and cfg-filtering. In *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, pages 1671–1676, 2007.

[41] Yusuke Miyao and Jun ichi Tsujii. Probabilistic disambiguation models for wide-coverage hpsg parsing. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL05)*, pages 83–90, 2005.

[42] Hiroko Nakanishi, Yusuke Miyao, and Jun'ichi Tsujii. Probabilistic models for disambiguation of an hpsg-based chart generator. In *Proceedings of 9th International Workshop on Parsing Technologies*, Vancouver, BC, Canada, 2005.

[43] Takashi Ninomiya, Yoshimasa Tsuruoka, Yusuke Miyao, Kenjiro Taura, and Jun'ichi Tsujii. Fast and scalable hpsg parsing. In *Journal of Traitement automatique des langues (TAL)*, pages 91–144, 2006.

[44] Takashi Ninomiya, Yoshimasa Tsuruoka, Yusuke Miyao, and Jun'ichi Tsujii. Efficacy of beam thresholding, unification filtering and hybrid parsing in probabilistic hpsg parsing. In *Proceedings of 9th International Workshop on Parsing Technologies*, Vancouver, BC, Canada, 2005.

[45] Fernando C. N. Pereira and Stuart M. Shieber. The semantics of grammar formalisms seen as computer languages. In *10th International Conference on Computational Linguistics and 22nd Annual Meeting of the Association for Computational Linguistics*, pages 123–129, 1984.

[46] Carl Pollard and Ivan A. Sag. *Head-Driven Phrase Structure Grammar.* University of Chicago Press and CSLI Publications, Chicago, Illinois, 1994.

[47] Sameer Pradhan, Kadri Hacioglu, Wayne Ward, James H. Martin, and Daniel Jurafsky. Semantic role parsing: Adding semantic structure to unstructured text. In *ICDM*, pages 629–632, 2003.

[48] Geoffrey K. Pullum and Gerald Gazdar. *Natural languages and context-free languages.* Linguistics and Philosophy, 1982.

[49] Vasin Punyakanok and Dan Roth. Shallow parsing by inferencing with classifiers. In Claire Cardie, Walter Daelemans, Claire Nédellec, and Erik Tjong Kim Sang, editors, *Proceedings of the Fourth Conference on Computational Natural Language Learning and of the Second Learning Language in Logic Workshop, Lisbon, 2000*, pages 107–110. Association for Computational Linguistics, Somerset, New Jersey, 2000.

[50] Ivan A. Sag, Ronald Kaplan, Lauri Karttunen, Martin Kay, Carl Pollard, Stuart Shieber, and Annie Zaenen. Unification and grammatical theory. In *Proceedings of the 1986 West Coast Conference on Formal Linguistics*, pages 238–254, 1986.

[51] Stuart M. Shieber. Using restriction to extend parsing algorithms for complex-feature-based formalisms. In *23rd Annual Meeting of the Association for Computational Linguistics*, pages 145–152, 1985.

[52] Stuart M. Shieber. A uniform architecture for parsing and generation. In *Coling Budapest 1988 Volume 1: International Conference on Computational Linguistics*, pages 614–619, 1988.

[53] Stuart M. Shieber, Gertjan van Noord, Robert C. Moore, and Fernando C. N. Pereira. A semantic-head-driven generation algorithm for unification-based formalisms. In *27th Annual Meeting of the Association for Computational Linguistics*, pages 7–17, 1989.

[54] Jeffrey Mark Siskind. A computational study of cross-situational techniques for learning word-to-meaning mappings. In *Journal of Cognition, 61(1)*, pages 39–91, 1996.

[55] Cynthia A. Thompson and Raymond J. Mooney. Acquiring word-meaning mappings for natural language interfaces. In *Journal of Artificial Intelligence Research*, pages 1–44, 2003.

[56] Yoshimasa Tsuruoka, Yusuke Miyao, and Jun'ichi Tsujii. Towards efficient probabilistic hpsg parsing: integrating semantic and syntactic preference to guide the parsing. In *Proceedings of the IJCNLP-04 Workshop on Beyond Shallow Analyses.*, 2004.

[57] Hans van Halteren, Jakub Zavrel, and Walter Daelemans. Improving accuracy in nlp through combination of machine learning systems. In *Computational Linguistics Volume 27, Issue 2*, pages 199–229, 2001.

[58] Markus Walther and Petra Barg. Towards incremental lexical acquisition in hpsg. In *Proceedings Joint Conference on Formal Gram-*

mar, Head-Driven Phrase Structure Grammar, and Categorial Grammar, Saarbrucken., pages 289–297, 1998.

[59] Jeremy Yallop, Anna Korhonen, and Ted Briscoe. Automatic acquisition of adjectival subcategorization from corpora. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, pages 235–242, 2005.

[60] Richard Zens, Franz Josef Och, and Hermann Ney. Phrase-based statistical machine translation. In *Proceedings of 25th German Conference on Artificial Intelligence (KI2002), volume 2479 of Lecture Notes in Artificial Intelligence (LNAI) Aachen, Germany, September.*, pages 18–32, 2002.