How Everyday Language Can and Will Boost Effective Information Retrieval

Eduard Hoenkamp
Queensland University of Technology and Radboud University

Peter Bruza
Queensland University of Technology

Abstract

Typing two or three keywords into a browser has become an easy and efficient way to find information. Yet, typing even short queries becomes tedious on ever shrinking (virtual) keyboards. Meanwhile, speech processing is maturing rapidly, facilitating everyday language input. Also, wearable technology can inform users proactively by listening in on their conversations or processing their social media interactions. Given these developments, everyday language may soon become the new input of choice. We present an IR algorithm specifically designed to accept everyday language. It integrates two paradigms of information retrieval, previously studied in isolation; one directed mainly at the surface structure of language, the other primarily at the underlying meaning. The integration was achieved by a Markov machine that encodes meaning by its transition graph, and surface structure by the language it generates. A rigorous evaluation of the approach showed, first, that it can compete with the quality of existing language models. Second, that it is more effective the more verbose the input. And third, as a consequence, that it is promising for an imminent transition from keyword input, where the onus is on the user to formulate concise queries, to a modality where users can express more freely, more informal, and more natural their need for information in everyday language.

Introduction

We may argue, inquire, negotiate, gossip, flirt, prattle and bluster, words often streaming like a river. So why then, when it comes to search engines, are we suddenly so tight-lipped? What is it that changes after someone approaches a search engine for the first time, blithely typing complete sentences in everyday language? Well, first there is the helpful bystander who usually steps in to demonstrate how much more efficient it is to type two or three keywords instead. And recall how quickly the novice gets better at selecting keywords that set useful documents apart from the rest, discouraged to ever again search using everyday language. Second, there is the World Wide Web that seems to be the one place where people usually want what they get, rather than get what they want, so much so that Google has an "I'm Feeling Lucky" button under its search box. So, is the search engine's performance so superb? Or do people easily and willingly adapt to whatever new technology comes along? Either way, there is little incentive for change. And so attempts to allow for natural language input such as Ask Jeeves, The Electric Monk, Answers.com and several others, were short-lived. It may be tempting to make a connection with early work in artificial intelligence (AI). In the early 1970s, for example, a program that would correctly answer questions like “how many feet are there in π light years” would count as valid AI research. Today, however, one can enter that same question (in many a form or language) into Google’s search box. Not only will Google return the correct answer, but it does this so fast, that yesteryear’s AI researcher would be awestruck. So, it seems queries in everyday language are already taken care of anyway. Perhaps, perhaps not. Sometimes queries are difficult to express in keywords. For example, the legal database Lexis-Nexus advises and supports natural language input for queries such as (gleaned from their reference guide[1]):

> Under the First Amendment, does a public school teacher have the right to express her personal political views in class? Under what circumstances can biological parents regain custody of adopted children after an adoption?

So perhaps everyday language would be appropriate for queries that are more general or conceptual in nature. Even then, it seems only as a last resort, that users are prepared to type in full what they would ask without reservation from a fellow human. If that is the reason they are so taciturn, then a way out is on the horizon. The increase in computer power over the decades makes Siri for the iPhone, Google’s Voice Search, and recent excellent dictation software, a viable replacement for typing. In other words, we expect people to start using everyday language when they no longer have to type on an ever shrinking keyboard, as their smart phone or smart watch can pick up their spoken queries. But then again, it is not just the ease of acoustic input over typing, just look at the social media. Tweets on twitter are confined to 140 characters, and yet the messages are mostly typed in an abbreviated form of everyday language, even those meant to gather information (?, ?). Social communication would soon stagnate if people started talking in keywords. Search engines could become proactive if they would listen in on the social media. They could become akin to the ‘Librarian’ in the science fiction novel Snow Crash (?, ?), the virtual assistant who appears when the protagonist needs information, and who has already scanned all available data before he is explicitly asked to do so. We have actually built a modest version of such a virtual assistant, which could listen during question time after a slide presentation. This prototype continually presents the speaker with information related to the questions being fielded (?, ?). Another prototype with monitor function was built to help women diagnosed with breast cancer form support groups. To this end, they were invited to type their diary on a website, and while they were typing the margin would display a list referring to diaries of others they might want to contact. While they typed, the list was updated in real time to reflect the evolution of their diary (?, ?). These are obviously examples where the underlying search engine had to process everyday language. Looking into the near future, we would like to add a final point in favor of search engines geared towards everyday language. Our population is aging rapidly, and people’s cognitive abilities start to decline already in middle age (age 45-49) (?, ?, ?). So one would expect that fewer and fewer people will stay keen enough to formulate concise queries and well

[1] http://www.lexisnexis.com/help/global/US/en_US/gh_natural.asp

chosen keywords. Thorough research has shown that cognitive decline affects all areas of memory, reasoning, vocabulary, and phonemic and semantic fluency (?, ?). All decline except one: the area best preserved into old age is the use of everyday language. And yes, even the current smart phone generation will sooner or later join those ranks. By now the reasons and the research focus of our approach to information retrieval will be clear. The remainder of this paper explains the foundations, the architecture, and the validation of our approach.

IR and the 'Compromised Information Need'

The introduction presented our rationale for the use of everyday language in IR. Future machinery that checks our social media, or listens in on our conversations must obviously be able to process everyday language. But in limited form we see this already today for systems that use speech recognition for automated phone dialogs that make reservations or inform us about train schedules. It would be odd indeed when people needed to formulate questions in the telegraph style they usually reserve for search engines. Yet, whether people are going to use everyday language when approaching future search engines is an empirical question, although the answer depends on two questions that can be investigated in the here and now:

- If given the option, do users prefer to express their information need in everyday language?
- Do queries in everyday language improve search results?

Let us look at the four possible answer combinations. If both answers are negative then there is no hope for improvement: Users would not like to use everyday language, and if they did anyway, it would not be useful. If both answers are positive, then users should abandon the habit of using keywords and short phrases. Currently, there is little option to choose everyday language for search engines. But the benefits are easy to imagine. Expressing queries orally is usually quicker than having to write them down. And a spoken query preserves word stress and tone of voice, information that might get lost in translation when typing it in. And since people find it easier to express their information needs to a fellow human in everyday language, there must be a reason why they don't in the case of search engines. One reason could be that the search engine's results would deteriorate. This has been investigated by Crestani et al. (?) who presented participants with standardized information needs (from the TREC collection). Using a two by two design with short and long queries in written and spoken form, he did not find such a deterioration. (In the experiment the spoken queries were recorded and transcribed to obviate the inaccuracy of speech technology at the time.) So perhaps the search interface is to blame, as Belkin et al. have argued in several publications (?, ?, ?). Hence, various experiments have been conducted that aimed at eliciting more verbose queries (regardless of the search interface). For example in some experiments a choice of extra keywords was offered that could be added to the query (?, ?). In others, the user was prompted to be more loquacious (?, ?, ?). Still others replaced the keyboard input by spoken input (?, ?). But despite the diversity of the approaches, the conclusion was time and again that the more verbose the queries the better the retrieval results. In addition, participants felt more comfortable expressing their information need in spoken rather than in written form, and they were also more satisfied with the results. All in all it seems that when searchers can express their information needs more akin to a conversation with a fellow human, it improves both the satisfaction with the search engine and the effectiveness of the search. In short, future interaction would become more like asking a librarian for help, where everyday language is the obvious mode of interaction. But particularly in the Library Sciences, the tension between the users' desire to express their information need in everyday language, and their struggle to be precise enough has been known since the nineteenth century (?, ?). A very influential idea that entered the IR literature from there is Taylor's (?) "The Process of Asking Questions." In it, he describes what happens to an inquirer's information need before he arrives at the desk of an information specialist, hoping to find an answer to his question. Taylor describes the evolution of an information need in four levels. The first level, the 'visceral need', is a vague awareness that information is missing. The second level, the 'conscious need', is the realization of a question. The concrete verbal expression of this question is the 'formalized need'. Finally inquirers may have to find a compromise between the formalized need and the terms in which the technology allows them to convey this need to the system. At the time of Taylor's publication this could for example be asking for a book, a report, or a diagram. This level is called the 'compromised need.' It may be clear how the same levels apply to someone approaching a search engine. Currently the compromised need is cast in keywords and phrases. What we envision is that the compromised need becomes less compromised when inquirers can state their question in everyday language, without the impediment of the search interface. Once this is possible, the question can initiate a dialog between the inquirer and the search engine. Indeed, recently such "conversational answer retrieval" (CAR) was put on the IR agenda by the Strategic Workshop on Information Retrieval (SWIRL) in an effort to move beyond the state of affairs of keyword input and ranked lists of documents as output (?, ?). In this section we discussed research on the benefits, adequacy, and promises of using everyday language for IR. Of course there is more at issue in conversational answer retrieval, and which has been known for a long time to librarians. This has been splendidly explained in a later publication by Taylor (?) by what he called 'question-negotiation,' and which is therefore also valuable for IR (?, ?). Yet, allowing everyday language for search interfaces goes a long way to narrow the gap between formal need and compromised need. In what follows we present a search technique that is not optimized for key word retrieval, but for everyday language in all its verbosity. In fact, we show that it has the desirable property to work better the more verbose the input, hence the more freely users can express themselves, and the fewer compromises they have make to the prevailing technology.

Statistical Language Modeling

We argued in the introduction that preparing search engines for the future means preparing them to accept everyday language. And even if they will act more proactively, look different or have become altogether invisible, it will not mean the end of search engines. Nor will it be the end of research into retrieval models. But we think that instead of adapting traditional models to new input, it may be better to build search engines from the ground up with everyday language in mind. What we aim for in this article is to carefully select and build on IR algorithms of proven value, provided they fit the paradigm of everyday language input. Let us therefore begin with some background on the search algorithms from which we will select later on. The dominant retrieval models in Information Retrieval are the *document space model* developed since the end of the 1960s (?, ?), and the more recent *language modeling approach* (?, ?). Recent as the approach may be in IR, language models go back to Markov's work around 1920 on the distribution of letters in Russian literary works, they found a revival in Shannon's theory of communication (?, ?), and were later applied in such diverse areas as speech recognition, DNA sequencing, cursive handwriting, machine translation, and spelling correction. The language modeling approach in IR turned out to be quite effective, and at least on par with the document space model. And since this paper uses the language modeling approach as part of its model, we will briefly recapitulate the necessary background. The language modeling approach posits for every document a distribution over word sequences. It then computes how likely it is for each distribution to generate the query, and the one most likely to do so identifies the most relevant document. A simple example can make this more concrete. Consider an abstract language of just the two words a and b, and two short documents in that language, D_1 and D_2:

$$\begin{aligned} D_1 &= [a\ a\ a\ a\ a\ b\ b\ b\ b\ b\ b\ a] \\ D_2 &= [a\ b\ a\ b\ a\ b\ a\ b\ a\ b\ a\ b] \end{aligned}$$

Now, given the query $Q = [a\ \ b\ \ a\ \ b]$, which document should be considered the most relevant? In the language modeling paradigm one has to start with defining a distribution for each of the documents D_1 and D_2.
This requires an answer to two questions:

- What kind of probability distribution should be used,
- How to derive the parameters of the selected distribution for each document.

Let us look at some answers that have been given in the past (see (?, ?) for a comprehensive survey). On first reading, the reader with no background in language modeling can safely skip to the next subsection ("Machines that produce documents"), which is self contained. The following paragraph is intended for readers who are familiar with language modeling, to make it easier to compare existing models to the one we will put forward. As for the first question, what distribution to use, Ponte and Croft (?) take a multiple Bernoulli model: each word has an independent probability to be present or absent in the document. As for the second question, the parameters are estimated from the frequency of the words in the document. In this case, D_1 and D_2 would get the same score, as all words in the query are also in the documents. The multinomial unigram model (?, ?) also assigns the same score because the frequencies of a and b are the same in D_1 and D_2 and hence the $p(Q|D) = \prod_i p(q_i|D)$ are the same. If Q were extended with a word c that does not appear in the documents, so that smoothing (?, ?) were called for, words would be discounted by the same amount, and again the documents would receive the same score. Basically, these methods are trying to estimate a relevance model (1) without further knowledge about the corpus, (2) under the assumption that the term occurrences are independent, and (3) in the absence of training data. These issues have received much attention. For example, several researchers have studied bigrams and trigrams (?, ?) or even studied the optimal distance over which to consider dependencies in general (?, ?, ?) or based on natural language constraints (?, ?). Metzler and Croft (?) in particular, distinguished among full independence, sequential dependence, and full dependence. The terms mean what they suggest: in sequential dependence the ranking of a document depends only on the dependency of adjacent words, whereas in full dependence any clique of words is to be considered. In this paper we consider a fourth option, halfway between sequential and full dependence, namely when a word comes after an other word, but separated by words in between. For example, in D_1 and D_2 above, one can accumulate the distances from every a to every b to derive a probability that a is followed by b. In the example, this probability is much higher for D_2 than for D_1. And as in Q every a is followed by b, could this then be a hint that D_2 is more like Q than D_1 and thus more relevant to Q? It could, but to state this formally would require a way to compare two distributions. And whatever definition one would proffer, one still has to decide on the kind of distribution to use and with what parameters. But why a particular distribution with what particular parameters is chosen over another in the IR literature is not clear. It has largely been determined on the basis of tests on large corpora, especially and almost solely from TREC (Text REtrieval Conference), which provides standardized relevance data and uniform scoring methods (?, ?). There is nothing wrong with this from an engineering standpoint. But it has meant that the best algorithm in the literature at a given point in time was one where the author could produce better results than the best results published before, and rarely because there were better arguments in favor of the chosen model, i.e. the distribution. Hence, what we will do in the next section is first elaborate how we derived and chose a particular distribution and only thereafter will we look at test results.

Machines that produce documents and the distributions they define

Imagine (or perhaps recall) that you just came back from a well-deserved vacation in the South Pacific. When someone asks you about your vacation, you are happy to recount how it was. First you tell it to the people at home, then to your neighbors, then to your colleagues at work. At first there will be much variation in your story, but by and by all has been said, and the rendition of your experience becomes stable, only mentioning the essential parts. Another example would be the evolution of a journal article that starts as a first draft and is being honed over many iterations. Or as a final example, think of an event that lands as late breaking news on your paper's front page. As days go by, the story may reappear a few times in denser form on later pages, but eventually all has been said. Now suppose a search engine would need to return the most relevant (as opposed to the most entertaining) story about your vacation. Should it be one from the earlier stages where it still meandered haphazardly along all that happened? Or one of the later more concise and orderly accounts? We prefer the latter, and we will show how such a more concise rendering can be achieved. Instead of continuing to talk about distributions as in the previous section, we are going to focus on the source of such distributions and their underlying processes. Imagine a searcher formulating a query. We model the searcher as a source that generates a stream of words, the query, according to some underlying stochastic process. The author of a document is modeled in the same way, generating the document. Intuitively, if a searcher tries to express the same content as the author of a document, then the document must be relevant to the query. Hence if we want to find the most relevant document for a given query, we do not compare query and document directly, but instead compare their sources. Of course, the source generates the stream of words, and so determines the eventual distribution of word sequences. But the standard language modeling approach compares distributions, whereas our approach compares the sources of distributions. Figure **??** illustrates and explains the approach for the documents D_1, D_2 and query Q from the last section. In the figure we use the word 'machine' where we have so far used 'source'. They are formally the same, but 'source' seems more appropriate in referring to humans, the searcher and the author. With the vacation story, you were the source, and your stories were different samples from that source. As the source is assumed to be stochastic, the words and their frequencies will change from one account to the next, as in the case of your stories. Imagine that, as in the vacation story that was told over and over again, sources M_1 and M_2 of D_1 and D_2 would go on for a long time producing one new document after another according to their distributions. If we assume for concreteness a dependency of no more than five words, then (as we will see) in the long run a would appear about as often as b for D_2 but twice as often for D_1. This is obviously different from the word counts that would suggest a 50% probability for each. Moreover, the distribution in the long run seems to reflect the impression that D_2 is more like Q than is D_1. This paper will show how the term dependencies of a particular document predict the asymptotic behavior of its source, and with it the term distribution that would be observed if the source would continue to produce new documents. The sections that follow show that the current

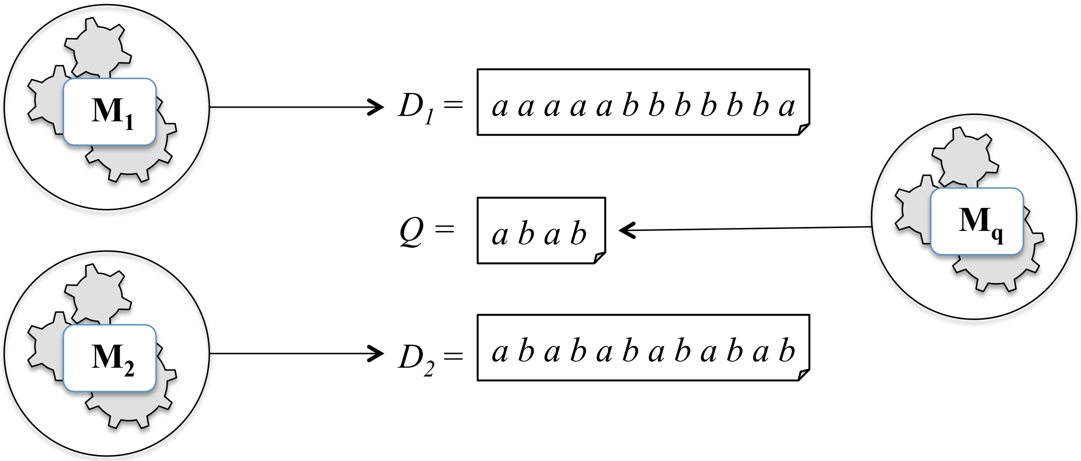


Figure 1. The language modeling approach of this paper in a nutshell. Documents are represented by machines that produce words according to some probability distribution. The figure is an example for a language of just two words, where machines M_1 and M_2 have produced documents D_1 and D_2. The query Q is likewise produced by machine M_q. We define the question of what the most relevant document is for the query by the question: *Which of M_1 and M_2 is the most difficult to tell apart from M_q by the best test* (of any test that may apply to the machines' behavior). According to the theory developed in this paper, M_2 is more difficult to tell apart from M_q than M_1, and in that sense document D_2 is more relevant to Q than D_1. (The text describes in detail how this approach was successfully applied to corpora of tens of thousands of documents consisting of hundreds of thousands of words.)

language models are overly general, as they do not confine themselves to everyday language. Restricting the language model to *everyday language makes the approach so powerful that the asymptotic behavior of the source can be derived from just one document.* This surprising result is accomplished by combining syntactic and semantic approaches that have elsewhere been used only in isolation.

A cognitive approach to the dependency problem

Before we continue, let us recapitulate what we said about language models. First, a language model needs to define the probability of a query Q given a document D, that is $Pr(Q|D)$. So assume the query Q consists of the word sequence $q_1, q_2, ..., q_n$ then the task is to find the joint probability $Pr(q_1, q_2, ..., q_n|D)$. If the terms were independent, this would simply be $\prod_i Pr(q_i|D)$. But terms are not independent of one another; they occur in context, i.e. depending on previous terms. And as we saw in the introduction, to find the actual $Pr(Q|D)$, authors have suggested to use bigrams, trigrams, and even higher order dependencies, using Bayes' chain-rule to compute the joint $Pr(q_1, q_2, ..., q_n|D)$ (?, ?, ?). There are well-known issues with this approach: First, if each term can depend on any other term, the dependency combinatorics can make applying Bayes' chain-rule intractable. Second, short documents and short queries are such small samples that they can barely represent a term distribution. Third, there may be terms in the query that are not in the document, and special methods are needed to compute probabilities for missing terms (?, ?). Finally, not only may computing the Bayesian chain rule become intractable, the needed knowledge about higher order dependencies is often simply unavailable.

The adequacy of lower order dependencies

Practical considerations aside, the question remains whether higher order dependencies would lead to better models, even if it is tempting to assume so. To begin answering this question, it is important to realize that the existing approach to language modeling is applicable to any stochastic source and the languages they produce (human, machine, or perhaps of unknown origin). The models pay no heed to the fact that the documents to be modeled are produced by humans. Yet this may throw out particular constraints that could make the methods more tractable. There are two different kinds of evidence that this may be the case:

- Many cognitive phenomena concerning memory and semantics can be understood sufficiently well in terms of word-pairs. Pertinent examples can be found e.g. in the research on memory (?, ?), work on the 'semantic space' (?, ?), and results ranging from old theories on 'spreading activation' (?, ?) to modern brain studies (?, ?) using ERP (Event-related brain potential).

- Even if higher order dependencies could theoretically lead to better search results, in practice bigrams already give a substantial improvement over unigrams (?, ?). In addition, Song and Croft (?), and others have shown that an interpolation of unigram and bigram models performs well. So perhaps the target language (a human language) is conducive to this finding.

The first point suggests that a language model need not consider more than two words at a time. The second point supports the assumption that these two words occur in sequence (evidenced by the efficacy of bigrams). As a consequence, the odds that a word appears in a sequence depends solely on the word that precedes it. So, for the sequence $q_1, q_2, ..., q_n$ in the beginning of this section, $Pr(q_1, q_2, ..., q_n) = Pr(q_1)Pr(q_2|q_1)...Pr(q_n|q_{n-1})$. This expression gives the probability of the whole sequence. But the crux is that the probability of a word to appear depends on the word that precedes it, and that word only. Now recall from figure **??** that we are interested in machinery that will crank out these words one after another, i.e. a machine M_q that models the searcher expressing query Q (and similarly for each document D_i an M_i to model its author). We can identify the state of M_q with the word it just produced since the current state completely determines the probability of the next state. So let us describe M_q as a (first order) Markov machine. The behavior of this machine is completely determined by defining the probabilities of changing from one state to the next. In this case, where there is only a finite number of states (as there are only a finite number of words) the machine's behavior is called a Markov chain. Recall that we confined the general language model by introducing cognitive constraints and practical considerations. Next we will further constrain the model by restricting the target language to everyday language.

The document as an ergodic chain

To the properties we spoke about in the previous paragraph, we will add another two that readers can easily verify for themselves. The three together circumscribe the candidate machine that models the queries:

- Machinery that produces queries or documents can be modeled as a (first order) Markov machine, hence queries and documents are samples from a Markov chain.
- These samples are word sequences from a natural language corpus. The chain therefore inherits the properties of a natural language. One property is that any two words in a sequence can be separated by any number of intermediate words. (Think of adding an extra adjective before a noun.) So there can be no cycles in the chain, or else the chain could not be extended at will. Such a Markov chain is called *aperiodic*.
- A second property is that you can always get from one word to another by continuing to produce text (words can never be used up). Consequently, the Markov chain is *irreducible*.

The first point was already proposed by Shannon in his famous article (?, ?), without the backup from cognitive science. A corollary of the next two points — that the chain is both aperiodic and irreducible — is that it is *ergodic*. An ergodic chain has the property that in the long run it reaches a stationary distribution (also called stationary kernel, or steady state), irrespective of the initial state. It is easy to sample a document and generate a new one on the basis of its distribution; see the examples in (?, ?), or any of the many sites on the web that offer programs to do this[2]. What we would like to compute however is the distribution of the source underlying the document. Or in the metaphor of the introduction,

[2] Search for 'Shannonizer' to find many links where you can enter text to generate a new text based on bi-grams.

we would like to model the final stable and concise story as the most relevant to the query about the vacation. With little knowledge of the source, one could use a Gibbs sampler, i.e. generate a long series of documents and sample until the distribution seems to converge. The Gibbs sampler was proposed for example by Wei and Croft (?, ?) to estimate the joint distribution of their model, which is a so-called LDA (Latent Dirichlet Allocation) model. Besides the benefits of that model, there are several issues to overcome: (1) it is computationally demanding, (2) it is hard to know when the process has converged, and (3) The fixed point may not be unique and e.g. depend on the initial state (the Gibbs sampler may *assume* the process is ergodic, but LDA does not imply this). The derivation above, that the process we advance here is indeed ergodic, obviates all three issues at once: The stationary distribution of the Markov chain can be efficiently computed (as we will show in the next section), no continued sampling is required to know whether the distribution has converged, and it is guaranteed to be unique. Note, first, that the properties we mentioned to derive this result are valid for natural languages in general. This means that the method may be used for languages other than English (and which are increasingly present and visible on the Web). Second, it also answers the question about the higher order dependencies, in that it is unlikely that these will contribute much to improving search results. With that answer comes an other question to the fore: how to compute the lower order dependencies given the documents. The next section offers a proposal, one we will use in an experiment further on, but it is by no means meant as the last word on finding initial distributions.

Deriving the Initial Distribution

In language modeling, the document source represents the author producing the document. As an author could produce different renderings of the same story, these renderings would be different samples of the source, and so the term distribution could differ from one document to the next.

Fortunately, the ergodic chain has a property that is very useful here, namely that its asymptotic behavior is independent of the initial state. In other words, if one would continue to sample the source, then in the long run it would not matter what sample, i.e. what document, was observed first; the asymptotic behavior would be the same. What remains then, is to derive an initial distribution given the document. This is where language models differ greatly from one another. As we mentioned in the introduction, an important distinction lies in the degree of term dependency that is assumed. In this paper we follow the approach of Lund and Burgess (?, ?) who computed co-occurrence statistics from a rich source of spontaneous conversations: Usenet newsgroups. They called the representation of these statistics the 'Hyperspace Analog to Language' or HAL. HAL is computed by sliding a window over the corpus and assigning weights to word pairs. A lower weight is assigned the greater the textual separation between the words, in a sense expressing a strength of semantic association. Or conversely, the theory posits that the nearer words are on average, the stronger they are taken to be associated in meaning. The weights are accumulated while the window moves over the whole text (or corpus). This way HALL can be represented as a matrix, whose cells contain the accumulated weights. The algorithm is probably easiest to explain with the concrete example in **Box 1**.

Box 1

For an n-word vocabulary, the HAL space is represented as an $n \times n$ matrix, with entries calculated by moving a window of constant size over the corpus. Ignoring punctuation and sentence and paragraph boundaries, the HAL theory defines a strength of association between words which diminishes with the number of intervening words. By way of example, and instead of a large-scale corpus, we illustrate the HAL algorithm with the following piece of text, after Dr Seuss (?):

One fish, two fish, red fish, blue fish
Some are red, Some are blue

Choosing a window of size 4 sliding to the right, the window becomes [one], [one fish], [one fish two], [one fish two fish], [fish two fish red], [two fish red fish] and so on. The association strength between the last word in the window to the preceding words decreases with 1 for each intervening word. The farther away, the lower the association strength. For a window of 4, the strength to a previous word is 3, to the word before it is 2, and so on. So, the first word 'fish' (in the second window) has a strength of 3 to the word 'one'. But the second 'fish' (in the fourth window) has a strength 1 to 'one', counting down over the intervening words. This is added to the previous strength, indicated in the table as 3+1. The word 'fish' does not co-occur with 'one' in any other window, so the total strength becomes 3+1=4. Looking at the fifth window, the word 'red' has a distance 3 to the 'fish' next to it, and 1 to the 'fish' further away. But after sliding the window further to the right a few times, we encounter another window containing the two words, namely [fish some are red]. This adds an extra 1 to the strength from 'red' to 'fish', so the total strength becomes 3+1+1=5.

	one	fish	two	red	blue	some	are
one	0	0	0	0	0	0	0
fish	3+1	2+2+2	3+1	3+1	3	0	0
two	2	3	0	0	0	0	0
red	0	3+1+1	2	0	0	2	3
blue	0	1+3	0	2+1	0	2	3
some	0	1+3	0	3	2	1	2
are	0	2	0	2	1	3+3	1

When the window slides off the text, the algorithm completes with the HAL matrix depicted above.

Lund and Burgess (?, ?) experimented with various window sizes, which obviously produce different HALs. They found that the associations that people make between word-pairs can best be modeled with a window size between 8 and 10. Other experiments confirmed that size as optimal to describe the correlation between word co-occurrance in corpora and strength of word association (?, ?). (The window size of 4 in **box 1** was chosen for clarity of exposition, not to model people's word associations.) If a word is connected to a second word via a small number, than it is more likely followed by that word than if the number had been high (e.g. the table shows that 'one' is more likely to be followed by 'fish' than the other way around). Based on this observation, the HAL matrix is transformed into a transition probability matrix *pHAL* by normalizing the column vectors (see e.g. (?, ?)). So, to find the document source distribution for a document requires only two steps:

1. Compute the ad-hoc distribution, in our case pHAL,
2. Compute the stable distribution (which we will call epi-HAL).

The *epi-HAL*, for 'ergodic process interpretation of HAL', is easy to compute in several ways, which follow from the ergodic property[3]. Doing this for all documents produces a source representation for each document. Now we have to choose how the documents should be ranked in order of relevance. Here are the options to compare:

- In the vector space model of IR, the distance between the query and a document is measured by the angle between their vector representations (the so-called cosine distance). The smaller the angle between document and query, the more relevant the document is to the query. Ordering the documents according to their distance to the query defines a relevance ranking.
- In the language modeling approach, the distributions of query and document are compared using the Kullback-Leibler (KL) divergence. The more a document distribution diverges from the query, the less relevant it is. (Although KL is not a distance, it is usually used as if it were.) This also defines a relevance ranking.
- In the epi-HAL model we compare the machine that is the source of the query with the machine that is the source of a document. As described in figure **??**, a document is more relevant to the query if the machine that defines it is more difficult to tell apart from the machine defining the query, by the best test that may apply. (This may be any test, from number of parts to the machine's behavior in terms of distribution).

As the behavior of the machine is defined by its source, and hence by the underlying Markov chain, we apply Kulback and Leibler's (?) theory to the stable distributions of the underlying machines. This leads to the same KL divergence formula as in the traditional language modeling approach. Hence formally the ranking is the same, and the only difference is the way the distributions are derived. As we will see, this makes it very easy to compare our approach to other language models. The algorithm we use is explained in **Box 2** using a very simple language for clarity.

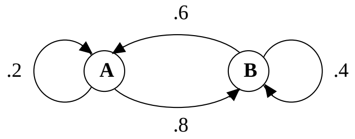
[3] We multiply the transition matrix by itself, multiply the result by itself, etc. As the number of transitions computed this way is exponential in the number of multiplications, the distribution converges very fast to a stable distribution, requiring only a few multiplications.

Box 2

For readers unfamiliar with the Markov approach, the essential steps in the algorithm are illustrated below. Assume a language of just the words a and b, with dependencies as defined by the transition probabilities in matrix H. H defines a Markov chain, where state **A** ouputs a and state **B** outputs b.

H	a	b
a	.2	.6
b	.8	.4

$\Rightarrow$



For initial state s_0 (e.g. **A** if started with word a), the next state is given by $s_1 = s_0 * H$, where

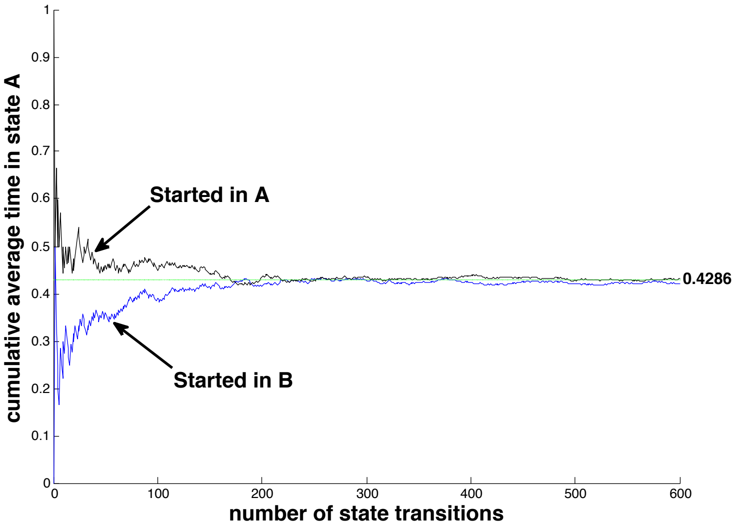
$$H = \begin{pmatrix} .2 & .6 \\ .8 & .4 \end{pmatrix}$$

followed by $s_2 = s_1 * H = s_0 * H^2, ..., s_n = s_0 * H^n$ with

$$H^n = \tfrac{1}{.8+.6} \begin{pmatrix} .6 & .6 \\ .8 & .8 \end{pmatrix} + \tfrac{-0.4^n}{.8+.6} \begin{pmatrix} .8 & -.6 \\ -.8 & .6 \end{pmatrix}$$

which converges to: $\lim_{n\to\infty} H^n = \begin{pmatrix} .4286 & .4286 \\ .5714 & .5714 \end{pmatrix}$

showing that the Markov chain becomes stationary with $P(a) = .4286$ and $P(b) = .5714$, independent of the initial state. The figure below is a simulation of the Markov chain, which illustrates, but does prove, the same results.



The stationary distribution can also be computed directly from the transition matrix. Doing this for the examples in the introduction, and constructing the HAL matrix with a window of size 4, the distributions converge to:
$D_1 = [a\ a\ a\ a\ a\ b\ b\ b\ b\ b\ b\ a]$, $P(a) = .36$ and $P(b) = .64$
$D_2 = [a\ b\ a\ b\ a\ b\ a\ b\ a\ b\ a\ b]$, $P(a) = .49$ and $P(b) = .51$
$Q = [a\ b\ a\ b\ ]$, $P(a) = .44$ and $P(b) = .56$
Computing the Kullback-Leibler divergence from Q to the documents yields
$KL(Q||D_1) = .017$, and $KL(Q||D_2) = .007$,
so D_1 diverges more from Q than D_2 and therefore D_2 is ranked as more relevant.

Note that a longer query corresponds to a larger sample from the source, so one would expect that longer queries would automatically be more effective. That in turn speaks in favor of the verbosity of everyday language to formulate queries. However, in light of an observation published by Bendersky and Croft (?), which seems to speak against this, empirical verification is called for. Therefore, the next section will add a more practical justification by showing that even a straightforward and simple implementation of our approach can already compete with that closely related but much more sophisticated language model.

Implementation and Evaluation

There certainly are other language models that use a Markov approach. Besides the work by Wei and Croft (?) we mentioned earlier, especially Cao, Nie, and Bai (?) and several others use the Markov chain for a similar reason as we do, namely to find a stable distribution to represent the document. But there are a number of choices made in (?, ?) that we do not depend on: we do not use WordNet (for semantic relationships), there are several parameters we do not have to set, and we don't use training for optimization. Furthermore, the authors of (?, ?) make use of a stationary distribution, but as with the other publications there is no indication, let alone a proof, that the algorithm has only one fixed point. So, e.g. depending on the initial state, the stationary distribution may or may not be the one sought after. But exploiting ergodicity as we propose, obviates the issues found in these publications: The final distribution can easily be computed without sampling, it converges very fast, and it is guaranteed to be unique. We will now turn to an experimental evaluation of our ergodic process interpretation of HAL (epi-HAL). We think that the more verbose the query, the more representative it is for the underlying Markov chain. Stated differently, we expect that the verbosity of everyday language improves search results over terse queries of two or three keywords. Note that this expectation follows from the algorithm we propose. It is not an empirical finding so the expectation needs to be verified.

Everyday language, verbosity, and long queries

In the section on IR and the 'Compromised Information Need', we reviewed experiments that studied how verbosity affects the quality of search results. Recall that they usually go in two phases: first a short query is issued and search results are collected. Then the query is expanded by adding more words and again the search results are collected. A comparison is then made between the search quality before and after query expansion. One can distinguish two experimental paradigms, depending on the amount of user interaction:

1. Expanding the query automatically through so-called *pseudo-relevance feedback*. The method is as follows. After a user has issued a query, the search engine retrieves (potentially) relevant documents and uses the top-ranked documents to select other relevant terms (up to several hundreds). Without user interaction, the query is expanded with these new terms and again passed to the search engine. The documents retrieved this second time are presented to the user. The method is known to lead to an overall improvement of search results (?, ?), although the results vary with conditions that were discovered later (?, ?).

2. Expanding the query by encouraging the user to elaborate. Although (?, ?) remarks that this approach was already undertaken in the 1980s, careful user experiments

came after automatic expansion had already shown its efficacy. And as we have seen, people are indeed more satisfied with search results after they were encouraged to use longer queries.

In brief, whether queries were expanded by the search engine or by the searcher, verbosity turned out to improve search results. Hence, we expect the more lengthy and wordy aspect of everyday language input to improve search results over the terse queries we are used to, which is good news. Since this covers the whole spectrum of query reformulation, it seems that further improvements can only be gained by improving the search engine itself. Notice, however, that the present paper is based on a language modeling approach, while the experiments just mentioned relied on the vector space model. So perhaps we should be careful before generalizing the promising results to the language modeling approach. The issue of verbosity in language modeling was first raised by Bendersky and Croft (?), who contend that current search engines "do not, in general, perform well with longer, more verbose queries" (p. 491). This could be true, but it contradicts the findings we mentioned earlier. Unfortunately the authors do not refer to other work to back up their claim. They do, however, report on an experiment they conducted on a much larger corpus, where increased verbosity lead to decreased quality of search results. They give an explanation for their finding, and if their result would generalize to our case, that would be bad news. So let us turn to that explanation and see if it holds for our approach. Bendersky and Croft simulate increasing verbosity by using TREC 'topics' and take the *description* field as a more verbose version of the *title* field. An example of a TREC topic and its fields is given in Table **??**. If our intuition were correct one would expect better results for the description than for the title. They found, however, the reverse to be true. We concur that the topic title represents the main concept to be searched for. So let us follow Bendersky and Croft's argument on p. 491 of their paper: (1) "there is no explicit information in the description itself to indicate which of these concepts [in the description] is more important", and (2) "A verbose query could also potentially contain two or more equally essential key concepts." In other words, their explanation is that the key concept gets blurred by the verbosity surrounding it (cf. Table **??**). We think this explanation leads to two questions, or rather, predictions:

< *title* >

Efforts to enact Gun Control Legislation

< *desc* >

Documents will relate to efforts to enact Gun Control Legislation. They may also discuss implication upon and interpretation of the Second Amendment to the U.S. Constitution as it relates to these efforts.

< *narr* >

Any discussion of Gun Control – Pro or Con – in editorials, letters to editor, news items, etc. This will include efforts at gun control at all political levels – city, county, state, and federal. Of particular interest are discussions which relate these efforts to the broader issue of just what relevance does the Second Amendment have to the question. This search will be limited to U.S. efforts.

Table 1: Example of a 'topic' definition for early TREC experiments in retrieving news from the Associated Press 1988-1990 news wires. The corpus was a classic bench mark used in many IR publications. The < *desc* > and < *narr* > stand for 'description' and 'narrative.'

- Assuming the explanation is valid, what would this predict if the description and title were taken together as the new query? Such a query could become less effective then the description, because it is more verbose. Alternatively, it could become more effective because someway the key concept becomes more prominent. Or, combining the two arguments, a safer guess might be that it lands between the efficacy of description and title in isolation. So here we have a hypothesis that can be tested.

- Given that the HAL representation captures the semantic relationships between words in the corpus (?, ?, ?), the cohesion between key concepts would be enhanced by the co-occurance of words expressing the concepts. In turn, that would increase the weight of certain words by increasing their value in the joint probability distribution (the query model). And so it would predict a higher effectiveness of title and description together, than either in isolation[4]. This is another hypothesis that can be tested.

We are now ready to evaluate the epi-HAL approach with hypotheses that will tell if increasing verbosity does or does not improve search results. The evaluation of epi-HAL follows a traditional paradigm in information retrieval, which has been used for years in the TREC experiments. To evaluate and compare search algorithms, TREC provides corpora and carefully crafted queries, called 'topics' of the sort we have seen already, to search in a given corpus. For each topic, TREC lists the documents that are relevant to the topic. (Details of how this list is compiled, can be found on the TREC site). We used corpus AP88/89 which consists of the press releases from the Associated Press during 1988 and 1989. This is a well-known corpus, which has been used may times to evaluate IR-based language models. At the same time, it offers a framework to compare epi-HAL with the Relevance model, which is a baseline language model with solid overall performance. Table **??** shows an example of a topic for AP88/89. After the experimental evaluation, we add more evidence by replicating Bendersky and Croft's experiment, using the same corpus and topics as theirs. They found that verbosity hinders rather than improves performance, and so we can pit their results against ours. The comparison, however, is not meant to show that we get better search results (although we do) but to demonstrate that the performance of our method also improves with verbosity for a corpus substantially larger than the AP corpus. To show this, we don't need more than one large collection, and for comparison we chose from one of their larger collections, ROBUST04, rather than their wide variety of corpora. To test a search algorithm, a topic is selected and the the corpus is searched for documents matching the topic. The result is a list of documents that the search algorithm considers relevant. This list is compared to what TREC lists as relevant, resulting in two numbers, 'precision' and 'recall.' Precision is the proportion of the documents found that are actually relevant, and recall is the proportion of relevant documents that were actually found. Note that users find precision usually more important then recall, as we confirmed experimentally in (?, ?). Hence we use precision as our yardstick. We can see in Table **??** how a topic is first formulated very concisely, and then is elaborated further in the description and even more in the narrative. In our experiment we take the first formulation to represent what users type into a search box, and the second and third to represent how they would ask for information from a fellow human. (In a future scenario the searcher's wearable technology could instead be listening in.) This way we can measure how precision

[4] Bendersky and Croft propose to enhance the focus on the key concept using a learning algorithm to weight the words in the query. A different approach that might lead to the same result.

changes on average with increasing verbosity. We used the freely available search engine toolkit Lemur[5] which can be parametrized for a wide range of language models.

Experimental Results

Besides the title and description from the TREC topics, we also added the narrative, as it is even more verbose than the description. Precision is expressed in two of the more common measurements used in the TREC experiments. Note that precision goes at the expense of recall and vice versa. That is, to retrieve more relevant documents, you can make the query less restrictive, but then have to accept that a greater proportion will be irrelevant. Conversely, making the query more restrictive increases the proportion of relevant documents in the set that is retrieved, but at the cost of not retrieving some relevant ones. Therefore the 'MAP' value in the table, for 'mean average precision,' averages the precision over different levels of recall. The other measure given in the table is 'prec@5', or 'precision at 5' which takes the 5 highest ranked documents, and computes the proportion of relevant documents among those. The results for AP88/89 in terms of MAP and prec@5 are given in Table **??**. The 'Baseline' are the numbers for the default settings for the Lemur toolkit. The second line gives the values when using the successful relevance model by Lavrenko and Croft (?), as implemented in Lemur. The last line show the results for our current model, which uses default values of the Lemur toolkit, but changed just one detail: instead of the default computation of term distributions, we used the epi-HAL distribution. Looking at the values for the baseline and the relevance model, we can confirm the observation by Bendersky and Croft that the more verbose description has a lower precision than the shorter title. But in the next column the title and description are taken together, which is more verbose than either in isolation, but the precision increases. In the column after that, the title and narrative are combined, giving an even more verbose

AP88/89	———	$< title >$	$< desc >$	$< title, desc >$	$< title, narr >$	$< title, narr_{-rc} >$
Baseline	MAP	23.6	22.7	28.8	31.7	31.9
	prec@5	41.2	44.4	48.8	50.8	50.0
Relevance model	MAP	29.5	29.0	32.3	32.8	33.0
	prec@5	43.6	44.0	42.8	48.8	46.4
Stable Distribution (epi-HAL)	MAP	32.3	32.4	35.7	39.5	39.3
	prec@5	46.0	46.4	46.2	60.0	58.2

Table 2: Comparing precision for various degrees of verbosity and different language models for AP88/89 topics 101-150. *title*, *desc*, and *narr* stand for the corresponding TREC fields. $narr_{-rc}$ stands for narratives with the topic 101-150 exclusion clauses removed. 'Baseline' is from Lemur's default simple language model, 'Relevance model' follows (?, ?), and 'epi-HAL' is the model proposed in the current paper.

[5] downloadable from http://www.lemurproject.org

query and again a greater precision. This contradict the hypothesis that greater verbosity leads to lower precision. It also means that Bendersky and Croft's explanation for the dip in precision, when going from title to description, cannot be valid. The last line shows our proposed model, for which precision increases with increasing verbosity. Trying to fine-tune our results even further, we selected topics 101-150 of AP88/89 for our experimentation, because it has an exclusion clause in the narrative. For example topic 102, describing Laser research for SDI (the Strategic Defense Initiative), ends with "However, a document clearly focused on use of low-power lasers in consumer products, surgical instruments, or industrial cutting tools is NOT relevant." We used two versions of each narrative, one with, and one without the exclusion clause. This way we could get an indication of the effect of verbosity: with the exclusion clause intact, the query is obviously more verbose, but more off focus. The numbers seem to tell that verbosity, even adding information that is related but not to be retrieved (the NOT clause) increases precision. Although in support of our claim, we have not investigated this further. Bendersky and Croft blame verbosity for blurring the key concepts to explain the dip in precision going from title to description. To counteract this concept blurring, they adopt a machine learning approach to identify which noun phrases in the description represent key concepts. We already showed that for epi-HAL precision increases with verbosity, the algorithm did not suffer the dip in precision. So if Bendersky and Croft are correct, epi-HAL apparently picks out the key concepts anyway. As an additional test, we pitted their model against ours for ROBUST4, a large corpus of over half a million documents. The results are shown in Table **??**. The MAP values are taken from their publication, which obviously gives no value for the combination of title and description. The last column in the table gives the mean average precision (MAP) for their best performing learning algorithm. It shows that greater verbosity lets epi-HAL improve significantly over even the best MAP results of the learning algorithm. To summarize the results of the two experiments we conducted: First, the epi-HAL approach is more effective the more verbose the query. Second, if the decrease in precision in some well-known language models is due to confusing the key concepts in more verbose queries, then apparently epi-HAL is not susceptible to this effect. Third, as epi-HAL performs better with increasing verbosity, it seems to focus automatically on the underlying concepts in everyday language input, hence is a promising approach for future search engines.

ROBUST04		$< title >$	$< desc >$	$< title, desc >$	$KeyConcept$
Baseline (B&C)	MAP	25.28	24.5	-	26.2
epi-HAL	MAP	31.10	31.0	33.1	-

Table 3: Mean average precision (MAP) results for ROBUST04, a large corpus of over 500,000 documents, tested for 250 topics. The first row shows the MAPs for a language model that uses a learning algorithm to detect key concepts in the description part of the topic. The second row are the epi-HAL results for the same corpus and topics. The first two columns are as in Table **??**.

Conclusion

In the introduction we argued that it is only a matter of time before everyday language becomes the input of choice for search engines. Not only will speech input become more ubiquitous, replacing ever shrinking keyboards. But before long we can expect that wearable technology will listen in to everyday speech from the people wearing it, or the social media they attend to, thus being able to proactively fill a person's need for information. We also mentioned that in contrast to all other cognitive abilities, people can use everyday language with agility into old age, making it an input method that works for life. Without extrapolating too much, we think that search engines will need to be able to cope with two aspects of everyday language. One is that people often do not follow textbook grammar and often do not finish sentences. But we think that this is already taken care of by the "bag of words" approach of traditional search algorithms. These algorithms produce such good results without caring about grammar, that they are sometimes the envy of more grammar oriented computational linguists. The other aspect, one that we focused on in this paper, is the verbosity of everyday language. Initially we just wanted to build an algorithm restricted by the fact that documents are written in natural language, and hence based on the intuition that existing language models were overly general in that respect. The algorithm we subsequently proposed is based on very simple observations about the structure of natural language, and on well documented cognitive principles that have been known for quite some time. This resulted in the ergodic markov model 'epi-HAL' that addresses the sentence structure of language in the distribution of the words it generates, whereas its transition matrix represents the underlying semantic relationships between words. The algorithm proceeds as a fairly traditional language model, except that the source of query and documents are compared not based on the surface distributions of the words they contain, but on the stable (asymptotic) distribution of the markov chain generating their distributions. We evaluated the algorithm on the well-known TREC corpus of Associated Press news releases, and found that even in its simplicity it can compete and even surpass other state-of-the-art language models. We also tested the algorithm on a more ambitious corpus (ROBUST4) and found that it was immune to the degeneration of precision with verbosity, observed by others. Hence we concluded that the algorithm by its nature can select key concepts from everyday language utterances. With its focus on everyday language, the proposed model is robust in dealing with verbosity, it facilitates a manner of input that can be used into old age, and it supports a language that has evolved, was honed, and been perfected over millennia as one of the most efficient and effective communication tools available to man.

References

Allan, J., Croft, W. B., Moffat, A., & Sanderson, M. (2012, June). Frontiers, challenges, and opportunities for information retrieval. *SIGIR Forum*, *46*(1), 2-32.

Anderson, J. (1983). *The architecture of cognition.* Cambridge, MA, USA: Harvard University Press.

Azzopardi, L., Girolami, M., & Crowe, M. (2005). Probabilistic hyperspace analogue to language. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval* (p. 575-576). New York, NY, USA: ACM.

Belkin, N. J. (2000). Helping people find what they don't know. *Commun. ACM*, *43*(8), 58–61.

Belkin, N. J., Kelly, D., Kim, G., Kim, J.-Y., Lee, H.-J., Muresan, G., et al. (2003). Query length in interactive information retrieval. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 205–212). New York, NY, USA: ACM Press.

Belkin, N. J., Marchetti, P., & Cool, C. (1993). BRAQUE: Design of an interface to support user interaction in information retrieval. *Information Processing and Management*, *29*(3), 325–344.

Bendersky, M., & Croft, W. B. (2008). Discovering key concepts in verbose queries. In *Sigir '08: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 491–498). New York, NY, USA: ACM.

Bruza, P., & Song, D. (2003). A comparison of various approaches for using probabilistic dependencies in language modeling. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA: ACM.

Buckley, C., Salton, G., Allan, J., & Singhal, A. (1994). Automatic query expansion using SMART. In *Proceedings of the third text REtrieval conference (trec-3). NIST Special Publication 500-225* (p. 225-237).

Burgess, C., Livesay, K., & Lund, K. (1998). Explorations in context space: Words, sentences, discourse. *Discourse Processes*, *25*, 211–257.

Cao, G., Nie, J. Y., & Bai, J. (2007). Using markov chains to exploit word relationships in information retrieval. In *the 8th conference on large-scale semantic access to content (riao07).*

Chang, Y.-W. (2013). The influence of taylors paper, question-negotiation and information-seeking in libraries. *Information Processing and Management*, *49*(5), 983 - 994.

Chwilla, D., & Kolk, H. (2005). Accessing world knowledge: Evidence from n400 and reaction time priming. *Cognitive Brain Research*, *25*, 589-606.

Crestani, F., & Du, H. (2006). Written versus spoken queries: A qualitative and quantitative comparative analysis. *JASIST*, *57*(7), 881-890.

Gao, J., Nie, J.-Y., Wu, G., & Cao, G. (2004). Dependence language model for information retrieval. In *Proceedings of the 27th annual international conference on research and development in information retrieval* (pp. 170–177). New York, NY, USA: ACM Press.

Hoenkamp, E. (2012). Taming the terabytes: A human-centered approach to surviving the information deluge. In J. Strother, J. Ulijn, & Z. Fazal (Eds.), *Information overload: An international challenge for professional engineers and technical communicators* (pp. 145–174). John Wiley & Sons, Inc.

Hoenkamp, E., & Overberg, R. (2006). Computing latent taxonomies from patients' spontaneous self-disclosure to form compatible support groups. In *Proceedings of the 20th international congress of the european federation for medical informatics (MIE 2006)* (p. 968-974). IOS Press.

Hoenkamp, E., & Van Vugt, H. (2001). The influence of recall feedback in information retrieval on user satisfaction and user behavior. In *Proceedings of the 23rd annual conference of the cognitive science society* (p. 423-428). Hillsdale, NJ: Lawrence Erlbaum Associates.

Holland, C. A., & Rabbitt, P. (1991). The course and causes of cognitive change with advancing age. *Reviews in Clinical Gerontology*, *1*, 81–96.

Kelly, D., Dollu, V. D., & Fu, X. (2005). The loquacious user: a document-independent source of terms for query expansion. In R. A. Baeza-Yates, N. Ziviani, G. Marchionini, A. Moffat, & J. Tait (Eds.), *Sigir* (p. 457-464). New York, NY, USA: ACM.

Kelly, D., & Fu, X. (2007). Eliciting better information need descriptions from users of information search systems. *Information Processing and Management*, *43*(1), 30-46.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Ann. Math. Statist.*, *22*(1), 79-86.

Lafferty, J., & Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference* *on Research and development in information retrieval* (pp. 111–119). New York, NY, USA: ACM.

Lavrenko, V., & Croft, W. B. (2001). Relevance based language models. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 120–127). New York, NY, USA: ACM. Available from `http://doi.acm.org/10.1145/383952.383972`

Lund, K., & Burgess, C. (1996). Producing high-dimensional semantic spaces from lexical cooccurrence. *Behavior Research Methods, Instruments, & Computers*, *28*(2), 203-208.

Metzler, D., & Croft, W. B. (2005). A markov random field model for term dependencies. In *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA: ACM Press.

Mitra, M., Singhal, A., & Buckley, C. (1998). Improving automatic query expansion. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 206–214). New York, NY, USA: ACM.

Ponte, J. M., & Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proceedings of the 21st conference on research and development in information retrieval* (p. 275-281).

Sagolla, D. (2009). *140 characters: A style guide for the short form.* Wiley.

Salton, G. (1968). *Automatic information organization and retrieval.* New York: McGraw-Hill.

Seuss, D. (2010). *One fish, two fish, red fish, blue fish.* HarperCollins Children's Books.

Shannon, C. E. (1948, July and October). A mathematical theory of communication. *Bell System Technical Journal*, *27*, 379-423 and 623-656.

Shiffrin, R. M., & Steyvers, M. (1998). The effectiveness of retrieval from memory. In M. Oaksford & N. Chater (Eds.), *Rational models of cognition* (p. 73-95). Oxford University Press.

Singh-Manoux, A., Kivimaki, M., Glymour, M. M., Elbaz, A., Berr, C., Ebmeier, K. P., et al. (2012, 1). Timing of onset of cognitive decline: results from whitehall ii prospective cohort study. *BMJ*, *344*.

Song, F., & Croft, W. B. (1999). A general language model for information retrieval. In *Proceedings of the 22nd conference on research and development in information retrieval* (pp. 279–280).

Stephenson, N. (1992). *Snow crash.* Bantam Books.

Taylor, R. S. (1962). The process of asking questions. *American Documentation*, *13*(4), 391 - 396.

Taylor, R. S. (1968). Question-negotiation and information seeking in libraries. *College and Research Libraries*, *29*(3), 178–194.

Voorhees, E., & Harman. (2005). *TREC: Experiment and evaluation in information retrieval.* MIT Press.

Wang, P. (2011). Information behavior and seeking. In I. Ruthven & D. Kelly (Eds.), *Interactive information seeking, behavior and retrieval* (pp. 15–41). London: Facet.

Wei, X., & Croft, W. B. (2006). Lda-based document models for ad-hoc retrieval. In *Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 178–185). New York, NY, USA: ACM Press.

Yuan, X., Belkin, N., Jordan, C., & Dumas, C. (2011). Design of a study to evaluate the effectiveness of a spoken language interface to information systems. *Proceedings of the American Society for Information Science and Technology*, *48*(1), 1–3.

Zhai, C. (2006). Statistical language models for information retrieval. tutorial presentation at the. In *29th annual international ACM SIGIR conference on Research and development in information retrieval.* New York, NY, USA: ACM.

Zhai, C., & Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval* (pp. 334–342). New York, NY, USA: ACM Press.