# Language Modeling for Information Retrieval

**Annual Progress Seminar Report**

Submitted in partial fulfillment of the requirements
for the degree of

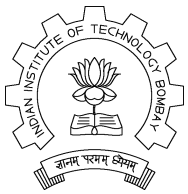
**Doctor of Philosophy**

by

**Ananthakrishnan Ramanathan**

under the guidance of

**Prof. Pushpak Bhattacharyya**
**and**
**Dr. M. Sasikumar**




Department of Computer Science and Engineering
Indian Institute of Technology, Bombay
Mumbai

**Abstract**

Language modeling is the task of estimating the probability distribution of linguistic units such as words, sentences, queries, utterances, or even complete documents. The probability distribution itself is referred to as a *language model*. Language models have been used in a variety of NLP tasks including speech recognition, document classification, optical character recognition, and statistical machine translation. Recently, the language modeling approach has been applied to information retrieval with considerable success. This approach is probabilistic in nature and is fundamentally different from the vector space methods that have been so popular in the IR community. Though it has been shown that the underlying probabilistic semantics of the language modeling approach are shared with the traditional probabilistic model, the approaches differ in the way their component models are estimated statistically. Empirical evidence suggests that the language modeling approach leads to better statistical models.

This report looks at (i) the conceptual model of information retrieval that the language modeling approach suggests, (ii) the advantages and disadvantages of this model over the traditional probabilistic models, and (iii) the manner in which this approach incorporates semantics in the smoothing model.

The report also presents results of some preliminary experiments in query expansion using language models and Wordnet.

# 1 Introduction

Language Modeling is the task of estimating the probability distribution of linguistic units such as words, sentences, queries, utterances, or even complete documents. The probability distribution itself is referred to as a *language model*.

The most popular technique for building language models is using $N$-grams. In an $N$-gram model [Jurafsky and Martin,2000], we try to predict the next word based on the previous $N$-1 words. Various other language modeling techniques such as class-based $N$-grams, probabilistic context free grammars, and decision trees are surveyed in [Rosenfeld, 2000].

Over the past couple of decades, with the increased availability of text data online, the quality of language models has increased vastly, and it has become possible to use language models in a wide variety of natural language processing (NLP) tasks including speech recognition, document classification, optical character recognition, and statistical machine translation. The language modeling approach has also led to a different perspective on information retrieval (IR), which has generated significant

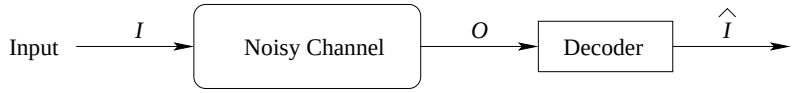



Figure 1: The Noisy Channel Model

interest in the last five years or so.

This report is organized as follows: the next section looks at how language models are used in different NLP tasks; section 3 provides an understanding of the conceptual model behind the approach; section 4 looks at how the language model is estimated and explains how the language modeling approach incorporates semantics in the smoothing model; section 5 compares the language modeling approach with the traditional probabilistic model of IR; section 6 discusses some preliminary experiments on query expansion and filtering using language models and reports initial results and observations; section 7 contains conclusions and directions for future work.

# 2 Applications

For NLP tasks, the application of language modeling is usually in the context of the the noisy channel model [Manning and Schutze, 2000, Rosenfeld, 2000]. The noisy channel model (see figure 1) sets up many NLP tasks as decoding problems. The metaphor suggests that the input $I$ is corrupted by the channel, resulting in the observed output $O$, and the task of the decoder is to find the most likely input $\hat{I}$ given $O$:

$$\hat{I} = \underset{I}{arg\,max}\; p(I|O) \tag{1}$$

Using Bayes' theorem,

$$\hat{I} = \underset{I}{arg\,max}\; \frac{p(I)p(O|I)}{p(O)} \tag{2}$$

Since the denominator is a constant (given $O$), we can write,

$$\hat{I} = \underset{I}{arg\,max}\; p(I)p(O|I) \tag{3}$$

Here, $p(I)$ is termed the *prior* and $p(O|I)$, the *likelihood*. The likelihood is a characteristic of the channel, and is often called the *channel probability* – it tells us how the noisy channel transforms the input to the output. As we shall see, the language model plays the role of the prior in certain tasks and that of the likelihood in certain others.

## 2.1 Speech Recognition

In speech recognition, given an observed acoustic signal $O$, the goal is to find the sentence $\hat{S}$ that most likely gave rise to $O$. Using the noisy channel metaphor,

$$\hat{S} = \underset{S}{arg\,max}\ p(S|O) = \underset{S}{arg\,max}\ p(S)p(O|S) \tag{4}$$

Here the language model is the distribution over sentences $p(S)$; that is, it plays the role of the prior.

## 2.2 Document Classification

The goal in document classification is to find the class $\hat{c}$ to which a given document $d$ belongs:

$$\hat{c} = \underset{c}{arg\,max}\ p(c|d) = \underset{c}{arg\,max}\ p(c)p(d|c) \tag{5}$$

Here a separate language model $p(.|c)$ is estimated for each of the predefined classes using a training set with document-class mapping. The language model for a particular class is the probability distribution of linguistic units within that class. In classification, the language model plays the role of the likelihood.

## 2.3 Statistical Machine Translation

In statistical machine translation, we suppose that the source language sentence $f$ to be translated was initially conceived in the target language as $e$. The noisy channel corrupts $e$ to $f$, and the task is to find $\hat{e}$, the most likely target language sentence given $f$:

$$\hat{e} = \underset{e}{arg\,max}\ p(e|f) = \underset{e}{arg\,max}\ p(e)p(f|e) \tag{6}$$

Here, the likelihood $p(f|e)$ is called the translation model, and the prior $p(e)$ is the language model.

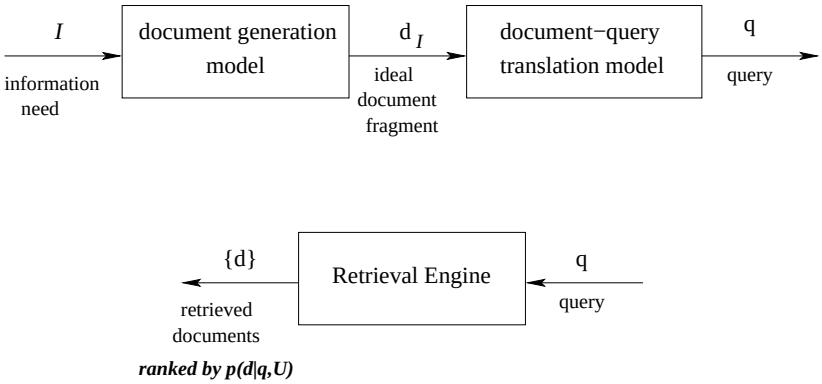



Figure 2: The conceptual model of query generation and retrieval

# 3 Language Modeling for Information Retrieval

The language modeling approach to IR was proposed by [Ponte and Croft, 1998]. In IR, the language model plays the role of the likelihood. The question asked is: What is the likelihood of the user's query given a particular document from the collection? In this section, we shall look at the conceptual model that provides this perspective on IR. The model, as described in [Berger and Lafferty, 1999], is as follows:

The user $U$ starts with a certain information need $I$. With this need in mind, the user conceives an "ideal document" $D_I$. Then, a few words or phrases from $D_I$ are selected as the query $q$.

Thus, there are two stages during query formulation: (i) generation of the ideal document, and (ii) translation of the ideal document into the query (see figure 2). This is not to suggest that queries are actually formulated this way, only that this model provides a useful perspective on the retrieval task.

Given the query $q$ and the user $U$, we want to find the most probable documents. That is, we want to rank the documents by $p(d|q, U)$. Again, using Bayes' theorem,

$$p(d|q, U) = \frac{p(d|U)p(q|d, U)}{p(q|U)} \tag{7}$$

For the purpose of ranking, we can ignore the denominator and define the relevance of a document as:

$$p_q(d) = p(d|U)p(q|d, U) \tag{8}$$

Incorporating the background and context of the user ($U$) into the retrieval model in a clean way happens to be a difficult challenge, and generally models tend to use the same ranking for all users:

$$p_q(d) = \underbrace{p(d)}_{document prior} \overbrace{p(q|d)}^{query\ likelihood} \tag{9}$$

The $p(q|d)$ term here, the query likelihood, is the language model. We also see that using Bayes' theorem has provided an explicit notion of the importance of a document irrespective of the query (the document prior). This can be used to factor in measures of "document prestige" such as Pagerank.

Figure 3 shows an example of the conceptual model explained above.

# 4 Estimation of the Language Model

In the language modeling approach, a separate language model is estimated for each document. The language model for document $d$ is just a probability distribution $p(w|d)$ over the words $w$ in the vocabulary. The query likelihood $p(q|d)$ is calculated by assuming that the query terms are independent, and then multiplying the probabilities for the individual terms. If the query $q = (q_1 q_2 \dots q_m)$ , then:

$$p(q|d) = \prod_{i=1}^{m} q_i \tag{10}$$

## 4.1 Smoothing the estimates

Looking at the example in figure 3, we see that the candidate document $d$ does not contain the query word *analysis*. Now, if we estimate $p(analysis|d)$ based on the count of *analysis* in $d$ (the maximum likelihood estimate), then this probability will be zero and the query likelihood in (10) will vanish. Thus, the language model for a document has to distribute some probability mass among words that are not in the document too. This task is called smoothing.

Smoothing techniques for language models applied to information retrieval are compared in [Zhai and Lafferty, 2001]. These methods combine the document model $p(w|d)$ with the collection ($C$) model $p(w|C)$ in some manner to smooth the probabilities. For example, the Jelinek-Mercer method linearly interpolates the document and collection models to give the corrected estimate $\hat{p}(w|d)$.

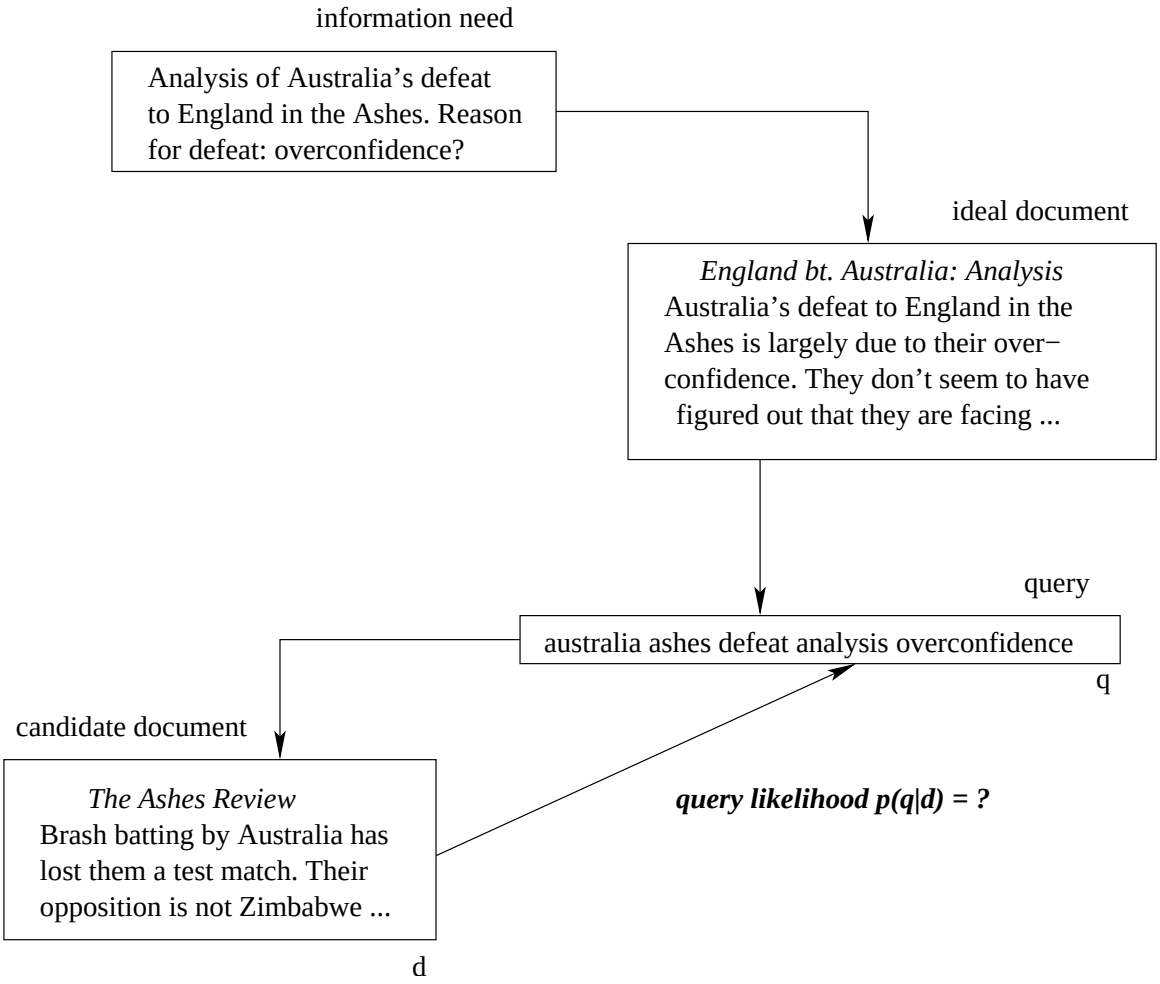



Figure 3: Query generation and retrieval based on query likelihood: example

$$\hat{p}(w|d) = (1 - \lambda)p(w|d) + \lambda p(w|C) \tag{11}$$

This kind of smoothing approximates the probability of *analysis* in $d$ by using the count of *analysis* in the entire collection $C$. But this is likely to be an underestimate for the candidate document in figure 3, because the document has the word *review* which is a synonym of *analysis*. So, when we think of the ideal document being transformed by the channel into the query, we have to incorporate semantics in the channel model. That is, the document-query translation model in figure 2 has to be sophisticated enough to allow a document word to be substituted in the query by a synonym or another related word. This is accomplished by estimating a translation model, which is a set of probabilities $t(q_i|w)$ for mapping a document term $w$ to a query term $q_i$. The query likelihood then becomes:

$$p(q|d) = \prod_{i=1}^{m} \sum_{w} t(q_i|w)p(w|d) \tag{12}$$

Consider figure 3 again. Now, the probability of *analysis* given the candidate document should be reasonable because of the link that is established through the document word *review*. That is, $p(analysis|d)$ will contain the term "$t(analysis|review)$ $\times$ $p(review|d)$". This approach, called "semantic smoothing", generally achieves much better smoothing of the probabilities.

For estimating the translation probabilities $t(q_i|w)$, [Berger and Lafferty, 1999] proposes the EM algorithm on a training corpus of query-document pairs. Since a large corpus of queries and associated documents was not available, synthetic queries were generated.

[Lafferty and Zhai, 2001b] proposes a simpler method for estimating the translation probabilities using Markov chains. This method is described in section 6.1 along with preliminary results obtained by employing the Markov chain method to filter queries expanded via Wordnet.

# 5 Language Modeling vs. the Traditional Probabilistic Model

In the traditional probabilistic model, the question asked is: What is the probability of relevance given *this* document and query? The ranking criterion is given by

$$p(r|d, q) \tag{13}$$

or equivalently, by the following log-odds ratio.

$$\frac{p(r|d,q)}{p(\bar{r}|d,q)} \tag{14}$$

The Robertson-Sparck Jones model ([Sparck Jones et al., 2000]), which has been the standard probabilistic model for many years, is derived from this. The ranking criterion for this model is

$$\frac{p(d|q,r)}{p(d|q,\bar{r})} \tag{15}$$

[Robertson, 1977] proves that this ranking is optimal if we assume that the relevance of a document to a query is independent of the other documents in the collection.

In the light of this, we need to ask whether the language modeling approach leads to an optimal ranking, especially when there is no relevance variable in the model. [Lafferty and Zhai, 2001a] shows that probabilistically the language modeling approach, also being derivable from $p(r|d,q)$, is equivalent to the Robertson-Sparck Jones approach, and therefore it too is optimal.

Though probabilistically equivalent, the two approaches differ in the way they factor $p(r|d,q)$, and consequently in the way they estimate their component models. The language modeling approach has certain advantages over the traditional approach in this regard:

- Since the conditioning in on documents, which are much larger units of text than queries, the language modeling approach provides a much larger foothold for estimating probabilities
- The approach provides a clean way of incorporating the notion of the prestige of a document via the prior $p(d)$
- Since the number of probability terms is the same across documents (number of query terms), there is no need for normalization of the probability

The one problem with the language modeling approach is that it is not possible to use relevance judgements directly. Whereas in the Robertson-Sparck Jones model, relevance judgements from the user can be used naturally to improve the estimates of $p(w|q,r)$ and $p(w|q,\bar{r})$, and these estimates can be used for judging the relevance of other documents.

# 6 Query Expansion Using Markov Chains and Wordnet: Preliminary Results

## 6.1 Estimating translation probabilities using Markov chains

The intuition behind the Markov chain method for finding translation probabilities is that related words often occur in the same documents. The translation probability is thus a measure of the co-occurrence of the words.

To estimate the probability of the word $w$ being translated as $q_i$, the chain proceeds as follows:

1. Given $w$, a document $d_i$ containing $w$ is chosen according to the probability $p(d_i|w)$, which is given by:

$$p(d_i|w) = \frac{p(w|d_i)p(d_i)}{\sum_d t(w|d_i)p(d)} \tag{16}$$

   The denominator sums over all documents in the collection.

2. Having arrived at $d_i$, another word from $d_i$ is chosen according the document language model $p(w|d)$

3. The chain is continued with probability $\alpha$, and with probability $1-\alpha$ the chain ends here. In our experiments[1], since it is very difficult to calculate chains of longer length, we have started with a chain of length 1.

If $N$ is the number of terms in the vocabulary and $M$ is the number of documents in the collection, let $A$ be the $N \times M$ matrix with entries $A_{w,d} = p(d|w)$, and $B$, the $M \times N$ matrix representing the language models, that is, $B_{d,w} = p(w|d)$. Then, for a chain of length 1, we have the $N \times N$ matrix $C = AB$, the entries of which represent translation probabilities:

$$C_{w_i,w_j} = t(w_i|w_j) \tag{17}$$

[1]done jointly with Manoj Kumar Chinnakotla, Research Scholar, KReSIT, IIT Bombay

## 6.2 Query expansion and filtering

Given the query $q = (q_1 q_2 \dots q_m)$, a candidate expansion term $w$ can be scored by $p(w|q)$. Assuming that the query terms are generated independently, this is given by

$$p(w|q) \propto \sum_{i=1}^{m} t(q_i|w)p(w) \tag{18}$$

We have experimented with the use of synonyms, hypernyms, and hyponyms from Wordnet as expansion candidates.

## 6.3 Preliminary experimental results and discussion

The TREC8 ad hoc task collection (topics 401-450 on disks 4 and 5) was used for the experiments. The translation probability matrix $C$ was computed (see section 6.1) on this collection. Associated with the collection is a set of 50 queries and corresponding relevance judgements.

The best reported results [Lafferty and Zhai, 2001b] have been obtained using terms from the local set as expansion candidates and taking the top 10 terms according to (18). We have tried the following configurations:

| expansion candidates | #terms added |
|---|---|
| local set, synonyms, hypernyms, hyponyms | 10 |
| local set, synonyms, hypernyms, hyponyms | 6 |
| local set, synonyms | 10 |
| local set, synonyms | 6 |
| local set only | 10 |
| local set only | 6 |

Table 1: Query Expansion Configurations

The precision-recall curves for these configurations are shown in figure 4. The following are the observations so far:

- Local feedback with 10 added terms outperforms all other configurations
- The top-ranked documents for some queries are better in the configuration where synonyms are used; we can see in the graph that the line for synonyms crosses the line for local feedback at low recall

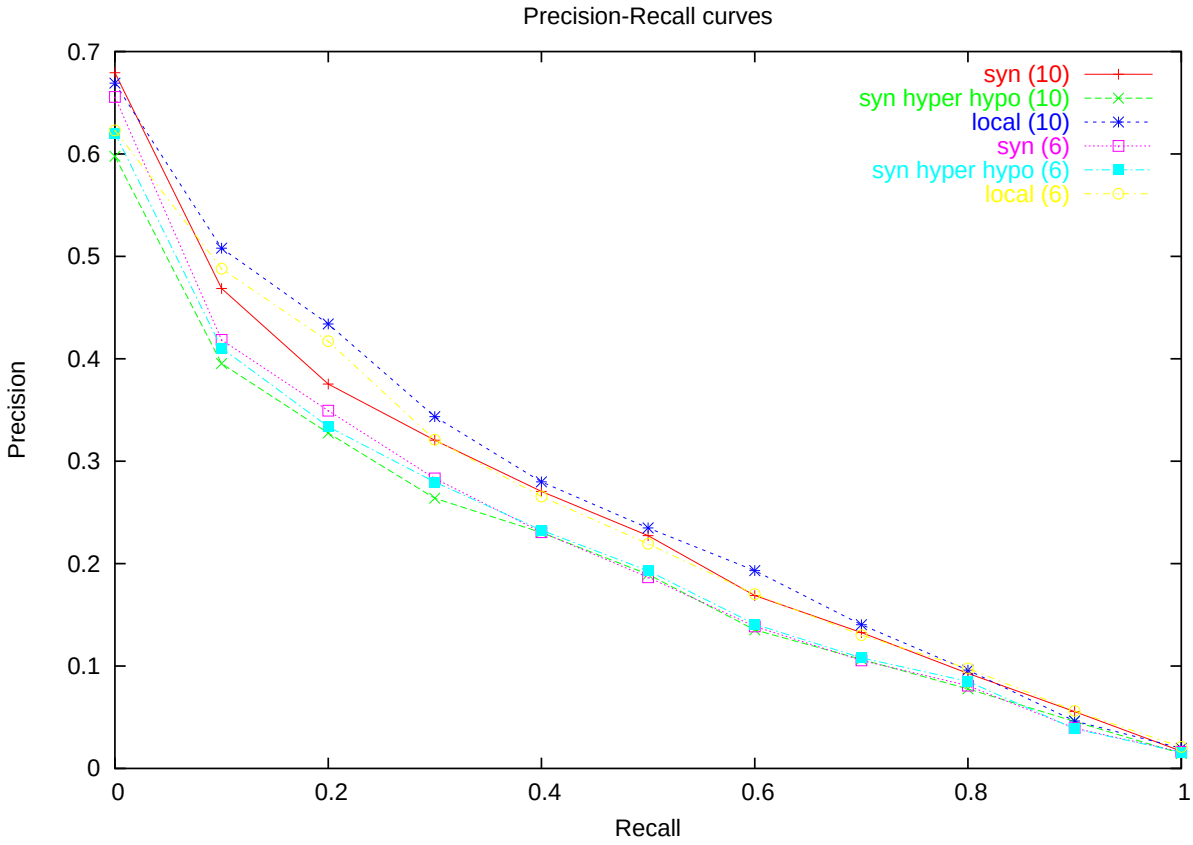



Figure 4: Results of query expansion using different configurations

- The assumption in (18) that query terms are independent often results in incorrect terms being added. For example, for the query "ireland peace talks", due to the independence assumption, being a synonym of *talk*, the expansion term *lecture* gets added, though in the context it is incorrect. The problem is that the translation model does not account for polysemy.

- The use of hypernyms and hyponyms leads to poorer results. This, however, maybe because the translation model does not filter properly, as just noted.

# 7 Conclusion and Future Work

Probabilistically, language modeling and the Robertson-Sparck Jones model of retrieval are two sides of the same coin. However, the use of the noisy channel model to reverse the probabilities provides various advantages – more reliable probability estimates, an explicit prior, and the absence of the need for normalization. These are advantages that have been recognized in other application areas of the noisy channel model too. In speech recognition, statistical machine translation, and other NLP tasks it has been found that predicting what is known (the input query in the IR case) from competing hypotheses is easier than directly predicting all the hypotheses. Additionally, the language modeling approach for IR can be naturally extended to incorporate semantics.

We have done some preliminary work in the use of language models for filtering expansion terms obtained from Wordnet. Though, we have not succeeded in improving on the best performance, there are certain interesting observations which can lead to further improvement. The plan for work in the immediate future is to revamp the translation model to account for both synonymy and polysemy. This could be done either by attacking the independence assumption in the model directly, or by adding constraints on the Wordnet synsets being added based on available sense disambiguation algorithms. We also plan to systematically vary the number of terms added in the query and find the best number for our approach.

# References

[Berger and Lafferty, 1999] A. Berger and J. Lafferty, Information retrieval as statistical translation, *Proceedings of the 1999 ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 222–229, 1999.

[Lafferty and Zhai, 2001a] J. Lafferty and C. Zhai, Probabilistic IR models based on document and query generation, *Proceedings of the Workshop on Language Modeling and Information Retrieval*, Carnegie Mellon University, 2001.

[Lafferty and Zhai, 2001b] J. Lafferty and C. Zhai, Document language models, query models, and risk minimization for information retrieval, *Proceedings of the 2001 ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 111–119, 2001.

[Jurafsky and Martin,2000] Daniel Jurafsky and James H. Martin, *Speech and Language Processing*, Pearson Education Inc, 2000.

[Manning and Schutze, 2000] Christopher D. Manning and Hinrich Schutze, *Foundations of Statistical Natural Language Processing*, The MIT Press, 2000.

[Ponte and Croft, 1998] J. Lavrenko and B. Croft, A language modeling approach to information retrieval, *Proceedings of the 1998 ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281, 1998.

[Sparck Jones et al., 2000] K. Sparck Jones, S. Walker, and S. Robertson, A probabilistic model of information retrieval: development and comparative experiments, *Information Processing and Management*, 36, pages 779–808, 2000.

[Robertson, 1977] S. Robertson, The probability ranking principle in IR, *Journal of Documentation*, 33, pages 294–304, 1977.

[Rosenfeld, 2000] R. Rosenfeld, Two decades of statistical language modeling: Where do we go from here?, *Proceedings of the IEEE*, 88(8), 2000.

[Zhai and Lafferty, 2001] C. Zhai and J. Lafferty, A study of smoothing methods for language models applied to ad hoc information retrieval, *Proceedings of the 2001 ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 334–342, 2001.