STATISTICAL APPROACHES TOWARD TITLE GENERATION

by

Rong Jin

A thesis submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

Carnegie Mellon University

2003

Approved by ____
Chairperson of Supervisory Committee

____

____

____

Program Authorized
to Offer Degree ____

Date ____

Carnegie Mellon University

Abstract

STATISTICAL APPROACHES TOWARD TITLE GENERATION

by Rong Jin

Chairperson of the Supervisory Committee: Dr. Alex G. Hauptmann
Department of Computer Science

A title is a compact representation that can help people capture a document's main idea without having to read through the entire document. Automatic title generation is a difficult natural language processing problem. It requires both the understanding of the essential content of a document and the knowledge of creating a headline that actually reflects the content with a few words. Therefore, the task of automatic title generation involves both natural language understanding and natural language synthesis.

Previous statistical approaches to title generation are based on the paper by Witbrock and Mittal (1999), where the process of title generation is divided into a phase of selecting title words for a document and a phase of organizing title words into a human readable sequence. In the work of Witbrock and Mittal and follow-ups, the phase of title word ordering is accomplished using an n-gram statistical language model and the phase of title word selection is realized by a Naïve Bayes method. In this thesis, we examine and compare seven different statistical methods for title word selection, including a nearest neighbor approach, K nearest neighbor approach, a decision tree approach, a statistical translation approach, a reverse information retrieval approach, a Naïve Bayes approach with a limited vocabulary, and a Naïve Bayes approach with a full vocabulary. In general, methods that are able to take into account all the words in the test document work better than methods that only consider a subset of document words.

The other dimension of this thesis is on the study of new model for title generation. A general probabilistic framework is proposed for the statistical approaches toward title generation, where previous works on title generation can be treated as special case of this framework. Furthermore, a new probabilistic model for title generation is derived from the general framework, which is able to overcome the problems with the previous statistical model on both the phase of title word selection and the phase of title word ordering. In the new probabilistic model, an intermediate state named 'information source' is proposed so that both the title and the document are created from this state. Unlike the previous work on title generation where titles are created directly from documents, in this new model, we will first infer the possible 'information sources' for a document and generate a title from those potential 'information sources'. Empirical studies over two different datasets have shown that the proposed model is able to outperform the previous model for title generation substantially.

Finally, we extend the title generation model to other related fields such as information retrieval and text categorization. For information retrieval, the basic idea is to treat a query as a 'title' and the problem of finding documents relevant to the query can be viewed as the problem of finding documents that fit in with the 'title'. Empirical studies over information retrieval have indicated that approaches based on the title generation model appear to work well for certain types of data.

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

ACKNOWLEDGMENTS

I would like to thank my advisor Alexander G. Hauptmann, who introduced me into the world of computer science, and has been extremely generous to me in guiding me through the sophisticated machine learning techniques and sharing me with his interesting and deep insights on many natural language problems. I would like to thank the Informedia project, which has been supporting my research work from the very beginning. I own a big thank to my wife, Chang Huang, who has kept on encouraging me whenever I felt frustrated and taken care of most of the house works so that I can focus on my research. Furthermore, I would like to thank my thesis committee members, Prof. Jamie Callan, Dr. Ralf Brown and Dr. Mike Witbrock, for spending a lot of time on reading and helping me with revising my thesis. Finally, I have to thank Prof. Zhai, Prof. Yang, my graduate fellows Luo Si, Jian Zhang, Rong Yan and Yan Lui, who have been extremely helpful in sharing their knowledge and ideas with me.

Chapter 1

INTRODUCTION

A title is a compact representation that can help people capture a document's main idea without having to read through the entire document. Automatic title generation is a difficult natural language processing problem. It requires both the understanding of the essential content of a document and the knowledge to create a headline that actually reflects the content in only a few words. Therefore, the task of automatic title generation involves both natural language understanding and natural language synthesis.

From a practical viewpoint, the significance of automatic title generation is that it produces a compact representation of the original document, which helps people to quickly understand the important information contained in a document. For example, most commercial search engines provide some kind of title for retrieved documents. These 'titles' help people save a large amount of time in finding the documents that they need. Furthermore, automatic title generation is not limited to written articles. It can be used to create titles for machine-generated texts, such as speech recognition transcripts and machine-translated documents. More interestingly, title generation can also be applied to the cross-lingual environment, where documents are written in one language and the created titles are in another. This idea was originally proposed in the paper by Witbrock and Mittal (1999). The cross-lingual title generation can be quite useful to cross-lingual information retrieval task where the created English titles may help English readers to find the relevant foreign documents quickly.

From the viewpoint of Natural Language Processing (NLP), title generation is a difficult and interesting problem. To generate a good title for a document, not only are natural language understanding techniques required in order to determine the essential content of the document but also natural language synthesis is needed for creating a human readable sentence. This comprehensive nature distinguishes automatic title generation from many other problems, such as key phrase extraction (Turney, 2000; Frank, 1999; Leung & Kan 1997), text summarization (Mani & Maybury, 1999), information retrieval (Jones & Willet, 1997) and information extraction (MUC-6, 1995), where the sole job is to identify the important content of documents. It makes title generation much more interesting and challenging. Therefore, solving the problem of title generation may significantly improve our understanding of natural language analysis and creation.

The approaches to title generation can be categorized into two groups: automatic text summarization based approaches and statistical approaches. The former approaches treat titles as summaries with extremely short length and use text summarization techniques directly for generating titles. Statistical approaches emphasize the idea of learning the title-document correlation from training corpus and applying the learned model to create titles for unseen documents. Compared to statistical approaches, the idea of treating title generation as an ultimate summarization doesn’t require the training corpus and is able to take advantage of existing research in the text summarization field. However, since most text summarization techniques are extraction-based approaches, the quality of summarization relies heavily on the quality of the original document. This nature makes text summarization approaches for title generation difficult to use for machine-generated documents such as speech-recognized documents and machine-translated documents where many text units are ‘corrupted’. Furthermore, the extraction nature of text summarization makes it impossible to use for cross-lingual title generation where documents are written in one language and the created titles are required to be in another language. On the other hand, unlike text summarization based approaches where the selection of text units as titles relies heavily on linguistic clues and handcrafted formulas, the statistical approaches obtain the knowledge of how to create titles out of documents automatically from the training corpus. Therefore, statistical approaches are domain and language independent and can be easily extended to different domains and different languages with small modification. Furthermore, since statistical approaches rely on few linguistic clues from documents, they will be robust to errors within documents. Thus, they will be suitable for degraded documents, such as speech-recognized transcripts, OCR transcripts and machine-translated documents. Of course, the biggest weakness of statistical approaches is their dependence on training corpora. Fortunately, since many web pages contain titles and document bodies and are freely accessible, we can simply crawl over the web to find enough training data for creating the statistics for the title generation. Because of the flexibility of statistical approaches and the potential existence of training data over web, in this thesis, we will focus on statistical approaches toward title generation.

Previous statistical approaches to title generation are based on the paper by Witbrock and Mittal (1999), where the process of title generation is divided into a phase of selecting title words for a document and a phase of organizing title words into a human readable sequence. In the work of Witbrock and Mittal and follow-ups, the phase of title word ordering is accomplished using an n-gram statistical language model. A Naïve Bayes approach is used for selecting title words. Many other methods have been proposed for the title word selection phase, such as a machine translation model (Kennedy & Hauptmann, 2000), an inverse information retrieval approach (Jin &

Hauptmann, 2001) and a k nearest neighbor approach (Jin & Hauptmann, 2000). In the machine translation model, the document-title pairs are treated as translation pairs and a statistical machine translation model is applied to learn the title-document correlation. In the inverse information retrieval approach, the document-title pair is viewed as a document-query relation and therefore information retrieval techniques are applied to obtain the correlation. For a k nearest neighbor approach, each title in the training corpus is treated as a class label and the task of creating titles for an unseen document is transformed into the task of finding the appropriate class label for that document.

In this thesis, we present a general probabilistic framework for statistical approaches toward title generation, which is able to incorporate previous work on title generation. Furthermore, a new probabilistic model for title generation is derived from the general framework, which is able to overcome the problems with the previous statistical model on both the title word selection phase and the title word ordering phase, which will be specified in the late part of this thesis. In the new probabilistic model, an intermediate state named ‘information source’ is proposed so that both the title and the document are created from this state. Unlike the previous work on title generation where titles are created directly from documents, in this new model, we will first infer the possible ‘information sources’ from a document and then generate a title from the potential ‘information sources’. With the help of this extra state, we are able to limit the influence of non-content words on the choice of title words. We call this model the ‘dual noisy channel model’.

Essentially, the task of creating titles for documents can be viewed as generating a concise representation for an object from a verbose representation. In this sense, it is related to many different tasks such as information retrieval and text categorization where both queries and text categories can be treated as a sort of concise representation for documents. To extend the title generation model to information retrieval, we can treat a query as a sort of title since titles and queries share many similar characteristics. The problem of finding documents relevant to the query can then be interpreted as the problem of finding documents that fit in with the pre-determined ‘title’, and thus the title generation model can be used for retrieving the relevant documents. Of course, due to the essential difference between titles and queries, modification to the original title generation model is required.

Chapter 2

OVERVIEW OF EXISTING METHODS FOR TITLE GENERATION

As already mentioned in the introduction chapter, the approaches toward title generation can be categorized into two groups, namely the text summarization based approach and the statistical learning based approaches. In the first section of this chapter, we will review the work that has been done in the field of text summarization and how title generation is related to text summarization. Then in the second section, we will introduce the machine learning view of title generation and discuss the work that has been done in terms of applying statistical learning algorithm to automatically generate titles for documents.

2.1 Automatic Text Summarization for Title Generation

Text summarization is the process of distilling the most important information from a source (or sources) to produce an abridged version for a particular user (or users) and task (or tasks). There are many uses of summarization such as headlines, previews, digests and biographies. According to the input to the automatic text summarization system and the output from the summarization system, the problem of automatic text summarization can be categorized in several ways:

1. Single document vs. Multiple documents: the summarization system can create a summary for a single document or it can be used to provide a summary across several documents.

2. Extract vs. Abstract: the summarization system can simply extract the representative text units from the original documents to form the summary or it can synthesize an abstraction of the original document based on its understanding of the document.

3. Generic vs. User-focused: a generic summary serves as a surrogate for the full text of the document, while a user-focused summary only provides the summary for the part of the document related to the user's information need.

4. Indicative vs. Informative: an indicative summary tends to indicate what topics are addressed in the source text, while an informative summary tries to cover the concepts in the original document to the extent given the compression ratio.

In general, the approaches toward automatic text summarization can be categorized into two groups: surface-level approaches and entity-level approaches. The surface-level approaches base their judgments of salient sentences on the surface-level features including the term frequency (Luhn, 1958), the location of sentence (Edmundson, 1969), the cue phrase (i.e., the phrase indicating the beginning of summary sentences) (Paice, 1990; Paice & Jones, 1993) and the number of key words or title words in a sentence (Edmundson, 1969). In terms of combining surface-level features, several machine learning algorithms have been used. In a paper by Kupiec and his colleagues (Kupiec et al., 1995), they proposed a Naïve Bayes approach to compute the coefficients for combination using a training corpus. Other learning algorithms such as decision tree, generic algorithm and semi-supervised learning algorithm have been examined for combining features in (Turney, 2000; E. Frank et al., 1999; Amini & Gallinari, 2002; Knight & D.

Marcu, 2000). Over all, the consensus is that, the location of sentences and the feature of cue phrases are far more informative than the other features. For example, people have found that for new articles, it is quite difficult to come up with a summarization strategy that is able to beat the simple algorithm which simply selects the leading sentences in the source text as the summary (Firmin & Chrzanowski, 1999). This fact indicates that, the right combination of features depends on the characteristics of the original documents, as well as the task. The entity-level approaches include syntactic analysis (Climenson et al., 1961; Skorokhodko, 1972), discourse analysis (vanDijk, 1980; Boguraev & Kennedy, 1997; Barzilay & Elhadad, 1997; Strzalkowski et al., 1998; Marcu, 1997) and semantic analysis (Reimer & Hahn, 1988; Rau et al., 1989). These approaches rely on the linguistic analysis of the source text to obtain linguistic structures such as discourse structure, syntactic structure and rhetorical structure to create summary. Furthermore, people have been working on combining surface-level approaches and entity-level approaches to have better summaries (Goldstein et al., 1999; Teufel & Moen, 1998; Hovy & Lin, 1997; Salton et al., 1997; McKeown et al., 1995; Radev & McKeown, 1998). Finally, automatic text summarization is not limited to machine-readable documents. People have tried to automatically summarize hand-written documents (Banko et al., 1999), speech of dialogues (Zechner, 2001) and diagrams (Futrelle, 1999).

One way to connect summaries with titles is to treat titles as summaries with extremely short length. Then, we can apply the methods for automatic text summarization to the task of automatic title generation. The advantage of this approach is its simplicity. We can simply take an existing text summarization system and ask it to generate summaries for documents with very high compression ratios and use the generated summaries as titles. One problem with the text summarization based approach is that, for many summarization systems, when the compression rate goes far below 10%, the quality of generated summaries can be significantly damaged (Firmin & Chrzanowski, 1999). Since a title usually consists of less than 10 words while a document contains hundreds of words, we would expect in general the compression ratio to be significantly less than 10%, which means that most text summarization approaches may create poor titles. Furthermore, it may not be appropriate to just consider a title as a summary with a short length since they could serve different purposes. The main goal of a summary is to summarize the main content of the original document into a short length while a title usually has to catch the attention of readers. For example, consider a news story about the results of an NBA regular game that the Wizard beat the Net. An appropriate short summary would be ‘The Wizard beats the Net with 108 to 97’. However, to make the story catchier, an author may come up with a title such as ‘Jordan flies again’, which doesn’t explicitly talk about the results of the NBA game between the Wizard and the Net. In this case, the learning algorithm appears to be more appropriate, which hopefully can learn from the training data that word ‘Jordan’ is frequently used in the title whenever the story is about the game of the Wizard.

In addition to the problem caused by the small compression ratio of title generation, the extractive nature of text summarization approaches constrains their application to title generation significantly in many other ways. Current research in the field of automatic text summarization focuses on the extractive approaches, i.e., to extract salient sentences from the source text to form a summary. One direct consequence is that the minimum unit for a summary has to be a full sentence, which may not be appropriate for titles since many titles are simply phrases. Therefore, applying text summarization approaches to automatic title generation task may result in overly long titles compared to human assigned titles. Another area where applying text summarization methods to title generation is difficult is the creation of human readable titles for machine-generated documents such as speech-recognized documents, OCR-recognized documents and machine-translated documents. Those documents in general contain many erroneous words, incomplete grammatical structures, and even ungrammatical structures. All these ‘corruptions’ within the documents will blur significantly the features used for selecting sentences and result in a poor choice of title sentences. Furthermore, these errors can create extracted sentences that are difficult to understand even though they are originally good candidates for titles. Even worse, for many machine-generated documents, there are no punctuations, no capitalization, and no sentence boundaries, which makes it almost impossible for extraction-based approaches to do their jobs. Therefore, the text summarization based approaches for title generation will have serious trouble with handling ‘corrupted’ documents. Another environment that will make text summarization approaches unsuitable is cross-lingual title generation, in which documents are written in one language and titles need to be generated in another language. For this task, since titles are required to be in a language different from the document, all the extraction-based approaches simply can’t do anything useful.

2.2 Statistical Methods for Title Generation

More recently, some researchers have moved toward statistical or learning approaches toward title generation. The setup of the problem is slightly different from the viewpoint of text summarization. In this approach, we assume that there are a large number of document-title pairs available. The system is asked to learn the correlation between documents and titles and then will apply the learned statistical model to create titles for unseen documents. Compared to the text summarization based approaches, the statistical learning approaches rely on the availability of training data, which is the disadvantage of these approaches. On the other hand, the ability of learning from training data can also be viewed as the strength of the statistical approaches. Unlike the text summarization based approaches where the rules of selecting representative sentences are more or less hand crafted into the systems, the statistical approaches are able to obtain the knowledge of how to compose a good title automatically from the training corpus. It is the ability of learning knowledge from a training corpus that makes the statistical approaches much easier to export to different languages and domains than the text summarization approaches. Unlike the text summarization based approaches, which usually looks at features such as the location of a sentences and the existence of cue phrases, statistical approaches focus on examining the correlation between title words and words in the document. Therefore, statistical approaches are able to take advantage of the word distribution in the original document, which provides more reliable information than features such as locations of sentences and existence of cue phrases, particularly in the case of 'corrupted documents'. In general, statistical learning approaches for title generation are quite robust to noises in documents, which makes it suitable for machine-generated documents. Finally, some statistical approaches are able to go beyond the idea of extracting important phrases from documents to form titles. The learning approaches can even produce titles containing words that don't appear in the source documents. This ability makes the statistical approaches suitable for cross-lingual title generation.

The disadvantage of statistical methods compared to the text summarization approach for title generation is that, statistical methods rely heavily on the availability of training data. Without training data, statistical approaches can do nothing while text summarization based approaches can still create titles for documents. Furthermore, because statistical methods need to compute the correlation between every title word and every document word (though the correlation matrix is rather sparse), it is computationally more expensive than text summarization based approaches. Therefore, statistical methods may not be desirable choices if either there is no training data available or the computational resource is limited.

In the first statistical framework for title generation, the process of title generation is divided into two phases, i.e., the phase of finding good title words for a document and the phase of organizing selected title words into sequences. To select appropriate title words for a document, the old title generation model takes a Naïve Bayes approach and trained the system to learn the correction between a document word ‘dw’ and a title word ‘tw’, i.e., the conditional probability P(tw|dw), and applied the learned correlation to compute the relevance score for title words to the given document. To order title words, the probability of a word sequence S, i.e. probability P(S), is computed using an n-gram statistical language model. More formally, the framework can be expressed as follows:

Let T represent a title for the document D. Since T is a word sequence, we can also write T as {tw_1, tw_2, …, tw_m} where m is the length of the title. The goal of the model is to find whether title T is suitable for the document D. In the language of probability theory, we need to compute the likelihood P(T|D), which can be expressed in the following way as suggested by Witbrock and Mittal (1999):

$$P(T \mid D) = P(tw_1, tw_2, ..., tw_m \mid D) = P(m) \prod_i P(tw_i \in T \mid D) \prod_j P(tw_j \mid tw_1, ..., tw_{j-1}) \quad (2.1)$$

As seen from the above expression, the expression for P(T|D) is expanded into three parts, namely P(m), P($tw_i \in T$|D) and P(tw_j|tw_1, …, tw_{j-1}). P(m) stands for the probability of having title length equal to m, P($tw_i \in T$|D) stands for the probability of choosing word tw_i as title words given the observation of the document D, and $P(tw_j|tw_1,\ldots,tw_{j-1})$ stands for the probability of putting word tw_j in the sequence after words $\{tw_1, \ldots, tw_{j-1}\}$. Since probability $P(tw_i \in T|D)$ helps select the title words appropriate for the content of the document D, it corresponds to the title word selection phase. On the other hand, probability $P(tw_j|tw_1,\ldots,tw_{j-1})$ helps determine the word order between the selected title words and thus corresponds to the title word ordering phase. In order to estimate which word is appropriate as a title word for the document, namely probability $P(w \in T|D)$, they simply approximate it as $P(w \in T|w \in D)$ and estimate it using Naïve Bayes approach, i.e.,

$$P(w \in T \mid D) \approx P(w \in T \mid w \in D) = \frac{P(w \in T \wedge w \in D)}{P(w \in D)} \qquad (2.2)$$

According to the above expression, we can simply count how many documents have word w in their titles and document bodies, and divide it by the number of documents containing word w in their bodies and use the ratio as the approximation for $P(w \in T|D)$. Furthermore, in their later implementation (Banko et al., 2000), both the surface strings of words and other features of words such as the part of speech tags and the position are considered for title word selection.

In order to illustrate this model better, let's look at a simple example: consider a test document with three word 'A', 'B', and 'C', and you would like to create a title with only two words. From the training corpus, we can estimate $P(w \in T|w \in D)$ for all the word 'A', 'B', and 'C'. They are: $P(\text{'A'} \in T|\text{'A'} \in D)=0.5$, $P(\text{'B'} \in T|\text{'B'} \in D)=0.3$, and $P(\text{'C'} \in T|\text{'C'} \in D)=0.2$. Meanwhile, we can estimate a bigram statistical language model for titles, which is listed in Table 2.1.

Table 2.1: A simple bigram statistical language model for titles

$P(W_2\|W_1)$	W_2 = 'A'	W_2 = 'B'	W_2 = 'C'
W_1 = 'A'	0.01	0.2	0.79
W_1 = 'B'	0.4	0.01	0.59
W_1 = 'C'	0.4	0.59	0.01

According to Equation (2.1), we can compute the most likely sequence with two words. The scores for all possible two-word sequence using word 'A', 'B', and 'C' are listed in Table 2.2. As indicated in Table 2.2, word sequence 'A C' has the highest score and should be used as the title for the test document. Interestingly, word 'B', which has the second highest score, is not included in the final title sequence. This is because even though word 'B' is scored higher than word 'C' according to the procedure of title word selection, it is used less frequently with word 'A' than word 'C'. Due to the competition between title word selection and title word ordering, the algorithm specified in (2.1) favors word 'C' instead of 'B" to be used in the title. This fact indicates that not only the procedure of title word selection is able to decide the appropriate title words but also the procedure of title word ordering.

Table 2.2: Scores for two-word sequences using words 'A', 'B', and 'C'

'W_1 W_2'	Score $=P(W_1\in T\| W_1\in D)P(W_2\in T\| W_2\in D)P(W_2\| W_1)$
'A C'	0.5*0.2*0.79=0.079
'A B'	0.5*0.3*0.2=0.03
'B A'	0.3*0.5*0.4=0.06
'B C'	0.3*0.2*0.59=0.0354
'C A'	0.2*0.5*0.4=0.04
'C B'	0.2*0.3*0.59=0.0354

One deficiency of the work by Witbrock and Mittal is on the approximation of $P(w\in T|D)$ as $P(w\in T|w\in D)$, which constrains the choice of title words to be one of the words appearing in the document and does not allow words outside the document to be used as title words. Similar to most extraction-based approaches, this limitation prevents the work from being applied to cross lingual title generation in which titles and documents are written in different languages. Furthermore, this simple approximation gives up all the word evidence in the document other than the title word itself. For example, words such as ‘basketball’, ‘volleyball’ and ‘soccer’ are very good indicators for suggesting putting the word ‘sport’ in the title, while this approximation is simply blind to that evidence. Therefore, it is important to find a better estimation for P(w∈T|D) such that not only all word evidence within document D can be used effectively for determining the title words but also the constraint that title words have to be one of the document words can be removed. In chapter 3, we will discuss different methods used for title word selection, which achieves both of these two goals.

The other issue with this model is its lack of solid theoretical analysis. The likelihood P(T|D) is expanded based on the intuition that in order to find a good title for a document, we need first select a set of title words appropriate for the content of the document and then organize them into a readable sequence. With a more careful examination on the expansion of P(T|D) in Equation (2.1), we find that there is something suspiciously wrong: The last part of the expansion, namely $\prod_j P(tw_j \mid tw_1,...,tw_{j-1})$, which is used to determine the word order, can actually be written as P(T), i.e. the probability for the word sequence T. Using P(T) to judge the quality of a word order is quite problematic because the probability for a word sequence is not only determined by the word order within the sequence but also influenced by the actual words in the sequence. Compared to a word sequence T’ with many rare words, a word sequence T with many common words have much more chance to be seen as titles for documents even though both of them are in perfect word order. As a result, titles ordered by the sequence probability P(T) tend to include unrelated common title words. Therefore, P(T) may not be a good choice for ordering title words and a more careful expansion of the likelihood P(T|D) is needed in order to obtain a better strategy for ordering title words.

In order to have an intuitive sense of why Equation (2.1) will favor the title sequence with common words, let’s consider an extremely simple case, where we have three documents in the training collection and all of them are identical to document D. Let’s assume that D contains four words ‘A B C D’. Two of the three documents in the collection have the title ‘A B’ and the other copy of D has the title ‘C D’. According to the model specified in Equation (2.1) and (2.2), we have title word selection probability P(w∈T|w∈D) as P(‘A’∈T|‘A’∈D)= P(‘B’∈T|‘B’∈D)=2/3 and P(‘D’∈T|‘D’∈D)=P(‘D’∈T|‘D’∈D)=1/3. The statistical bigram language model for titles is {P(‘A’)=P(‘B’)=2/3, P(‘C’)=P(‘D’)=1/3, and P(‘B’|‘A’)=P(‘D’|‘C’)=1}. Then, consider that you have a test document which is again identical to the document D. From the simple statistics, we know that the ratio of the probability of using ‘A B’ as the title for document D to the probability of using ‘C D’ as the title for document D, i.e., P(T=‘A B’|D)/P(T=‘C D’|D), should be 2. However, according to Equation (2.1), P(T=‘A B’|D) should be 2/3*2/3*2/3*1=8/27 and P(T=‘C D’|D) should be 1/3*1/3*1/3*1=1/27. Therefore, based on Equation (2.1), the ratio P(T=‘A B’|D)/P(T= ‘C D’|D)=8, which is four times larger than the true ratio 2.

Chapter 3

EVALUATION

Evaluating the machine-generated titles effectively and efficiently is an important aspect of automatic title generation. Relying only on human subjects to assess the quality of machine-generated titles is not efficient because the human judgments for the set of titles created by one method cannot be used for the evaluation of another set of titles that are generated for the same set of documents but using a different method. On the other hand, only using automatic methods for evaluating machine-generated titles can also be very problematic because the quality of titles, such as the readability of titles, is very hard for any computer program to judge. In this chapter, we will discuss both the automatic evaluation methods and the method of manual evaluation. Furthermore, we will examine the correlation between the two types of evaluations empirically.

In general, there are two major factors that will influence the quality of machine-generated titles:

1. Consistency: i.e., whether the machine-generated title is able to reflect the main content of the document. Since the function of a title is to provide a very brief summary for a document, a good title should be able to indicate the main points of the document clearly.

2. Readability: i.e. whether the machine-generated titles are readable to human subjects. This is the issue that distinguishes automatic title generation from the automatic key phrase extraction, which is to extract

key words from documents without having to put them into a readable sequence.

We will first discuss the automatic evaluation metrics for automatic title generation according to the factors of consistency and readability, and then examine manual evaluation.

3.1 Automatic Evaluation Metrics

The problem of how to evaluate the machine-generated summaries has been studied extensively in the literature of automatic text summarization (Turney, 1997; Goldstein, 1999; Jing et al., 1998; Mani, 1999). The main idea is to use the information retrieval metrics to measure the quality of machine-generated summaries. More specifically, by aligning the machine-generated summaries with the summaries extracted by human subjects, we can measure the precision, recall and F1 scores of machine-created summaries, and use them as the basis of evaluation. The same idea can be applied to the evaluation of machine-generated titles, namely we can compute the precision, recall and F1 score based on the number of overlapped words between human-assigned titles and machine-generated titles. Furthermore, since a title is a sequence of words, not just a bag of words, the evaluation metric should also be able to measure the difference between titles in terms of their word order. In this section, we examine two different kinds of automatic evaluation metrics. The first one is called F1 metric, which measures the difference between two titles based on the number of matched words. The second one is based on the edit distance, which measures the difference between two titles based on their word orders and the matched words.

3.1.1 F1 metric

For the consistency issue, namely to what extent the machine-generated titles are able to capture the contents of documents, we can use the word matches between the machine-generated titles and the human assigned titles as the measurement. As an analogy to information retrieval, we can treat the set of machine-selected title words as the retrieved documents in information retrieval and the set of human selected title words as the marked relevant 'documents'. Therefore, we can easily compute the 'precision' and 'recall' measurement, which have been broadly used in IR. More precisely, for automatic title generation, the 'precision' of a machine-generated title with respect to the human-assigned title is defined as the number of matched words between the machine-generated title and the human-assigned title divided by the length of the machine-generated title. Similarly, the 'recall' of a machine-generated title with respect to the human-assigned title is defined as the number of matched words between them divided by the length of the human-assigned titles.

However, similar to the situation of information retrieval, neither 'precision' nor 'recall' is good for measuring word matching. If we only consider the 'precision' metric, a simple strategy to gain the highest 'precision' is to return nothing. On the other hand, using 'recall' alone is not good either, because in the extreme case, we can return everything and ensure the highest recall. The tradeoff between 'recall' and 'precision' has been very well studied in the field of information retrieval, and people found that combinations of 'precision' and 'recall' appear to be much better metrics than the 'precision' and 'recall' alone, including mean reciprocal rank (MRR), average precision across 11 recall points, and F_β measurements. Among them, the 'F1' metric is the most popular one (Rjiesbergen, 1979), which is a special case of the F_β

measurements with β set to be 1. The definition of the ‘F1’ metric can be expressed as follows:

$$\mathrm{F1} = \frac{2 \times \mathrm{precision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recall}}$$

As indicated in the above expression, ‘F1’ gives an equal emphasis to both ‘precision’ and ‘recall’. When either the ‘precision’ or the ‘recall’ is small, the value of ‘F1’ will be small. The ‘F1’ score is high only when both the ‘precision’ and ‘recall’ are large, and will reach the maximum value 1 only when both the ‘precision’ and the ‘recall’ reach their maximum values (i.e., one). With the equal emphasis on both ‘precision’ and ‘recall’, we are able to avoid favoring titles with no words and titles with all words. Since a high ‘precision’ is related to the case that the machine-generated title has most of its words shared with the human-assigned title, and a high ‘recall’ is related to the case that most of words in the human-assigned titles appear in the machine-generated title, a high ‘F1’ value indicates that the machine-generated title shares many words with the human-assigned title and meanwhile has a title length close to that of the human-assigned title, which appears to be a desirable title.

For most experiments conducted in this thesis, we will only use ‘F1’ as the automatic evaluation metric to determine how well the machine-generated titles match with the human assigned titles, due to its simplicity and strong correlation with the human judgments, which will be discussed in a later section of this chapter.

3.1.2 Edit Distance

The metric considered in the previous subsection treats a title as a bag of words and therefore the word order of a title is ignored in the comparison.

However, titles with the same set of words can have dramatically different meaning if their word orders are different. A simple example would be when the human assigned title is ‘Bush beats Gore’ and the system comes up with a title ‘Gore beats Bush’. If we only consider the word matches between these two titles, the machine-generated title will have perfect ‘F1’ score. However, the meanings of these two titles are completely different due to the switch of the two words ‘Gore’ and ‘Bush’ in the machine-generated title. Therefore, in order to examine to what extent the machine-generated titles are able to correctly reflect the contents of documents, a better metric should not only be able to compare the number of words matched between two different titles but also take into account the order of the matched words.

A well-known type of distance between complex strings is the so-called edit distance. If S is a set of allowable edit operations on the set of all strings in question, then the edit distance between two strings A an B is the minimal number of edit operations from set S required to transform string A into string B. In the case of word sequence, the natural atomic edit operations are insertion, deletion and substitution of words. More detailed information about the edit distance can be found in (Nye, 1984).

The edit distance metric is a good candidate for such a task because it counts the number of operations that are able to transform one word sequence into another, which not only reflects the number of words matched but also how well the word order matches between titles. Consider the previous example. In terms of edit distance, to transform the machine-generated title ‘Gore beats Bush’ into the original title ‘Bush beats Gore’, we need at least two substitution operations, i.e., substituting the first word ‘Gore’ with ‘Bush’ and the last word ‘Bush’ with ‘Gore’. Since there are only three words in the title, two operations of transformation indicates that the machine-generated title is actually not good at all. As you can see, for this simple example, the edit distance metric appears to be better than the ‘F1’ metric in terms of judging the quality of machine-generated titles. However, the down side of the edit distance based metric is that all the editing operations are given equal weights, which may be not desirable. For example, consider a machine-generated title ‘Bush beats dogs’ for the same document as the title ‘Gore beats Bush’. According to the edit distance, the title ‘Bush beats dogs’ can be transformed into the true title ‘Bush beats Gore’ by only a replacement operation, i.e. replacing word ‘dogs’ with word ‘Gore’. Therefore, the title ‘Bush beats dogs’ appears to be a better title than the title ‘Gore beats Bush’. This conclusion is wrong because the title ‘Bush beats dogs’ indicates a story completely different from the true title while the title ‘Gore beats Bush’ still indicates a similar story as the true title except with the wrong conclusion. In short, the idea of using the edit distance to measure the quality of machine-generated titles has the advantage of taking into account of word orders. But the fact that each edit operation is given an equal weight may cause edit distance to give incorrect predictions for the quality of machine-generated titles. In the following experiment, we particularly measure the number of correct words in the machine-generated titles that have the same order as the original titles. For easy reference, we call it ‘CWCO’.

3.2 Human Judgments

As seen in the above section, what the automatic evaluation metrics do is to measure the difference between the machine-generated titles and the human assigned titles and use the difference to indicate the quality of machine-generated titles. However, there are at least three problems with this approach:

1) The human assigned title is only one of the possible good titles for the document. Even though the machine-generated title differs from the human assigned title quite significantly, it can still be quite close to another good unseen title. Therefore, only using the human assigned title as the comparison pattern may significantly underestimate the quality of machine-generated titles.

2) The surface distance between titles may not appropriately indicate the semantic difference between them. For example, if the original title is 'President Clinton addressed the opening ceremony' and the machine-generated title is 'President Gore addressed the opening ceremony'. Both F1 metric and the editing distance based metric (i.e. 'CWCO') will give high scores to the machine-generated title because most of the words between these two titles are matched and in the same order. However, from the human perspective, the machine-generated title is completely wrong. Thus, the surface distance may not tell the true story about the semantic difference between titles.

3) The readability of machine-generated titles. The above automatic evaluation metric can't tell whether the machine-generated is human readable or not.

Considering the limitations of automatic evaluation metrics, we believe that the judgments of human subjects are very important for evaluating the quality of machine-generated titles. Of course, because of the flexibility of titles, it could be quite difficult for an assessor to judge the quality of titles. People can have different standards for good titles and therefore different people can have quite different opinions for the same titles. For example, some people tend to favor 'clever' titles that try to catch the attention of readers and some people don’t. In our experiment, we set up very simple standards for assessors, i.e., a title is good as long as it reflects the main content of a document and is organized in a human readable way. By clearly defining the evaluation standard for the assessor, we are able to avoid the fluctuation in the judgments due to the different taste in titles. Of course, this criterion forms a low baseline for machine-generated titles. However, as will be indicated by the experiments, even with this simple criterion, most machine-generated titles are far from being good. In the experiment, each title will be assigned to one of the five categories:

- ‘Very good’ category: The title reflects the main content of the document and the word sequence is readable.

- ‘Good’ category: The title indicates the content of the document however some words are not smoothly connected with the others.

- ‘Ok’ category: The title contains important phrases however some of the word orders are not correct.

- ‘Bad’ category: Many words within the title are not related to the document even though one or two of them are important content words. The whole word sequence is totally not readable.

- ‘Extremely Bad’ category: None of the words in the title makes sense to the document.

A simple score scheme is developed with score 5 for the category ‘very good’, score 4 for ‘good’, score 3 for ‘ok’, score 2 for ‘bad’ and score 1 for ‘extremely bad’. The average score of human judgment is used as the final evaluation metric. Clearly, the higher the score of human judgment, the better the quality of machine-generated titles.

The other issue with human judgments is that it can be very expensive to obtain the human judgments both time-wise and money-wise. Therefore, in our experiment, we combine the automatic evaluation metric and the human judgments into a two-layer evaluation system: the first layer is the automatic evaluation metrics. It will filter out the approaches that lead to low evaluation numbers. The second layer consists of human judgments. Titles generated by the 'promising' approaches, i.e., approaches having a high automatic evaluation number, will be sent for human judgments. The quality of the titles generated by those 'promising' approaches will be determined by the human judgments. Therefore, most machine-generated titles are judged only by the automatic evaluation metrics and only titles generated by 'promising' approaches are evaluated using human judgments. Of course, with this two-layer scheme, there is a chance that a good method of automatic title generation may be missed. Due to the limitation of time and expensiveness, we are willing to take this risk.

3.3 Empirical Study of Evaluation Metrics

In this experiment, we would like examine three issues:

1) Are human judgments able to reflect the quality of titles correctly? As we raised the question in the last section, due to the flexibility of titles and the different taste in titles for different people, the human judgments may not be able to indicate the quality of titles reliably. Therefore, in this experiment we will test the quality of human assigned titles against the human judgments. The difference between the perfect score '5' and the averaged human judgments will be able to tell how good the reference

titles are from the viewpoint of the assessor. Meanwhile, the same set of titles with stopwords removed will also be presented for human judgments. Since the removal of stopwords should influence the readability of titles, the difference between the scores for titles with stopwords and scores for titles without stopwords will be able to indicate the sensitivity of human judgments. Furthermore, since in the rest of this thesis we will focus on creating titles without considering stopwords, this experiment will give us a sense of the loss in the quality of titles caused by throwing away stopwords.

2) Are automatic evaluation metrics able to indicate the quality of machine-generated titles correctly, and which automatic evaluation metric is more consistent with human judgments? As in the last section, we raised the question about the reliability of automatic evaluation metrics and thus we need to examine this issue empirically. In the experiment, we will apply three different title generation methods to create titles and then evaluate the machine-generated titles using the 'F1' score, 'CWCO' (the edit distance based metric) and human judgment. Essentially, we would like to examine the correlations between the automatic evaluation metrics and the human judgments because if the automatic evaluation metrics are able to distinguish good titles from bad titles to some extent, we would expect to see a positive correlation between the automatic evaluation metric and the human judgments.

3) How effective is the statistical model compared to the automatic summarization approach for generating titles? As an extension of this empirical study, we will compare the titles generated by the statistical method to the titles created by automatic text summarization using the automatic evaluation method. Compared to the automatic text

summarization approach, we feel that the statistical model is able to learn the correlation between title words and document words from training data and as a result creates better titles. In the experiment, we will see if this claim is true or not.

3.3.1 Design of Experiment

The experimental dataset comes from a CD of 1997 broadcast news transcriptions published by Primary Source Media (1997). There are a total of 50,000 documents and corresponding titles in the dataset. The training dataset is formed by randomly picking four document-title pairs from every five pairs in the original dataset. Thus, the size of training corpus was 40,000 documents with corresponding titles. The remaining 10,000 documents are used for testing. By separating training data and test data in this way, we ensure a strong overlap in the topic content between the training dataset and the test dataset, which gives the learning algorithms a chance to play a significant role in the headline generation. Finally, among the 10,000 documents used for evaluation, two hundred documents are randomly selected for the human judgments.

To obtain human judgments for machine-generated titles, a female assessor is hired. For each document, she is asked to read the whole document first before judging the titles. The human assigned titles are mixed with the machine-generated titles randomly. The female assessor works four to five hours a day and titles of 200 documents usually cost her four to five days to finish. Therefore, the averaged time for evaluating titles for each document is about 10 minutes. In order to make judgments consistent, we have her judged all the titles.

Three simple methods are used for generating titles in this experiment:

1) A Naïve Bayes approach with limited vocabulary (**NBL**). Essentially, this algorithm is the work by Witbrock and Mittal (1999), which has been described in Chapter 2. A Naïve Bayes approach is used to estimate P(tw|dw) and only the conditional probability P(tw|dw=tw) is used for selecting appropriate title words. A bigram statistical language model is used to order the chosen title words into the final sequence (Clarkson & Rosenfeld, 1997).

2) A term frequency and inverse document frequency approach (**TF.IDF**). A term frequency (i.e. TF) for a word refers to the number of times that the word appears in a document. An inverse document frequency (i.e. IDF) for a word is defined as the logarithm of the ratio of the total number of documents to the number of documents containing that word. Usually a high inverse document frequency of a word indicates that the word appears rarely in the collection. The product of these two factors, i.e. TF.IDF, measures the importance of a word related to a document (Salton & Buckley, 1988). For this simple approach, document words with the highest TF.IDF scores were chosen as the title word candidates, and a bigram statistical language model was used to order the selected words. Notice that, in the method, no length normalization was used because the length normalization factor for each word in the same document will be almost identical. Therefore, adding the length normalization to the scores of words is essentially equal to scale all the scores with a common factor, which makes no change in the final outcome.

3) A nearest neighbors approach (**NN**). This algorithm is similar to the KNN algorithm applied to topic classification in (Yang & Chute, 1994). It treats the titles in the training corpus as a set of fixed labels. For each new document, instead of creating a new title, it tries to find an appropriate

‘label’, which is equivalent to searching the training document set for the closest related document. This training document title is then used for the new document. In our experiment, we use SMART (Salton, 1971) system for indexing our training documents and test documents. The ‘ltc’ term weighting was used for weighting both the training documents and test documents, which is defined as

“l”: $\log(tf + 1)$

“t”: $\log(\frac{collection\_size}{df})$

“c”: Euclidian vector length normalization

The similarity between two documents is computed as the dot product between the two document vectors.

Finally, each method was forced to create titles with fixed 6 words, which is the average title length of training documents.

The reason we used these three methods for the examination of the automatic evaluation metrics is because they are substantially different from each other. Both the nearest neighbor approach and the NBL approach are able to learn the title-document correlation from the training corpus while the TF.IDF approach simply uses the product of the term frequency and the inverse document frequency to score document words. Meanwhile, both the NBL approach and the TF.IDF approach use a bigram statistical language model to find the appropriate word order while the nearest neighbor method simply finds the training document that is most similar to the test document and uses the title of that training document as a generated title without resorting to a statistical language model for ordering words. Because of the large variance among these three title generation methods, we may be able to find the reliable correlation between the automatic evaluation metrics and the human judgments.

3.3.2 Results & Discussions

3.3.2.1 Reliability of Human Judgments

Table 3.1 shows the averaged human judgments for the reference titles with and without stopwords. First, according to Table 3.1, the averaged score of human judgments for human assigned titles with stopwords is 4.7, which is very close to the perfect score 5. Therefore, we believe that the score of human judgments is able to reflect the quality of titles reasonably well. Secondly, by removing stopwords from the human assigned titles, the quality of those titles should degrade substantially, which is consistent with the change in the human judgments, dropping from the averaged score 4.7 to only 4. Particularly, the averaged score 4, which corresponds to the 'good' category as described before, indicates that titles without stopwords are mainly readable and able to capture the main content of the document, only with some discontinuity between phrases due to the missing stopwords. Therefore, it is still quite reasonable to study an automatic title generation system under the assumption that stopwords can be ignored. One reason for insisting on not considering stopwords for automatic title generation is because in our studies, by taking into account the stopwords, the performance of automatic title generation degrades substantially. The reason for that is because stopwords appear in almost every title and therefore they can be associated with any document word according to the statistics of the correlation between title words and document words. Thus, if stopwords are allowed, they will have more chances to be chosen as candidates for title words than other words, which may result in machine-generated titles with only stopwords. Furthermore, as will be shown in the later chapters of this thesis, the averaged scores of machine-generated titles judged by human subjects are still far from the averaged score

4.0. Therefore, how to include stopwords in automatically generated titles is not one of the main concerns of this thesis.

Table 3.1: Human judgments for reference titles with and without stopwords

	Without Stopwords	With Stopword
Human Judgments	4.0	4.7

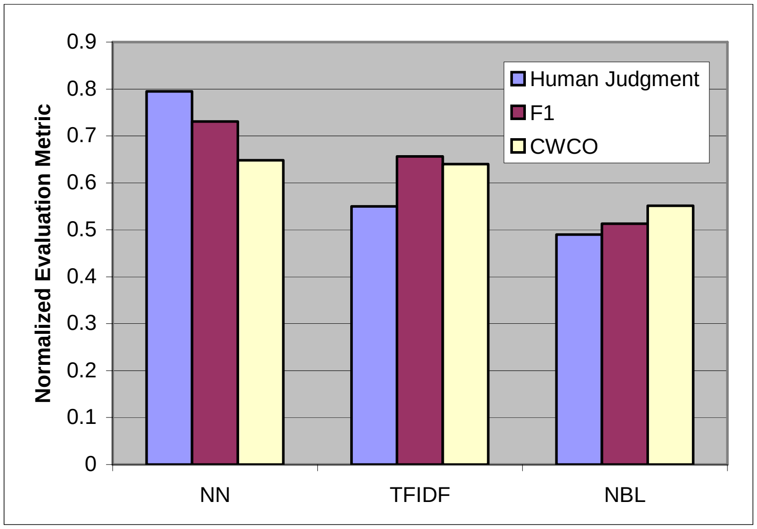

Figure 3.1: Comparison of different evaluation metrics. The three evaluation metrics are the averaged human judgments, the 'F1' score, and the CWCO metric (e.g., the edit distance based metric). Three different title generation methods are evaluated, including a nearest neighbor (NN), a term frequency and inverse document frequency (TFIDF), and a Naïve Bayes method (NBL). Notice that the vertical axis represents normalized evaluation metric, which means that each evaluation metric is scaled with a different factor in order to fit in this diagram.

3.3.2.2 Reliability of Automatic Evaluation Metrics

Figure 3.1 shows the results of three different evaluation metrics for the three title generation methods. In order to fit the three different evaluation results into a single diagram, for each evaluation metric, we use a different scale factor. More specifically, the scale factor is 3.9 for the human judgments, 0.30 for the F1 score, and 1 for the metric 'CWCO' (e.g. the edit distance based metric).

According to Figure 3.1, based on the human judgments, the nearest neighbor approach was ranked as the highest with a normalized score of 0.795, while the NBL approach was judged worst with a normalized score of 0.49 and the term frequency and inverse document frequency approach is scored as 0.55. The results from the F1 metric and 'CWCO' (i.e. the edit distance based metric), as also shown in Figure 3.1, demonstrate the same tendency, namely the nearest neighbor approach is scored the highest and the TF.IDF approach is scored higher than the NBL approach. A more careful examination of Figure 3.1 indicates that the F1 metric appears to be more correlated with the human judgments than the 'CWCO' metric. In order to see which automatic evaluation metric is more correlated with the human judgments, we compute the Pearson correlation coefficient between the two automatic evaluation metrics and the human judgments. According to the definition, the Pearson correlation coefficient is computed as following:

Consider two metrics A and B. Over a set of n objects, metric A yields measurements $x_1, x_2, ..., x_n$ and metric B yields measurements $y_1, y_2, ..., y_n$. Then, the Pearson correlation coefficient between these two metrics is computed as:

$$\text{Pearson\_Correlation(A, B)} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

The higher the Pearson correlation coefficient is, the stronger the two metrics are correlated.

According to our computation, the Pearson correlation coefficient between the metric F1 and the human judgments is 0.87 while it is only 0.71 between the metric ‘CWCO’ and the human judgments. Therefore, the metric F1 correlates with the human judgments substantially stronger than the metric ‘CWCO’. Furthermore, the metric ‘CWCO’ appears to be quite flat over the three groups of machine-generated titles, which indicates that it may be a less sensitive metric. In fact, the averaged relative change in the ‘CWCO’ metric between these three groups of titles is only 8%. For the F1 metric, the averaged relative change between the three groups of titles is about 15%, which is almost twice as large as that of the ‘CWCO’ metric. Based on these observations, we can conclude that even though both the F1 metric and the ‘CWCO’ metric appear to be positively correlated with the human judgments, the F1 metric appears to correlate with human judgments substantially more strongly than the ‘CWCO’ metric, and is more sensitive to the quality of titles than the ‘CWCO’ metric.

On the other hand, as indicated in Figure 3.1, the results for human judgments appear to show that the NN approach is noticeably better than both the NBL approach and the TFIDF approach while the results for the F1 metric and ‘CWCO’ metric appear to show that the NN approach performs similarly to the TFIDF approach and both the NN approach and the TFIDF approach perform considerably better than the NBL approach. We think one possible explanation for this discrepancy is that since the titles generated by the NN approach are actual human assigned titles for training documents, they are much more readable than titles created by the NBL approach and the TFIDF approach which are ordered by a bigram statistical language model. Therefore, the user tends to give a higher score to the titles generated by the NN approach than the other two approaches. This observation indicates that the automatic evaluation metrics cannot fully reflect human readability, which has significant influence on the human judgments. Therefore, though both automatic evaluation metrics are positively correlated with the human judgments, we still nee the human judgments for the final determination of the quality of machine-generated titles.

3.3.2.3 Comparison of Statistical Models for Title Generation to the Automatic Summarization Approach

In order to compare the effectiveness of statistical methods to the automatic summarization approach for title generation, we use the AutoSummarize function within the Microsoft Word to generate short summaries for the same 10,000 test documents. For each test document, Microsoft Word is invoked to create a summary with a specified length (e.g. 6) and the generated short summary is used as the title for the test document. For easy reference, we simply call this method 'AutoSummarize'.

The F1 scores of the NBL method and the 'AutoSummarize' method are listed in Table 3.2. Clearly, the NBL method achieves a much better F1 score than the AutoSummarize method. The fact that the 'AutoSummarize' performs poorly for title generation is consistent with the finding by Firmin and Chrzanowski (1999), who claimed that when the compression ratio of summarization is below 10%, many automatic text summarization approaches don't work well. Thus, we conclude that compared to the automatic summarization approach, the statistical model appears to be much better in catching the right title words. Since the F1 score of the automatic summarization approach is so low, we didn't bother asking the human subject to check its quality. To get more sense about the quality of titles created by the automatic summarization approach and the NBL approach, we listed five examples of machine-generated titles for both methods in Table 4.6 and 4.15. The reference titles for the same set of stories are listed in Table 4.14.

Table 3.2: F1 scores for the NBL method and the text summarization method for title generation (AutoSummarize)

	NBL	AutoSummarize
F1	0.154	0.03

3.3.3 Conclusion

In this experiment, we examined three different evaluation metrics, namely the F1 metric, the ‘CWCO’ metric (e.g. the edit distance based metric), and human judgments. First, based on the empirical results in Table 3.1, we conclude that the score of human judgments is a reliable indicator for the quality of titles. Secondly, based on results shown in Figure 3.1, both the F1 metric and the ‘CWCO’ metric are positively correlated with the human judgments. Furthermore, the F1 metric appears to correlate with the human judgments more strongly than the ‘CWCO’ metric, and appears to be more sensitive to the quality of machine-generated titles than the ‘CWCO’ metric. Therefore, we conclude the F1 is a better automatic evaluation metric than the ‘CWCO’ metric. Based on these observations, we will only use the F1 as the automatic evaluation metric in the rest of this thesis. Finally, we examine the automatic summarization approach for title generation and find that it performs significantly worse than the statistical models.

Chapter 4

TITLE WORD SELECTION METHODS

As already discussed in chapter 2, one problem with the previous work on title generation is the estimation of $P(w \in T|D)$, i.e., the likelihood of choosing word 'w' as a title word given the observation of document D. In the previous work, the likelihood $P(w \in T|D)$ is simply approximated as $P(w \in T|w \in D)$, which doesn't take into account all the words appearing in document D. Furthermore, this simple approximation restricts choices of title words to be the words in the document. This constraint can significantly limit the situations for which the title generation model can be used, such as creating English titles for foreign documents. In this chapter, we will consider other statistical methods for estimating the likelihood $P(w \in T|D)$, which can go beyond the limitation.

In order to simplify the annotations, in the rest of the thesis we will always write 'tw' to indicate a word in a title, namely $w \in T$, and 'dw' to indicate a word in a document, namely $w \in D$. For example, $P(w \in T|D)$ will be written as $P(tw|D)$ and $P(w \in T|w \in D)$ will be written as $P(tw|dw)$.

4.1 Text Categorization Method

One way of estimating probability $P(tw|D)$ is to treat every word 'tw' as a category label and interpret $P(tw|D)$ as the probability of assigning the document D to the category 'tw'. Therefore, we can apply text categorization methods to estimate the probability $P(tw|D)$. Many learning techniques have been applied to automatic text categorization, such as K nearest neighbors

(Friedman, 1994), Rochioo (Schapire et al., 1998), decision tree (Ape et al., 1998), Naïve Bayes (McCallum & Nigram, 1998), and support vector machine (Joachims, 1998). However, one thing that makes the estimation of P(tw|D) different from the standard text classification problem is that for text classification, we are only concerned with the binary answer, i.e., whether or not a document belongs to a particular category. Whereas for the estimation of P(tw|D), we really need a probability number for the likelihood P(tw|D). Therefore, many classification methods are not suitable for the likelihood estimation since they only output confidence numbers, which cannot be interpreted as probabilities. In this thesis, we consider two common methods used in text categorization: the decision tree method and the K nearest neighbor method.

4.1.1 Decision Tree Method (DT)

Decision tree is a very common machine learning method (Quinlan, 1986) and has been successfully applied to text categorization (Apte et al., 1998). The basic idea of using decision tree for estimating P(tw|D) can be stated as follows:

To understand why a title word 'tw' is used in the title, we can first examine all the documents whose titles contain the word 'tw' and then see which word in those document has the highest ability to distinguish those documents from others. In practice, the metric 'information gain' is used as the measurement of distinguishing ability. The document word with the largest information gain for the title word 'tw' will be used as the top node of the decision tree and let's call this document word 'dw_1'. After that, all the documents are separated into two sets: the set of documents with the word 'dw_1' and the set of documents without the word 'dw_1'. Then, for each of these two sets of documents, we will conduct the same search for a document word that is most informative to the title word ‘tw’, which results in the second level of the decision tree. The procedure of dividing and searching will repeated again and again until either all documents in a set belongs to a single class or there are only a small number of documents left in the set. In real implementation, the stop criteria is either 95% of documents within the divided set belong to a single class or the number of documents within the set is no more than 3. For each title word ‘tw’, we will build a separated decision tree. In the collections that we test on, the number of different title words is on the order of 10,000. Therefore we need to build around 10,000 different decision trees, which seem to be very an expensive operation. However, as will be discussed later, since most title words appear in no more than 10 documents, it is actually not expensive at all to create 10,000 different decision-trees.

Of course, in the above description of a decision tree, each document word is treated as a binary feature and the information of term frequency is ignored. Therefore, as a further improvement, we can take into account the term frequencies of document words for building decision trees. More specifically, for each node of a decision tree, we search for the most informative document word ‘dw’ as well as the most informative term frequency *t*. Then, documents are separated into two groups as follows: documents in the first group have the word ‘dw’ appearing no less than *t* time and documents in the other group have the word ‘dw’ appearing less than *t* times. In practice, we find these two versions of decision tree have almost identical performance and therefore in the later part of this thesis we will call them the ‘decision tree method for title word selection’ without distinction.

Compared to the NBL method, the decision tree method is able to look at more document words for the selection of title words. In fact, we can treat the NBL method as a simple version of decision tree: for each title word ‘tw’, create a decision tree with a single node and the document word in the top node is actually the title word itself. However, there is one serious problem with the decision tree approach. For automatic text categorization, in order to build up a reliable decision tree for each category, we usually need at least tens of examples. In the context of automatic title generation, many title words actually appear only around ten times or even less, which means that it may be difficult to build reliable decision trees for those rare title words. Furthermore, for those rare title words, the created decision trees can be considerably shallow, which may cause them to be overly favored. The extreme example would be that a word 'tw' only appears in one title among the whole training corpus. Then, any unique word 'dw' inside the corresponding document can be a good indicator for title word 'tw'. Therefore, for the test documents, the word 'tw' will surely be favored as long as those unique document words {'dw'} appear. For the broadcast news of 1997, which is the testbed described in section 3.3, 81% of the title words appear in no more than 10 training titles (the total number of training title is 40,000). In summary, even though the decision tree has the ability of taking advantage of all the words in the document, the sparse nature of title words makes many of the decision trees very shallow and therefore only a very small number of document words are actually used as evidence for selecting title words.

4.1.2 K Nearest Neighbor Method (KNN)

K nearest neighbor is an instance based learning method and has demonstrated very good performance for automatic text categorization (Yang & Chute, 1994). In order to apply the k nearest neighbor method to estimate P(tw|D), we will first compare the test document D to the documents in the training corpus and select the training documents that are most similar to the test document D. Then, by counting the occurrences of the words in the titles of those selected training documents, we can compute the title word distribution and use it as the estimation for the likelihood P(tw|D).

Following the paper by Yang (1997), we use SMART (Salton, 1971) to index our training documents and test documents along with the weight schema "ltc" (Salton et al., 1988), which is defined as follows:

"l": $\log(tf + 1)$

"t": $\log(\frac{collection\_size}{df})$

"c": Euclidian vector length normalization

The similarity between documents is computed as the dot product between document vectors.

One interesting case of the KNN method is when we only consider the training document that is most similar to the test document. In this case, we don't have to go through the procedure of word ordering. Instead, we can simply use the title of the most similar training document as the title for the test document. This is called nearest neighbor approach ('NN'). Obviously, the advantage of this method over other methods is that we don't have to rely on the statistical n-gram language model to order the title words, which has shown to be a serious problem in automatic title generation. The disadvantage of the nearest neighbor approach is that it can't produce any new titles, which is unacceptable for some tasks.

Compared to the decision tree method and the NBL method, the k nearest neighbor method is able to take advantage of all the words in documents since the document-document similarity is measured based on the word matching between documents.

4.2 Reverse Information Retrieval Method (IR)

In this section, we will discuss the idea of converting the title word selection problem, i.e. estimating P(tw|D), into a information retrieval problem. Usually, in order to estimate P(tw|D), we need to estimate P(tw|dw) for every word 'dw' in the document D and combine the estimations of P(tw|dw) together as the estimation for P(tw|D). Following this idea, each title word 'tw' can be represented as a vector of document words $< P(tw \mid dw_1), P(tw \mid dw_2),..., P(tw \mid dw_n) >$ and each element inside the vector, i.e. P(tw| dw_i), tells the correlation between title word 'tw' and document word 'dw_i'. Then, we can treat likelihood P(tw|D) as a sort of vector similarity between the representation vector for title word 'tw' and the term vector for document D. When the similarity between the term vector for document D and the representation vector for title word 'tw' is large, there will be many words 'dw' in D that are strongly correlated with the title word 'tw' (i.e. with large P(tw|dw)) and therefore word 'tw' should have a high probability of being used as a title word for document D. Clearly, this is similar to the vector model of information retrieval, where both the query and the document are represented as term vectors and the similarity between them is computed as the dot product between them. The advantage of treating the title word selection problem as a variant of information retrieval is that we can use all the techniques developed in information retrieval, such as the tf.idf term weighting scheme (Salton & Buckley, 1988) and pseudo-relevance feedback, for selecting good title words. Of course, the essential difficulty with handling the title word selection problem as an IR problem is how to compute the representation vectors for title words, which is discussed in the following subsection.

4.2.1 Vector Representation of Title Words

To find out the optimal representation vectors for title words, let's assume that we have already obtained the representation vectors for all the title words. If those vectors are good for representing title words, by applying our information retrieval model, we should be able to generate title words for documents similar to what human subjects have created. More specifically, the difference between the human assigned title words and machine-generated title words over all the training documents will be minimized. Therefore, we will search for the set of representation vectors that minimize the difference between human assigned title words and machine-generated title words.

Let N be the number of documents in the training collection, N_{tw} be the number of distinct no-stopwords in the title of the documents in the training collection and N_{dw} be the number of distinct no-stopwords in the documents in the training collection.

Let **D** be a matrix with N rows and N_{dw} columns. An element $\mathbf{d}_{ij}$ in **D** represents the number of occurrences in the i-th document of the j-th document word. Let $\mathbf{d}_i$ be the i-th row vector in **D** (of length N_{dw}). The row vector characterizes the i-th document.

Let **T** be a matrix with N rows and N_{tw} columns. An element $\mathbf{t}_{ij}$ in **T** is the number of occurrences in the title of the i-th document of the j-th title word. Let $\mathbf{t}_i$ be the i-th row vector in T (of length N_{tw}). The row vector $\mathbf{t}_i$ characterizes the title of the i-th document.

In the standard Information Retrieval paradigm, a query **q** is represented by a row vector of length N_{dw} and a document **d** is represented by a row vector of length N_{dw}. The numbers in these vectors represent the weights of the corresponding words in the given query and document. The strength of the match between the query **q** and the document **d** is given by the inner product $\mathbf{qd}^{\mathrm{T}}$. When we issue a query to a search engine, we get back a list of documents with their similarity scores, which are the inner products $\mathbf{qd}_1^{\mathrm{T}}$, $\mathbf{qd}_2^{\mathrm{T}}$, ..., $\mathbf{qd}_N^{\mathrm{T}}$ between the given query **q** and the documents in the collection $\mathbf{d}_1$, $\mathbf{d}_2$, ..., $\mathbf{d}_N$. More concisely, we can introduce a score vector **s** of length N and define **s** as $\mathbf{s}=(\mathbf{qd}_1^{\mathrm{T}}, \mathbf{qd}_2^{\mathrm{T}}, ..., \mathbf{qd}_N^{\mathrm{T}})$, or $\mathbf{s}= \mathbf{qD}^{\mathrm{T}}$.

We wish to adapt this paradigm to assign title words to documents. To do so, we need to represent every title word by a row vector of length N_{dw}, where the numbers in this vector represent the strength of the connection between each document word and the given title word. We will introduce a matrix **M** for this purpose.

Let **M** be a matrix with $\mathrm{N_{tw}}$ rows and $\mathrm{N_{dw}}$ columns. An element $\mathbf{m}_{ij}$ in **M** is an estimate of the strength of the connection between the i-th title word and the j-th document word. Let $\mathbf{m}_i$ be the i-th row vector in M. The row vector $\mathbf{m}_i$ is a vector of length $\mathrm{N_{dw}}$ and represents the i-th title word as a vector in document word space. Later, we will show how to calculate **M** from **D** and **T**.

Let $\mathbf{d}_{\mathrm{test}}$ be the document vector (of length N_{dw}) for a document taken from the testing set. Our goal is to select title words for $\mathbf{d}_{\mathrm{test}}$. We treat the document $\mathbf{d}_{\mathrm{test}}$ as if it were a query vector in information retrieval, and we produce a list of title words with their scores, where the scores are the inner products $\mathbf{d}_{\mathrm{test}}\mathbf{m}_1^{\mathrm{T}}$, $\mathbf{d}_{\mathrm{test}}\mathbf{m}_2^{\mathrm{T}}$, ..., $\mathbf{d}_{\mathrm{test}}\mathbf{m}_{Ntw}^{\mathrm{T}}$ between the given document $\mathbf{d}_{\mathrm{test}}$ (analogous to **q**) and the title words $\mathbf{m}_1$, $\mathbf{m}_2$, ..., $\mathbf{m}_{Ntw}$ (analogous to $\mathbf{d}_1$, $\mathbf{d}_2$, ..., $\mathbf{d}_N$). We take the top K title words with the highest scores as our chosen title words for the given document $\mathbf{d}_{\mathrm{test}}$. Similar to the treatment in information retrieval, we simplify the score expression using matrix multiplication. Let **s** be the score vector of length N_{tw} and **s** is defined as $\mathbf{s}=(\mathbf{d}_{\text{test}}\mathbf{m}_1^T, \mathbf{d}_{\text{test}}\mathbf{m}_2^T, ..., \mathbf{d}_{\text{test}}\mathbf{m}_{Ntw}^T)$, or $\mathbf{s}=\mathbf{d}_{\text{test}}\mathbf{M}^T$. The i-th element in the score vector **s** is the score of the i-th title word for the test document $\mathbf{d}_{\text{test}}$.

Therefore, for any document i in the training collection, the corresponding title word score vector $\mathbf{s}_i$ (of length N_{tw}) can be written as $\mathbf{s}_i=\mathbf{d}_i\mathbf{M}^T$. We measure the error in the score vector $\mathbf{s}_i$ for the document i in the training collection as the sum of the squares of the differences between the author's title words $\mathbf{t}_i$ and the machine generated title words $\mathbf{s}_i$, i.e.

$$err = \sum_{i=1}^{N} \| \mathbf{t}_i - \mathbf{s}_i \|_2 = \sum_{i=1}^{N} \| \mathbf{t}_i - \mathbf{d}_i\mathbf{M}^T \|_2 \qquad (4.1)$$

where the $\| \ \|_2$ represents the Euclidean vector length, i.e. the sum of the squares of all the numbers in the vector.

We are trying to minimize the difference between the matrix **T** (N rows and N_{tw} columns) of authors' titles and the matrix $\mathbf{DM}^T$ (N by N_{dw} times N_{dw} by N_{tw}) of mechanically assigned titles. In principle, it is possible to optimize **M** to minimize the error function *err*, using the Singular Value Decomposition (SVD) (Hildebrand, 1952). However, for large N_{tw} , N_{dw} and N, it will be very expensive to find the matrix **M** that minimizes the error function in Equation (1), even with Singular Value Decomposition (SVD) package for sparse matrices (Press et al., 1993). Therefore, to avoid the computational complexity, we change the objective function to maximize the similarity between the author assigned title $\mathbf{t}_i$ and the machine generated title $\mathbf{s}_i$, i.e. $\sum_{i=1}^{N} \mathbf{t}_i \mathbf{s}_i^T$. Then, the approximated error function, named *err'* is changed to

$$err' = -\sum_{i=1}^{N} \mathbf{t}_i \mathbf{s}_i^T = -\sum_{i=1}^{N} \mathbf{t}_i (\mathbf{d}_i \mathbf{M}^T)^T = -\sum_{i=1}^{N} \mathbf{t}_i \mathbf{M} \mathbf{d}_i^T \qquad (4.2)$$

However, there is one problem with minimizing the approximated error function *err'*. Since the error function *err'* is linearly dependant on the matrix **M** and there is no constraint on the matrix **M,** the error function *err'* will have no lower bound. To avoid the case that the error function *err'* goes to negative infinity, we enforce the Euclidean length of the title word representation vector $\mathbf{m}_\mathrm{i}$ to be 1, i.e. $\|\mathbf{m}_i\|_2 = \sum_{j=1}^{N_{dw}} \mathbf{m}_{ij}^2 = 1, \ \forall i$.

The optimal solution **M** that minimizes the error function *err'* in Equation (4.2) and also satisfies the set of constraints in the above, can be found using the method of undetermined Lagrangian multiplier (Hildebrand, 1952). For all values of $\mathbf{m}_{ij}$, we set to 0 the partial derivatives with respect to $-\sum_{i=1}^{N} 2\mathbf{t}_i \mathbf{M} \mathbf{d}_i^T + \sum_{i=1}^{N_{tw}} \lambda i \sum_{j=1}^{N_{dw}} \mathbf{m}_{ij}^2$, which results in the solution for $\mathbf{m}_{ij}$ as

$$\mathbf{m}_{ij} = \frac{\sum_{k=1}^{N} \mathbf{t}_{ki} \mathbf{d}_{kj}}{\sum_{j=1}^{Ndw} \left\{ \sum_{k=1}^{N} \mathbf{t}_{ki} \mathbf{d}_{kj} \right\}^2} \qquad (4.3)$$

Equation (4.3) gives the analytic solution of matrix **M** that is able to minimizes the difference between the human assigned title words and the machine selected title words. The i-th row of the matrix **M** corresponds to the representation vector for the i-th title word.

4.2.2 Procedures for Retrieving Good Title Words

With the representation vectors computed in the previous subsection, we can now apply information retrieval methods to find the good title words for documents, which can be further divided into two steps:

- Weight the title word representation vectors $\mathbf{m}_i$. Since we treat each "title word" as a "document" in information retrieval, we can view the whole set of representation vectors for title words as a "document collection" in information retrieval. Thus, the standard term weighting scheme used in information retrieval can be applied directly to weight representation vectors for title words. In our experiment, we use 'ltc' term weighting scheme within the 'SMART' system that has been described in the section 4.1.2.

- Use the standard information retrieval system 'SMART' to compute the similarity between the test document and the representation vector for each title word 'tw'. Then, the similarity is used as the estimation for P(tw|D). Similar to most information retrieval system, stopwords are removed and words are stemmed using the porter algorithm (Porter, 1980).

4.2.3 Discussion

Compared to the Naïve Bayes method and the decision tree method, this algorithm has the advantage of being able to use all the document words as evidence for selecting title words because a title word is considered to be a good candidate for a document only when the whole representation vector of that title word matches well with the document vector.

Furthermore, this algorithm benefits from the optimized representation vectors for title words and the tuned term weighting schemes in information retrieval:

- With the optimized representation vectors for title words, we are able to weight the correlation between title words and document words correctly, which is crucial to the title word selection task.

- Good term weighting schemes (Salton & Buckley, 1988; Jones & Willett, 1997), such as TF.IDF and their variants, have been carefully crafted to take into account the factors of the word frequency within a document (i.e., TF), the word frequency within the collection (i.e, DF), and the document length (i.e. normalization factor). This algorithm is going to benefit from the term weighting schemes of information retrieval in two ways: First, the TF.IDF term weights usually reflect the importance of a term related to a document. In this algorithm, title words are represented as vectors of document words. By using TF.IDF term weights to weight the document words in representation vectors, we promote the connection between the important document content words and title words, and de-emphasize the connection of the trivial document words with title words. Secondly, in information retrieval, the normalization factor in term weighting schemes avoids the takeover of long documents. In this algorithm, the normalization factor helps us overcome the noises introduced by common title words. According to this algorithm, most numbers in the representation vectors for common title words will be nonzero because common title words co-occur with most of the document words. Thus, the common title words are similar to the 'long documents' in information retrieval. Without the normalization factor, these common title words will always be chosen because their representation vectors usually have very large overlap with the test document vector. With the help of the normalization factor, the numbers in the representation vectors for common title words will be scaled down substantially and the chance for common title words to be selected will decrease dramatically.

4.3 Expanding P(tw|D)

According to the above discussion, two issues influence the estimation of P(tw|D) significantly: on one hand, we would like to use all the words in the documents as the evidence for selecting good title words. Methods such as Naïve Bayes suggested by Witbrock and Mittal (1999) and decision tree have the problem of only using a very small number of words in the document for determining title words. On the other hand, we need the flexibility in choosing the title words individually. Methods such as k nearest neighbor have the disadvantage that all the words in the titles of most similar documents will be chosen as title word candidates. In this section, we will introduce a general method for estimating P(tw|D) that has advantages on both aspects.

In order to take into account all the words within the document, we can expand the likelihood P(tw|D) as following:

$$P(tw \mid D) \approx \sum_{dw \in D} P(tw \mid dw) P(dw \mid D) \approx \frac{1}{|D|} \sum_{dw \in D} P(tw \mid dw) tf(dw, D) \qquad (4.4)$$

As indicated by the above equation, P(tw|D) is expanded as a weighted sum of probabilities P(tw|dw), i.e., the probability of putting word ‘tw’ on a title given that word ‘dw’ appears in document D. Therefore, the choice of title word ‘tw’ is collectively decided by all the words in the documents. Of course, using tf(dw,D)/|D| as an estimation for P(dw|D) is problematic. Particularly, it may over-count the contribution of common document words, which should have little to do with the selection of title words. This issue will be brought up again when we discuss the ‘dual noisy channel model’ for title generation. For now, let's only consider the form of P(tw|D) in Equation (4.4). Clearly, the critical issue with this expansion is how to estimate the conditional probability P(tw|dw). In the following subsections, we will introduce two different methods, namely a Naïve Bayes approach and a statistical translation approach, for title word selection.

4.3.1 A Naïve Bayes Approach (NBF)

One simple idea for estimating P(tw|dw) is to expand it using the Bayesian rule, i.e.

$$P(tw \mid dw) = \frac{P(tw \wedge dw)}{P(dw)} \tag{4.5}$$

According to the above expression, in order to estimate P(tw|dw), we can simply count how many training examples have both word 'tw' in their titles and word 'dw' in their document bodies, and divide it by the number of documents with word 'dw' in their document bodies.

Clearly, this simple approach can be treated as a natural extension of the Naïve Bayes approach suggested by Witbrock and Mittal (1999), where only P(tw|dw=tw) is used for title word selection. The advantage of this method over the NBL method is that, this method allows any word to be used as a title word that is not necessarily a word in the original document. For future reference, we call this method a 'Naïve Bayes approach with a full vocabulary' (NBF).

One problem with this simple approach is that, even though we observe that word 'tw' appears in the title and word 'dw' appears in the document, these two words still may not be correlated with each other because the appearance of 'tw' in the title may be caused by document words other than 'dw'.

Therefore, by counting each co-occurrence of a title word with a document word equally, this simple approach can have significant incorrect counts of observed evidence. As a result, this defect may cause the algorithm to overly favor common title words because common title words will co-occur frequently with many different document words and therefore, according to this algorithm, they tend to have strong association with many document words. One simple way to see why the Naïve Bayes method in (4.5) can overly estimate P(tw|dw) is to examine the sum of P(tw|dw) over all title words 'tw'. Theoretically, the sum should be one. But in the case of using Naïve Bayes estimation for P(tw|dw), the sum can exceed one. To see this more clearly, let's consider a collection with a single document whose title is 'A B'. Therefore, for any word 'dw' in the document, according to Equation (4.5), both P(A|dw) and P(B|dw) is one. Then the sum of P(A|dw) and P(B|dw) is actually two, which is against the law of probability.

Of course, this issue may be less significant in the case of the NBL method, where only P(tw|dw=tw) is used for selecting title words. But, it still does exist. For example, consider a collection with two sets of documents. One set of documents is about the NBA games and therefore the name 'Michael Jordan' appears frequently in both the titles and the documents. The other set of documents is about the research progress in machine learning. Since Prof. Michael I. Jordan from Berkeley has been the top researcher in this field, many of the articles mentioned his name in their documents but not in their titles. Clearly, the reason why the word 'Jordan' is used in titles is not only because the corresponding documents contain the word 'Jordan' but also because those documents have words such as 'NBA' and 'Wizard'. If we apply the NBL method for title word selection, it will always rank the word 'Jordan' highly as long as the word 'Jordan' appears in documents no matter what the contents of the documents.

Based on the above analysis, a better algorithm for estimating P(tw|dw) needs to account for the fact when the appearance of a title word ‘tw’ can be explained very well by some document words, the leftover document words should only get a small amount of credit for the appearance of the word ‘tw’ in the title. In the next subsection, we will introduce the statistical translation method for title word selection suggested by Kennedy and Hauptmann (2000), which has the effect of letting the document words compete with each other for the credit of explaining title words.

4.3.2 A Statistical Translation Approach (ST)

The idea of applying statistical translation model to the title generation task is proposed in paper by Kennedy and Hauptmann (2000). It can be intuitively understood as follows: since documents are generally quite verbose in describing information while titles usually are much more concise, they show very different characteristics in using languages. Therefore, we can think of documents and titles as files with similar information but in different languages. More precisely, we can view documents as information written in a ‘verbose’ language and titles as in a ‘concise’ language. Then, every document-title pair in the training corpus can be treated as a translation pair and the process of creating a title from a document can be viewed as the process of translating a document (i.e., information in ‘verbose’ language) into a title (i.e., information in ‘concise’ language). With this concept in mind, we can interpret the conditional probability P(tw|dw) as a word ‘translation’ probability, i.e., the probability of translating a word ‘dw’ in the ‘verbose’ language into a word ‘tw’ in the ‘concise’ language, which can be estimated using the IBM statistical translation model (Brown et al., 1993).

The goal of the translation model is to find the optimal set of word 'translation' probabilities P(tw|dw) that can explain the document-title pairs in the training corpus. Formally, we can write the objective goal as follows:

$$M^* = \arg\max_{M} \prod_{i} P(T_i \mid D_i, M) \tag{4.6}$$

where M stands for the model which includes the set of word translation probabilities {P(tw|dw)} and the likelihood P(T|D,M) stands for the probability of translating document D into title T using model M. In the IBM statistical translation model I (Brown et al., 1993), the likelihood P(T|D,M) is expanded as follows:

$$P(T \mid D, M) = \frac{\varepsilon}{(|D|+1)^{|T|+1}} \left(\sum_{dw \in D} P(tw \mid dw) tf(dw, D) + P(tw \mid \phi) \right) \tag{4.7}$$

where ε is a constant which accounts for the uncertainty of title length. Symbol ϕ stands for the null word, which accounts for the words in title that can't be explained well by the words in documents.

The optimization of Equation (4.6) can be accomplished using the Expectation-Maximization algorithm (EM) (Dempster et al., 1977). More interestingly, unlike other cases where the EM algorithm usually results in a local maximum, for the objective function stated in Equation (4.6), the EM algorithm is guaranteed to find the global maximum. The detailed proof can be found in the literature (Brown et al., 1993). Here, we just state the resulting EM updating equations:

$$P'(tw \mid dw) = \frac{1}{Z(dw)} \sum_{\{D \mid dw \in D \wedge tw \in T\}} \left(\frac{P(tw \mid dw) tf(dw, D)}{\sum_{dw \in D} P(tw \mid dw) tf(dw, D) + P(tw \mid \phi)} \right) \qquad (4.8)$$

where Z(dw) is the normalization factor for word 'dw' which can be expressed as follows:

$$Z(dw) = \sum_{tw} \sum_{\{D \mid dw \in D \wedge tw \in T\}} \left(\frac{P(tw \mid dw) tf(dw, D)}{\sum_{dw \in D} P(tw \mid dw) tf(dw, D) + P(tw \mid \phi)} \right) \qquad (4.9)$$

The interesting part of this method is the introduction of competition among different document words. By looking at Equation (4.8), we can see that word 'translation' probability P(tw|dw) is expressed as the sum of contributions from all the instances where document word 'dw' and title word 'tw' co-occur together, i.e. $\frac{P(tw \mid dw) tf(dw, D)}{\sum_{dw \in D} P(tw \mid dw) tf(dw, D) + P(tw \mid \phi)}$. Unlike the Naïve Bayes approach stated in previous subsection where each co-occurrence instance is counted equally, the translation model lets different document words compete with each other for the share of co-occurrence instances. By dividing $P(tw \mid dw) tf(dw, D)$ by the sum $\sum_{dw \in D} P(tw \mid dw) tf(dw, D) + P(tw \mid \phi)$, we will give more shares of the co-occurrence instance to those document words that are strongly associated with the title word 'tw' and vice versa. Therefore, this method is able to reduce the amount of incorrect counting of evidence in the Naïve Bayes method. As will be shown later in the experiment, this method achieves substantially better performance than the Naïve Bayes method.

4.4 Empirical Study

In this section, we will examine empirically the effectiveness of different methods for title word selection. First, we will examine the Naïve Bayes method suggested by Witbrock and Mittal (1999) (NBL) in two aspects: 1) whether the Naïve Bayes estimation is good for P(tw|D), and 2) whether the constraint that any document word can only suggest itself as a title word is necessary. Then, a comparison of all six different methods for title word selection will be presented.

Similar to the setup in section 3.3, we use the CD of 1997 broadcast news transcriptions as the testbed. The training corpus consists of 40,000 document-title pairs and the test corpus consists of 10,000 document-title pairs. Different title word selection methods are used to compute the scores for title words. For each document, the first 100 title words with highest scores are selected, and a bigram statistical language model is used to create a sequence with six words that are among the 100 selected title words. The reason for choosing title length to be six is because that is the averaged title length of the training documents. Stopwords in all the titles are removed. The F1 score is used as the automatic evaluation metric because of it strong correlation with the human judgments as described in Chapter 3.

4.4.1 Examination of Naïve Bayes Method with Limited Vocabulary

In Chapter 2, we introduced the Naïve Bayes approach with a limited vocabulary (NBL) for title word selection, which approximates the likelihood P(tw|D) as P(tw=dw|dw). As you can see from the description of this algorithm, two factors can influence the performance of this title word selection method:

1) The constraint that only P(tw|dw=tw) is used for selecting title words. This constraint prevents document words other than word ‘tw’ from being used as the evidence for determining word ‘tw’ as a title word. Furthermore, this constraint keeps any words outside the document from being used as title words.

2) Use the Naïve Bayes method for the estimation. As already pointed out before, one problem with the simple Naïve Bayes method for estimating P(tw|dw) is that it may give an incorrect count of the co-occurrence events between document words and title words and therefore result in a unreliable estimation.

To clearly see the effects of these two factors on the task of title word selection, we compare this algorithm to the other two methods, namely a Naïve Bayes approach with a full vocabulary (NBF) and the term frequency and inverse document frequency approach (TF.IDF). These two methods have been described in Section 4.3.1 and Section 3.3.1, respectively. Compared to the NBL method, the NBF method uses the same Naïve Bayes method for estimating P(tw|dw) but with the consideration of all the words in the document. Meanwhile, the TF.IDF method has the same constraint as the NBL method, namely a document word can only suggest itself as a title word, however with tf.idf scores for scoring words. Therefore, by comparing the NBL method to the NBF method and the TF.IDF method, we can see the effects of the two factors.

Table 4.1: F1 scores for title word selection methods NBL (the Naïve Bayes approach with a limited vocabulary), NBF (a Naïve Bayes approach with a full vocabulary) and TF.IDF (using tf.idf for scoring title word candidates).

	NBL	NBF	TF.IDF
F1	0.154	0.04	0.197

Table 4.1 shows the ‘F1’ results for these three methods. Surprisingly, the NBF method performs much worse than NBL. The difference between the NBF and NBL method is that the NBL method assumes a document word can only generate itself as the title word while the NBF method allows a document word to suggest any title word. Because of the rough nature in the Naïve Bayes estimation, the simple constraint of the NBL method actually makes the results much more stable than the NBF method. By examining the titles generated by the NBF method, we notice that most of the title words in the NBF-generated titles are actually the common title words. As an example, the popular word ‘clinton’, which appears in the 6% of training titles, is actually used by 66% of the titles generated by the NBF method. In table 4.7, examples of titles generated by the NBF method are listed. Clearly, almost all the titles created by the NBF method are simply those most common title words. Furthermore, as will be shown later, by replacing the Naïve Bayes estimation with the estimation by the statistical translation model for P(tw|dw), we are able to achieve a much better performance than the original NBF method (see Table 4.3 for the ST method). Based on these two observations, we can see that the Naïve Bayes method is not a good estimator for P(tw|dw) and the success of the NBL method is due to the constraint that only P(tw|dw=tw) should be used for selecting title words.

The other surprising observation from Table 4.1 is that the simple TF.IDF method appears to work even better than the NBL method. The difference between the NBL method and the TF.IDF method is that, the TF.IDF method uses the tf.idf metric to score document words while the NBL method uses the the Naïve Bayes estimation for P(tw|dw=tw). Intuitively, the NBL approach takes advantage of the training corpus for estimating the correlation between title words and document words, and therefore should create more appropriate title words than the TF.IDF method. Since the title word selection method is intertwined with the bigram statistical language model for generating sequences, it may be due to the inference with the language model. Therefore, in addition, we compare the NBL method to the TF.IDF method by simply looking at the top six title words ranked by both algorithms. The ‘F1’ scores of both algorithms for the case of using and not using a statistical language model for word ordering are listed in Table 4.2.

Table 4.2: F1 scores for the NBL method and the TF.IDF method. The first row corresponds to the case when no word ordering procedure is applied and the second row corresponds to the case when a bigram statistical language model is used for ordering words.

F1	TF.IDF	NBL
Without word ordering	0.199	0.215
With word ordering	0.197	0.154

As indicated in Table 4.2, when we only look at the top title words selected by both methods, the NBL method did slightly better than TF.IDF. However, after applying the n-gram language model to order words, the F1 score for NBL method drops significantly while for the TF.IDF method, the F1 score appears to be almost unchanged. This comparison suggests that, even though the NBL method did a reasonable job of scoring title words on the top of the rank list, it may do poorly for the title word candidates that are not in the top. In Figure 4.1, we plot the distribution of normalized scores over different ranks for both the NBL method and the TFIDF method. The normalized score is defined as the original score of a title word divided by the score of the title word ranked as #1. According to Figure 4.1, the distribution of normalized scores for the NBL method drops much more quickly than that of the TFIDF method. Therefore, for title word candidates in the tail of the rank list, the NBL method always gives them similarly small values. This fact implies that the NBL method may have a poor estimation for words that are not in the top of the rank list. Since a bigram statistical language model is used to make a smooth sequence out of the selected title words, it can pull out words in the tail of the rank list, and words on the top of the rank list may not be used in the sequence. Therefore, not only do the scores of words on the top of the rank list matter but also the scores of words in the tail. Based on this analysis, we think that the reason why the NBL method does poorly after the title words are ordered may be attributed to the fact that the NBL method may have a poor estimation for words in the tail of the rank list.

As the summary of this experiment, we conclude that the success of the NBL approach is due to the constraint that any word in the document can only suggest itself as a title word. Furthermore, based on the fact the NBF method fails miserably in selecting good title words and the NBL method performs worse than the simple TFIDF method, we find that the Naïve Bayes estimation for P(tw|dw) appears to be problematic. To show visually how these methods are different from each other, we include 5 samples of titles generated by these three methods in Table 4.6-4.8, respectively. For the purpose of reference, we listed the author-assigned titles in Table 4.14.

4.4.2 Comparison of Title Word Selection Methods

In the subsection, we will examine the effectiveness of six different title word selection methods that have been discussed before. They are:

1) A Naïve Bayes approach with limited vocabulary (NBL), which has been described in Chapter 2.

2) A decision tree approach (DT) as described in section 4.1.1. For each title word 'tw', a decision tree is built with the stop criterion that either at least 95% of the documents under a node are 'pure' (e.g., either all 95% of the documents under that node have title word 'tw' or don't) or the number of

documents under that node is no more than 3. Non-binary decision trees are built in the similar way except term frequencies are also used for splitting documents. The possible term frequencies that can be used for splitting a document set is {1, 2, 4, 6, 8, >8}. Since both binary decision tree and non-binary decision tree have the identical performance, we simply refer them as a decision tree approach.

3) A k nearest neighbor approach (KNN) as described in section 4.1.2. In the experiment, we take the top five training documents that are most similar to the test document and use them to create a new title for the test document. More specifically, the word distribution in the titles of the top five most similar documents are used to estimate P(tw|D).

4) A nearest neighbor approach (NN) as described in section 3.3.1. This is the KNN approach with K=1. Furthermore, unlike most other approaches, which rely on a bigram statistical language model to order selected title words into sequences, this method simply uses the title of the training document that is most similar to the test document as the generated title.

5) The reverse information retrieval approach (IR) as described in Section 4.2. A representation vector for each title word is computed using Equation (4.3). Then, the similarity between the representation vector of each title word ‘tw’ and the term vector of test document D is computed as the estimation for P(tw|D).

6) A statistical machine translation approach (ST) as described in 4.3.2. A EM algorithm is used for computing P(tw|dw) according to Equation (4.8) and (4.9).

For all these six methods, the top 100 title words with the highest scores of P(tw|D) are selected and a bigram statistical language model is used to create a title sequence with six words from the 100 selected title words. We tried out various number of top selected title words, ranging from 100 to 300 and all of them turn out to have almost identical F1 scores. Therefore, in this experiment and all the experiments in the later part of this thesis, we will always use the top 100 selected title words as the candidates of title words. Table 4.3 lists the F1 results of the six different title word selection methods for both the case of using the word ordering procedure and the case of not using the word ordering procedure.

Table 4.3: F1 scores for six different title word selection methods, namely the Naïve Bayes approach with limited vocabulary (NBL), the decision-tree approach (DT), the K nearest neighbor (KNN), the nearest neighbor approach (NN), the reverse information retrieval approach (IR) and the statistical translation approach (ST). Results for cases of using word ordering and not using word ordering are listed.

F1	NBL	DT	KNN	NN	IR	ST
w.o ordering	0.215	0.201	0.220	0.219	0.235	0.228
w.i ordering	0.154	0.187	0.212	0.219	0.220	0.223

First, comparing the results for the case of using the word ordering procedure to the case of not using the word ordering procedure, we can see that the F1 scores of all the methods degrade except for the nearest neighbor method, which doesn't use the procedure of title word ordering. Among them, the NBL method suffers the most significant degradation, as already discussed in the previous section. For all other methods, the degradations are less significant.

Secondly, for the case of not using title word ordering, the six different methods perform similarly. The reverse information retrieval method (IR) achieves the best performance with F1 = 0.235 and the statistical translation method has the second best performance with F1= 0.228. In contrast, for the case of using the word ordering procedure, the difference between the six different approaches are quite significant. The DT approach and the NBL approach perform substantially worse than the other four methods. The four methods, namely the KNN method, the NN method, the reverse IR method and the ST method, achieve similar performance, ranging from 0.212 to 0.223. The ST method performs slightly better than the other three methods. The reason that the NBL approach and the decision tree approach perform worse than the other four methods is because both these two approaches only use a small subset of words in the document as the evidence to determine the title words, while the other four methods are able to examine all the words in the document. By examining decision trees for title words, we find that 78% of decision trees have no more than 3 layers. In order to see the difference between these two set of approaches further, we plot the distribution of normalized scores over different ranks for five methods in Figure 4.1 (the NN method doesn’t score title words). The normalized score is defined as the score of a title word candidate divided by the score of the title word ranked as #1. As indicated in Figure 4.1, we can see two different types of behaviors: for the methods NBL and DT, the normalized score drops very quickly and title word candidates in the tail of the rank list have extremely small scores. Whereas for the methods KNN, ST and IR, the normalized scores over different ranks change much more smoothly and the title word candidates in the tail can still have reasonably large scores. This is consistent with the fact that the NBL method and the Decision Tree method only consider a small number of evidence for selecting title words, which makes the scores for title words in the tail of the rank list significantly smaller than that of title words on the top of the list. For the methods KNN, ST and IR, they are able to use all words in a document as evidence to compute the scores of title words and therefore even the title words in the tail of the rank list can still have reasonably large values.

Based on the above observations, we can conclude that a title word selection method that is able to use many document words as evidence for selecting title words will usually result in a better F1 score. Of course, we should be careful with this statement. Recall that in the previous section we found that the NBF method performs extremely poorly compared to other methods. Even though both the NBF method and the ST method have similar combination form in terms of taking into account the opinion of all the document words (compare Equation (4.7) with Equation (4.4)), the ST method performs significantly better than the NBF method. This phenomenon can be explained by their different ways of estimating the conditional probability P(tw|dw) as already discussed in section 4.3.2. For the NBF method, the likelihood P(tw|dw) is computed without taking into account the correlation between different document words. Whereas, for the ST method, as explained in section 4.3.2, the competition between different document words has been considered in the EM algorithm. Therefore, the ST approach gives a better estimation for P(tw|dw) than the NBF method, which leads to better titles. Again, the fact that the ST method outperforms the NBF method indicates the importance in estimating P(tw|dw).

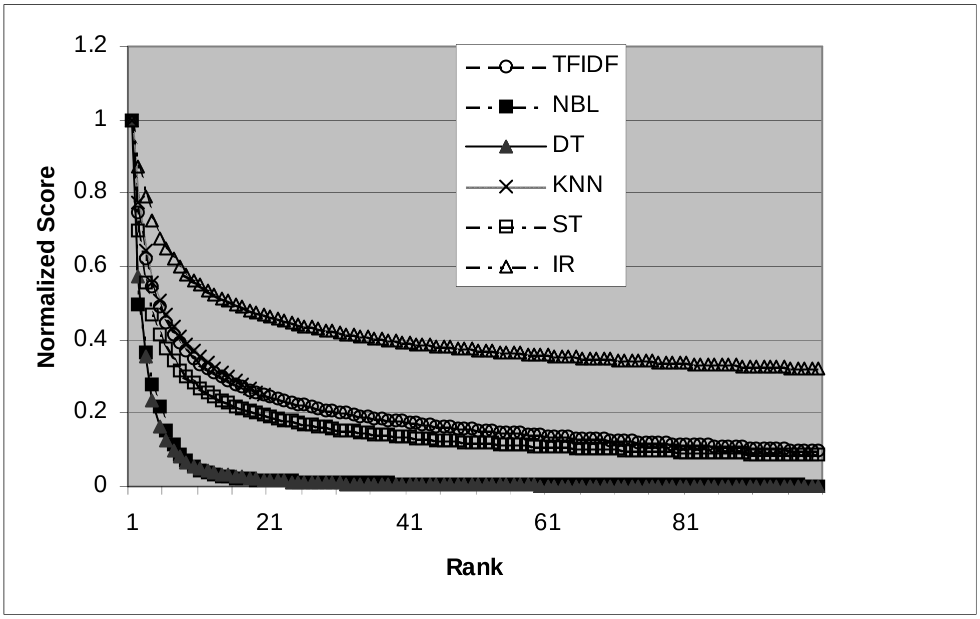

Figure 4.1: The distribution of normalized scores for different ranks of selected title words for methods KNN, IR, NBL, ST, TFIDF and DT. The normalized score is defined as the score of a title word candidate divided by the score of the #1 ranked title word.

The other interesting observation is based on the comparison of the F1 score of the KNN method to that of the NN method. Even though the KNN method composes the title for the test document using the top five most similar training documents, it doesn’t outperform the nearest neighbor (NN) approach, which only looks at the top one most similar training document. On the contrary, the KNN performs slightly worse than the NN approach. To further investigate the influence of different K values on the quality of generated titles, we did experiments with K=3, 4, 5, 10, 20 and the F1 scores for those KNN approaches are listed in Table 4.4. As indicated from Table 4.4, increasing the value of K doesn’t help the F1 score. In contrast, when K is set to be large, such as K=20 in Table 4.4, the F1 score drops substantially. First, the reason that the NN approach outperforms the KNN approach can be explained by the fact that the nearest neighbor approach doesn’t rely on a bigram statistical language model to form word sequences. As indicated by the comparison of F1 scores between the case of using word ordering and the case of not using word ordering in Table 4.3, we can see that usually the word ordering procedure is going to bring down the F1 score. Therefore, for the KNN approach, due to the word ordering procedure by a bigram statistical language model, it doesn't improve the F1 score. The other explanation could be that the top few retrieved documents are much more similar to the test document than other retrieved documents. In order to see if this is correct, we plotted the distribution of the normalized scores of retrieved documents over different ranks in Figure 4.2. The normalized score is defined as the score of a retrieved documents divided by the score of the ranked 1 retrieved document. According to Figure 4.2, the normalized score of retrieved documents drops much faster at the beginning of the distribution than at the tail. Since a similarity score for a document implies the relevance of that document to the test document, from Figure 4.2, we can infer that most relevant documents probably concentrate at the top of the retrieval list and therefore increasing K to a large value will not provide more valuable information for creating titles.

Table 4.4: F1 scores for the K nearest neighbor approach using different K values

K	1	3	4	5	10	20
F1 (w.i. word ordering)	0.219	0.210	0.210	0.212	0.212	0.173

Finally, the reverse IR approach performs well, only slightly worse than the ST method. Unlike the ST method, which relies on a EM algorithm to compute P(tw|dw), the reverse IR method has an analytic solution as shown in Equation (4.3) and the scores of title words for a test document can be computed using a standard text retrieval engine. Therefore, the reverse IR method is computationally cheaper than the ST method. However, on the other hand, the reverse IR method is non-statistical method and therefore the output scores for title words don't have the probabilistic interpretation. This defect makes it difficult for the reverse IR method to be incorporated into a probabilistic framework, whereas the ST method can be easily used as a component of a probabilistic framework.

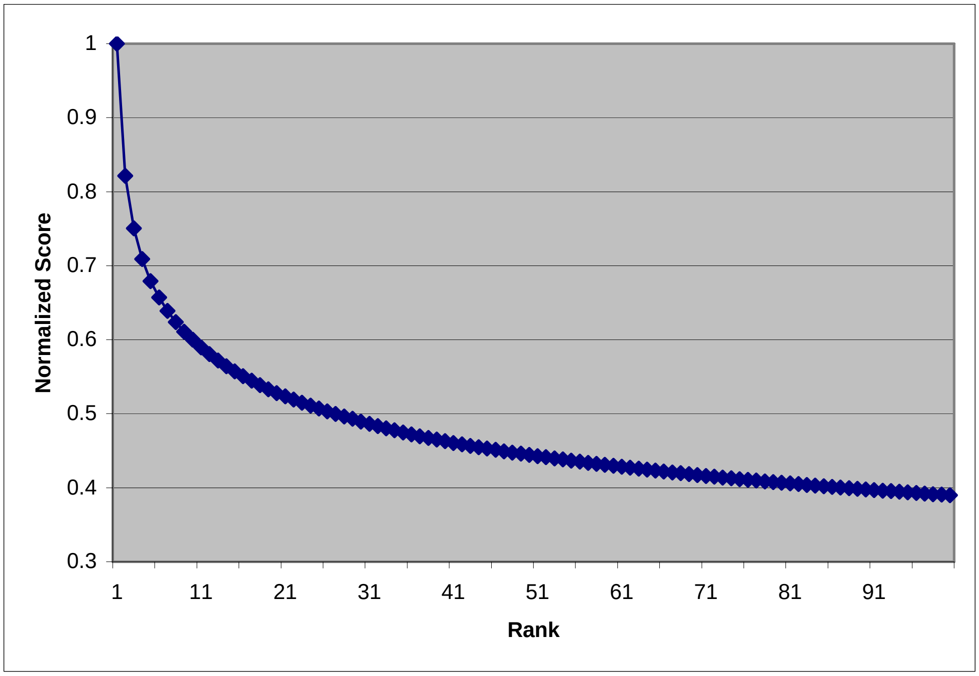

Figure 4.2: The distribution of normalized similarity scores over different retrieval ranks for the KNN method. The normalized score is defined as the score of a retrieved document divided by the score of the #1 retrieved document.

Table 4.5: The averaged human judgments for six different title word selection methods, namely the Naïve Bayes approach with limited vocabulary (NBL), the decision-tree approach (DT), the K nearest neighbor (KNN), the nearest neighbor approach (NN), the reverse information retrieval approach (IR) and the statistical translation approach (ST).

	NBL	DT	KNN	NN	IR	ST
Human Judgments	2.0	2.3	2.8	3.0	2.5	2.5

In order to further justify the quality of the machine-generated titles by these six different methods, we employ a human subject to evaluate the quality of those titles. 200 documents out of the 10,000 test documents are randomly chosen and titles of those documents are sent for human judgments. The results are listed in Table 4.5. Comparing Table 4.5 to 4.4, except the titles generated by the K nearest neighbor approach (including the KNN method and the NN method), the F1 scores of all the other methods are consistent with the human judgments. Titles generated by the KNN method are much smoother than the titles generated by other methods and therefore gain significantly better judgments by the human subject. Furthermore, the fact that the result of KNN (K=5) is worse than the NN method is because the NN method doesn’t rely on the n-gram statistical language to order the selected title words while the KNN method does. To get a sense of the titles generated by these methods, we listed 5 examples of machine-generated titles by each of the six methods in Table 4.8 to 4.13. The reference titles for the same set of documents are listed in Table 4.14, and the transcripts of the original documents are included in the appendix at the end of this dissertation.

4.5 Conclusion

In this chapter, we examine five different methods for title word selection other than the Naïve Bayes approach with a limited vocabulary. They are the decision tree method, the K nearest neighbor method (KNN), the Naïve Bayesian method with a full vocabulary (NBL), the reverse information retrieval method (IR) and the statistical translation model (ST). All these methods try to address two defects with the original NBL method, namely how to use all the words within the documents as hints for selecting appropriate title words and how to estimate the conditional probability P(tw|dw). Compared to the decision tree and Naïve Bayes methods, the statistical translation method, the reverse information retrieval method and the K nearest neighbor approach appear to be the best title word selection methods. Empirical studies show that these three methods perform significantly better than the NBL method and the decision-tree method in terms of F1 score and human judgments. Empirical studies also show that the K nearest neighbor approach has significant advantage over other method in terms of human judgments due to the smoothness of the generated titles. Based on the good performance of the statistical translation method and its flexibility in adding more probabilistic components compared to other methods, later in this thesis, unless explicitly specified, we will always use the statistical translation method for title word selection.

Table 4.6 Example titles generated by the NBL method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	NBL
1	congress latest plastic documents landmark patrol
2	jazz violinist kreme kremes tango music
3	nichols jury selection timothy mcveigh trial
4	news week stock market gains tax
5	russian space mir computer news morning

Table 4.7 Example titles generated by the NBF method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	NBF
1	interview chairman president clinton news week
2	interview ceo discusses business news week
3	timothy mcveigh oklahoma city bombing trial
4	top financial news week stock market
5	business news week space station mir

Table 4.8 Example titles generated by the TF.IDF method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	TF.IDF
1	illegal immigration border green cards card
2	classical music classics violinist play argentina
3	jury selection timothy mcveigh defense attorneys
4	capital gains tax dean witter Reynolds
5	mir computer earth space mark Moscow

Table 4.9 Example titles generated by the decision tree (DT) method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	Decision Tree (DT)
1	illegal immigrants card fraud california technology
2	jazz masters master center concert festival
3	jury selection oklahoma city bombing trial
4	stock market technology stocks wall street
5	earth space mission update murder investigation

Table 4.10 Example titles generated by nearest neighbor (NN) method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	Nearest Neighbor (NN)
1	aliens fake documents
2	tango takes hold women argentina
3	oklahoma city bombing trial begin Denver
4	expert discusses stock market
5	crisis mir

Table 4.11 Example titles generated by K nearest neighbor (KNN) method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	K Nearest Neighbor (KNN)
1	immigrant increase illegal aliens fake documents
2	living classics violinist maxim vengerov argentina
3	Jury selection oklahoma city bombing trial
4	expert discusses stock market news week
5	problems mir repair space station back

Table 4.12 Example titles generated by the reverse information retrieval (IR) method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	Reverse Information Retrieval (IR)
1	credit card fraud counterfeit illegal immigration
2	newsroom violinist tango classical music band
3	Mcveigh jurors jury selection oklahoma bombing
4	financial stocks sends stock market heights
5	mir's oxygen docks mir cosmonauts repair

Table 4.13 Example titles generated by the statistical translation (ST) method (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	Statistical Translation Model (ST)
1	ins fake cards illegal immigration week
2	entertainment tango violinist classical music band
3	mcveigh jurors sluggishly selection oklahoma bombing
4	stocks dow sends stock market heights
5	Mir's crew mir cosmonauts repair mission

Table 4.14 Examples of reference titles for 1997 broadcast news (Notice that all the stopwords are removed). The corresponding documents are included in the Appendix.

ID	Reference Titles
1	illegal immigration
2	tango homage
3	Jury selection mcveigh trial continue tomorrow
4	expert discusses stock market
5	power outage space station mir

Table 4.15 Example titles generated by the AutoSummarization function of Microsoft Word (Notice that all the stopwords are removed). The second line is empty because Microsoft Word couldn't come up with a short summarization. The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	Automatic Summarization (AutoSummarize)
1	even green card
2	
3	stephen jones mcveigh's attorney
4	katharine stocks resumed their slide wednesday lou
5	good morning mark mike

Chapter 5

FORMALIZATION OF THE TITLE GENERATION MODEL

In the last chapter, we examine different methods for title word selection. However, we still work within the framework proposed by Witbrock and Mittal (1999), where a process of title generation is divided into a phase of title word selection and a phase of title word ordering, and the phase of word ordering is accomplished using the n-gram statistical language model. As pointed out in section 2.2, the idea of using the sequence probability P(T) to order the selected title words appears to be problematic. In this chapter, we will first formalize the problem of title generation more carefully and then come up with a better solution for automatic title generation.

5.1 Formalization of the Title Generation Model

The goal of the title generation model is to find an appropriate title for a given document. From the viewpoint of probability theory, the task can be interpreted as the search of a word sequence T for document D such that likelihood P(T|D) is maximized, or

$$T = \arg\max_{T'} P(T'|D) \tag{5.1}$$

To come up with a word sequence T for the document D, we have to decide two things, i.e. what words should be used in the sequence and how should we order them. Let word set {tw∈T} stand for words used in the sequence T and

O(T) stand for the word order in the sequence T. Word sequence T can be written as T={tw∈T}^O(T) and likelihood P(T|D) can be expanded as:

$$\begin{aligned} P(T \mid D) &= P(\{tw \in T\}^\wedge O(T) \mid D) \\ &= P(\{tw \in T\} \mid D) P(O(T) \mid \{tw \in T\}, D) \end{aligned} \qquad (5.2)$$

As seen from Equation (5.2), the likelihood is expressed as the product of two terms, namely term P({tw∈T}|D) and term P(O(T)|{tw∈T},D). Term P({tw∈T}|D) stands for the probability of using the set of words {tw∈T} as title words for a given document D, and therefore corresponds to the phase of title word selection. Term P(O(T)|{tw∈T},D) is a new term, which doesn't appear in the old framework for automatic title generation (e.g. Equation (2.1)). Since P(O(T)|{tw∈T},D) stands for the probability of using word order O(T) given the content of document D and the set of selected title words {tw∈T}, it can be treated as a correspondence to the phrase of title word ordering. Compared to the old framework, where probability P(T) is used for ordering selected title words, P(T) and P(O(T)|{tw∈T},D) are two different terms. Interestingly, since a word sequence T can be written as T={tw∈T}^O(T), we can decompose P(T) as $P(T) = P(O(T) \mid \{tw \in T\}) P(\{tw \in T\})$. The first term in the decomposition, i.e., P(O(T)|{tw∈T}), appears to be similar to P(O(T)|{tw∈T},D) except that given document D is not included in the evidence set. Therefore, term P(T) is going to perform a similar function as term P(O(T)|{tw∈T},D) except that in the old framework, the content of given document D is not used for word ordering. Furthermore, the decomposition of P(T) contains an extra term P({tw∈T}) which has nothing to do with the word ordering. Since P({tw∈T}) stands for the probability of using the set of words {tw∈T} as title words, term P(T) will not only favor word sequences in correct word order but also sequences with common title words, which is not desirable for the phase of title word ordering. Based on the above analysis, we can see that using P(T) to order selected title words is not appropriate because it ignores the influence of the test document on the word ordering and may favor word sequences with common words. To improve the phase of title word ordering from the previous framework, we need to do a better estimation for term P(O(T)|{tw∈T},D).

5.2 Estimation of P(O(T)|{tw∈T},D)

One simple way to estimate the likelihood P(O(T)|{tw∈T},D) is to ignore the evidence D. Then, we can simplify P(O(T)|{tw∈T},D) as

$$\begin{aligned} P(O(T)\,|\,\{tw\in T\},D) &\approx P(O(T)\,|\,\{tw\in T\}) \\ &= \frac{P(\{tw\in T\}\wedge O(T))}{P(\{tw\in T\})} = \frac{P(T)}{P(\{tw\in T\})} \end{aligned} \qquad (5.3)$$

As indicated by the above equation, term P(O(T)|{tw∈T},D) is approximated as the ratio of P(T) to P({tw∈T}). Unlike the previous framework where P(T) is used directly for ordering title words, in Equation (5.3), by dividing P(T) by P({tw∈T}), we are able to remove the factor that P(T) favors sequences with common words. In order to illustrate this point, let’s reconsider the example used at end of Chapter 2, which is a collection of three identical documents D={‘A’, ‘B’, ‘C’, ‘D’}, and two of them have title ‘A B’ and one has title ‘C D’. If the test document is also D, according to the NBL method, the ratio P(T=‘A B’|D)/P(T=‘C D’|D) is 8 while the true ratio is 2. On the other hand, using Equation (5.3), the computed ratio is only 4 (notice that P({‘C’, ‘D’}) = 1/3 and P({‘A’, ‘B’}) = 2/3), which is closer to the true ratio 2 than the ratio estimated by the NBL method. The leftover overestimation is due to the independence assumption used in the process of generating title words.

Of course, the approximation in Equation (5.3) ignores the influence of the test document on the word order of the generated title. In the later discussion of P(T), we will consider how to incorporate the content of the test document into the estimation of the title language model, which provides room for the test document to have influence on the determination of appropriate word order for its title.

Combining Equation (5.2) with (5.3), we have

$$\begin{aligned} P(T \mid D) &= P(\{tw \in T\} \mid D) P(O(T) \mid \{tw \in T\}, D) \\ &\approx \frac{P(\{tw \in T\} \mid D) P(T)}{P(\{tw \in T\})} \end{aligned} \qquad (5.3')$$

The above equation states the basic idea of this new framework. Comparing Equation (5.3') to Equation (2.1), the biggest difference is the introduction of the new term P({tw∈T}), which alleviates the problem in using P(T) for ordering title words because P(T) can favor title sequences with common words. For the easy reference, we call the model specified in Equation (5.3') a '**direct model**'. The reason for that will be clear in the section 5.7 when we compare this model to the model used for automatic speech recognition.

According to Equation (5.3'), the proposed 'direct model' has three major components: P(T), P({tw∈T}) and P({tw∈T}|D). In the following sections, we will discuss these three components separately.

5.3 Estimating P(T)

In order to estimate P(T), we can use the n-gram statistical language model (Clarkson & Rosenfeld, 1997). More specially, if word sequence T is written as 'tw_1 tw_2… tw_n', we can write the probability P(T) as

$$\begin{aligned} P(T) = P(tw_1 tw_2 ... tw_n) &= \prod_i P(tw_i \mid tw_1 tw_2 ... tw_{i-1}) \\ &\approx \prod_i P(tw_i \mid tw_{i-1}) \end{aligned} \tag{5.4}$$

In the last step of Equation (5.4), instead of using all the words in the history to determine the appropriate word ‘tw$_i$’ for the current position, we only consider the word at the previous position and assume that words beyond the last position will have little influence on the choice of the word for the current position. This is called a bigram statistical language model. Clearly, we can have different cut offs in the word history. The longer we keep the word history, the more accurate the model will be. When the last two words in the history are kept for predicting the current word, namely $P(tw \mid tw_{-1} tw_{-2})$, we call it a trigram statistical language model. On the other hand, if we keep a long history for predicting the word at the current position, we may suffer severely from the sparse data problem in estimating $P(tw_i \mid tw_1 tw_2 ... tw_{i-1})$ because the longer the word history is, the more parameters we have. More details about the statistical language model can be found in (Katz, 1987; Baul, 1989; Niester, 1996).

According to Equation (5.4), the core part of the bigram statistical language model is the set of conditional probabilities {P(tw$_i$|tw$_{i-1}$)}, which can be estimated by simply counting the number of times that phrase ‘tw$_i$tw$_{i-1}$’ appears in the training titles and dividing it by the number of occurrence of word ‘tw$_{i-1}$’ in the same training data. However, there are two problems with this simple approach:

1. *Sparse data problem*. Due to the limited amount of training data, many two-word phrases may not appear in the training corpus. By simply counting their frequency, we will always have a zero probability for any

two-word sequence that is not observed in the training data. Methods such as Good Turing and ‘back off’ (Jelinek, 1999) have been widely used in the speech recognition community in order to smooth the estimation by simple counting. But, for titles, because the amount of title data used for training a title language model is considerably smaller than the data used for training a document language model (considering each title only consists of an average of six words while each document usually contains hundreds of words), the problem can be more severe, even with the help of standard smoothing methods. This suspicion is further confirmed by the experiment of automatic title generation using a trigram language model, which results in a F1 score worse than a simple bigram language model. Therefore, we may need methods other than Good Tuning and ‘back off’ methods to alleviate the sparse data problem in training a title language model.

2. *Word order independence problem.* If we only rely on the title data in the training set to build up the title language model, the resulting estimation of word order will be independent from the content of the test document. Clearly, this word order independence problem can give rise to completely wrong titles. For example, the content of the original document is about U.S. air fighters bombing Iraq. However, due to the ignorance of the content of the test document, the generated title can be ‘Iraq bombed U.S.’. Therefore, we may need to incorporate the content of the test document into the title language model.

In the following sections, we will examine these two problems separately.

5.3.1 Sparse Data Problem

In speech recognition, the sparse data problem has been one of the biggest problems in creating a reliable language model. The main ideas for alleviating the sparse data problem are: adding more training examples or introducing some kinds of smoothing strategies. In this section, we will consider three methods that have potential to alleviate the sparse data problem in training a title language model.

5.3.1.1 Incorporating Document Data into Training Set

The simplest way to deal with the sparse data problem is to add more training data. Even though the amount of title data is small, the corresponding document data is large. Therefore, we can use documents as extra training data for training a title language model by assuming that both documents and titles are created from a similar language model. The advantage of this approach is the significant expansion of training data. However, the disadvantage of this approach is that, since documents and titles are two different sets of objects, they should be created from different underlying stochastic procedures. Mixing together document data and title data for training a title language model can severely dilute the characteristics of the 'true' title language model. As a result, the estimated P(tw|tw') and P(tw) may not be able to reflect the correct composition of titles.

5.3.1.2 Class based Language Model

In the 'back off' approach, a bigram probability P(tw'|tw) will be estimated by unigram probability P(tw') whenever the phrase "tw' tw" is not observed in the training data. When the data are extremely sparse, many bigram probabilities will be backed off to the corresponding unigram probabilities, which can significantly degrade the estimation. One way to improve the quality of estimation is to avoid the direct back off of a bigram probability to the corresponding unigram probability by introducing a class variable for each word. That is called the class (category) based language model (Niesler, 1997). The main idea can be stated as follows: Let C be the function that maps a word to a class and C(w) stand for the mapped class for word ‘w’. An example of such a class can be the part of speech tagging (POS) and the corresponding mapping function C is the speech-tagging algorithm that tags each word with ones of the predefined POS. With the class function C, the original conditional probability P(tw’|tw) can be written as P(C(tw’), tw’|tw). So, if the word pair “ tw tw’ ” doesn’t appear in the training corpus, instead of relying on the unigram language model to estimate the bigram probability, we can back off to conditional probability P(tw’|C(tw’))P(C(tw’)|tw), i.e., first generate the class of words C(tw’) and then generate the actual word “tw’” from the estimated class C(tw’). As the last step, if we still haven’t observed the combination “ tw C(tw’) ” in the training corpus, we will then back off to the probability P(tw’). As you can see from the above analysis, by introducing the class information for each title word, we are able to avoid the direct jump from a bigram probability to a unigram probability in the back off approach. In our implementation, we use the part of speech (POS) as the class variables and the toolkit used for extracting the POS information is downloaded from the web site of Infogistics (http://www.infogistics.com/posdemo.htm).

5.3.1.3 Long Distance Language Model

The other approach to avoid the sparse data problem is to examine more evidence in the word history. As indicated in Equation (5.4), for a bigram statistical language model, in order to decide how likely it is that the word tw_i will be used for the i-th position, we will only look at the word at the previous position tw_{i-1} and if the word tw_{i-1} provides no clue for the determination of tw_i, we will use the ‘back off’ approach, which is to determine word tw_i by the unigram probability $P(tw_i)$ with some discounts. However, there could be words in the depth of the word history that can help with the choice of the word tw_i. By taking into account more words in the history, we are able to collect more hints for the selection of words for the current position. In the speech recognition community, several language models with consideration of long distance dependency have been proposed. They are the trigger model (Rosenfeld, 1994), the structured language model (Chelba, 1997) and the multi-span language model (Bellagarda, 1998). The difference between them is the way that they capture the long distance dependency. The structured language model relies on the syntactic structure of a sentence to find the most relevant evidence in the long history while the trigger model and multiple-span language model capture the long distance dependency by extracting useful semantic evidence out of the word history.

In this thesis, we consider a very simple long distance language model. The basic idea is similar to the trigger model (Rosenfeld, 1994), namely combining more words in the deep history to predict the word for the current position. In the trigger model, the most informative triggers are the cases when a word in the history repeats itself at the current position. For a title, it is unlikely that the same word appears twice in a single title. Therefore, we have to take into account all the words in the history. Furthermore, for the sake of simplicity, we take the linear approach for combining the predictions of all the history words. More specifically, the likelihood $P(tw_i|tw_1\ tw_2\ldots\ tw_{i-1})$ is approximated as a linear expansion of the following expression, i.e.,

$$P(tw_i \mid tw_1 tw_2 ... tw_{i-1}) \approx \sum_{j=1}^{i-1} P(i-j) P(tw_i \mid tw_j) \qquad (5.5)$$

As illustrated by Equation (5.5), to decide the appropriate word for the i-th position, we will use all the words in the history and weight their opinions based on their distances from the current position. Thus, we are going to have two sets of parameters: P(w|w’), i.e. the parameters for the correlation between words, and P(i-j), i.e., the parameters for the correlation between different positions. Notice that a simple bigram language model is a special case of the ‘long distance language model’ by simply setting P(k)=δ(k,1). To obtain the estimation for both sets of parameters, we can deploy the Expectation-Maximization (EM) algorithm (Dempster, 1977). The updating equations of the EM algorithm for P(w|w’) and P(i-j) are:

$$\begin{gathered} P'(tw'|tw) = \frac{1}{Z(tw)} \sum_{\{T=\ldots,tw',\ldots,tw,\ldots\}} \frac{P(tw'|tw)P(pos(tw',T)-pos(tw,T))}{\sum_{j=1}^{pos(tw',T)} P(pos(tw',T)-j)P(w(T,pos(tw',T))\,|\,w(T,j))} \\ P'(l) = \frac{1}{Z_0} \sum_{k=0} \sum_{\{T\|T|>l+k\}} \frac{P(w(T,k+l+1)\,|\,w(T,k+1))P(l)}{\sum_{j=1}^{l+k-1} P(l+k+1-j)P(w(T,l+k+1)\,|\,w(T,j))} \end{gathered} \tag{5.6}$$

where the pos(tw,T) stands for the position of word ‘tw’ in title ‘T’ and w(T,j) stands for the j-th word in the title T. Both Z(tw) and Z_0 are normalization factors.

According to the above discussion, the advantage of this ‘long distance language model’ is that all the words in the history are involved in selecting words for the current position. Therefore, we may not suffer from the sparse data problem as severely as the bigram language model. However, this may not be true. Due to the Markov nature of natural languages (i.e., Shannon’s game), the previous word will be most informative to the choice of the current word. Therefore, we may expect a very quick decrease in the distribution of P(k=i-j) in Equation (5.5), which can significantly limit the influence of words in the long history on the choice of the current word.

5.3.1.4 Experiments

In this subsection, we are going to compare the proposed three language models to the simple bigram language model for estimating P(T). The basic setup of this experiment is same as that stated in section 4.4, namely the CD of broadcast news of 1997 is used as the testbed with 40,000 training document-titles and 10,000 test document-titles. The statistical translation method is used for title word selection. For the class-based language model, we use the part of speech tags (POS) as the class labels of words. For the long distance language model, we use the last five words in the history to predict the current word. For each method, the generated titles are of six words. The evaluation metric used in this experiment is F1. Table 5.1 lists the F1 results for all four different language models.

Table 5.1: F1 scores for four different language models.

	Bigram LM	Bigram LM + Documents	Class-based LM	Long distance LM
F1	0.223	0.197	0.214	0.220

Surprisingly, nothing works better than the simple bigram language model. The idea of incorporating the documents into the training data for building up a title language model works worst. We think it is because documents and titles are two different sets of objects and therefore simply using the mixture of documents and titles to train a statistical title language model will dilute the special word-to-word correlation in titles. In order to show that documents and titles are different in statistics, we examine the unigram language models of these two types of objects and compute the averaged ratio $P(w \mid M_{title}) / P(w \mid M_{title+doc})$ over all the title words. The averaged ratio is 2.4. The fact that this ratio is substantially larger than one indicates that the language model of documents is actually quite different from the language model of titles. The fact that the class-based language model is not able to improve the standard bigram language model is consistent with what people found in automatic speech recognition, where the class based language model only gives very modest improvement in the word error rates even though it is able to decrease the perplexity of training data noticeably (Jurafsky & Martin, 2000). The failure of the linear long-distance language model is because the position-dependent correlation probabilities P(k=i-j) decrease very quickly when the two words are far away. Therefore, even though the long-distance language model tries to collect all the hints in the history for the prediction of the current word, the multiplication of the position-dependent correlation probabilities P(k=i-j) with the word-dependent correlation probabilities P(w|w') in Equation (5.5) causes the influences of words in the deep history to become almost negligible. The distribution of position-dependent correlations P(K) at different distance K (i.e., i-j) are plotted in Figure 5.1, which clearly indicates a strong exponential decay in P(K). Due to this defect, the long distance language model expressed in Equation (5.5) has not been able to improve the standard bigram language model. In conclusion, even though two methods have been tried in order to deal with the sparse data problem in training a bigram title language model for titles, neither of them is able to improve the F1 scores. This is similar to the situation in automatic speech recognition, where many sophisticated language models have been proposed but almost none of them is able to improve over the trigram statistical language model noticeably, particularly in terms of the word error rate.

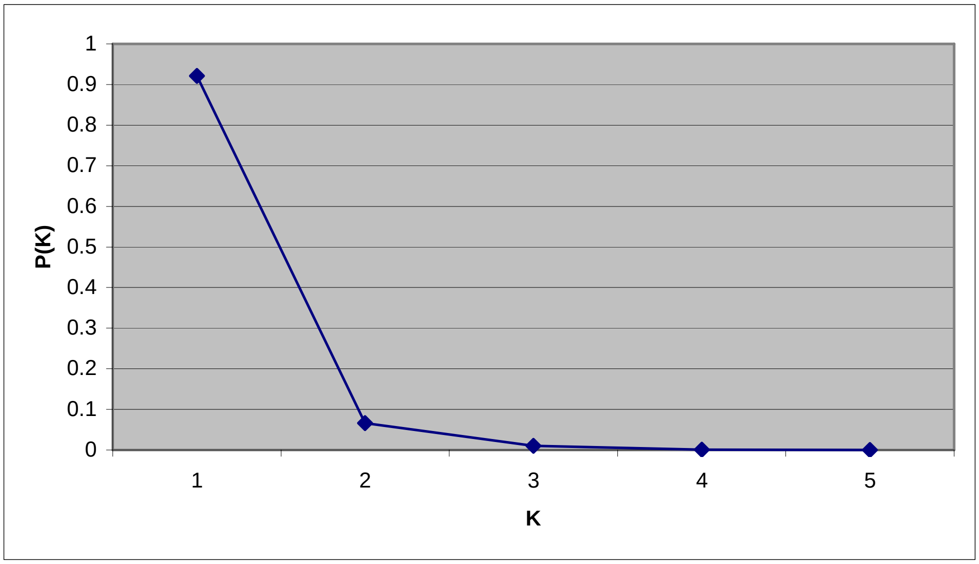

Figure 5.1: The distribution for the position-dependent correlation P(K) over different positions. The horizontal axis represents the distance between words and the vertical axis represents the probability distribution of position-dependent correlation.

5.3.2 Word Order Independence Problem

In order to incorporate the content of the test document into the determination of word order for the created title, we need to consider two different types of language models, namely the language model for titles (i.e., M_T), and the language model for the test document D (i.e., M_D). In the simple bigram language model, P(T) is estimated using only the title language model M_T. The simplest way of putting these two language models together is to use the mixture of language models M_T and M_D as the 'expanded title language model', and order the selected title words using the 'expanded title language model'. More specifically, we can write P(T) as $P(T|M_T,M_D)$ and expand it as a linear combination of the estimations based on each individual language model, i.e.,

$$
\begin{aligned}
P(T \mid M_T, M_D) &= \prod_{i=1}^{|T|} P(tw_i \mid tw_{i-1}, M_D, M_T) \\
&\approx \prod_{i=1}^{|T|} \left\{ \lambda P(tw_i \mid tw_{i-1}, M_T) + (1-\lambda) P(tw_i \mid tw_{i-1}, M_T) \right\}
\end{aligned}
\tag{5.7}
$$

where λ is the combination constant which falls into the region [0,1].

Similar to the previous subsections, we use the CD of 1997 broadcast news as the testbed with 40,000 training document-titles and 10,000 test document-titles. A statistical translation model is used for title word selection. The F1 results for different smoothing constants λ are listed in Table 5.2. First of all, according to Table 5.2, for all different values of the smoothing constant λ, the mixture model always outperforms the simple title language model in terms of F1 score except when λ is 0.9. This fact indicates that the language model of the test document is useful in ordering title words. Secondly, the F1 score reaches the maximum value when the smoothing constant λ is 0.3. After that, increasing the smoothing constant λ only causes the F1 score to degrade. Particularly, when λ is 0.9, the performance degrades significantly. Therefore, even though the document language model is useful in creating title sequences, the title language model still plays an important role in determining the word order. In the later experiment, we will always use the smoothing constant λ equal to 0.3 whenever the expanded title language model is applied.

Table 5.2: F1 scores for different smoothing constant λ.

λ	0	0.1	0.3	0.5	0.7	0.9
F1	0.223	0.237	0.247	0.241	0.231	0.170

5.3.3 Conclusion

In this subsection, we experimented with several different language models in order to address the sparse data problem and the word order independency problem. Four different language models are proposed and examined, including a bigram title language model trained over the data of both titles and documents, a class-based language model using the part of speech tags, a linear long-distance language model, and the expanded title language model with a mixture of the original title language model and the language model of the test document. We find that only the 'expanded title language model' is able to outperform the baseline bigram language model in terms of F1 metrics and all the other language models don't help the performance. Since the expanded title language model gives the best performance, in the later part of this thesis, unless explicitly specified, we will always use the expanded title language model for estimating P(T).

5.4 Estimating P({tw∈T})

As illustrated in Equation (5.3'), the introduction of term P({tw∈T}) is able to compensate for the fact that P(T) favors word sequences with common words. In this section, we will discuss three different ways of estimating term P({tw∈T}), namely an unigram expansion method, an exponential model and an order expansion method.

5.4.1 Unigram Expansion

The simplest way of estimating probability P({tw∈T}) is to approximate it as the product of unigram probabilities, i.e., P(tw), under the assumption that each word 'tw' in the set {tw∈T} appears independently from each other. Therefore, the probability P({tw∈T}) can be simply written as,

$$P(\{tw \in T\}) \approx \prod_{tw \in T} P(tw) \qquad (5.8)$$

One problem with this simple unigram estimation is that it will usually underestimate probability P({tw∈T}) when the set of words {tw∈T} are strongly correlated with each other. For example, say the words ‘President’ and ‘Clinton’ always come out together as ‘President Clinton’. Assuming that ‘President Clinton’ occurs in one out of every hundred titles, we will have P(‘President’) = P(‘Clinton’} = 1/100. Therefore, using the unigram estimation, the probability P({‘President’, ‘Clinton’}) will be only 1/10000, which is much less than 1/100. In order to prove this idea more quantitatively, for the top 100 popular two-word phrases in titles, we compute the ratio of joint probability P({tw, tw’}) with respect to the unigram estimation for P({tw,tw’}), i.e. P(tw)P(tw’). This ratio is 2090, which indicates that the simple unigram expansion substantially underestimates the joint probability.

5.4.2 Exponential Model

To account for the fact that many title words are actually strongly correlated with each other, we consider a simple exponential model for the estimation. The exponential model is a well-known model that is able to handle correlated features (Pietra et al., 1997) better than the simple independence assumption. In general, to estimate the probability for a vector **x**, it assumes P(**x**) has the following form

$$P(\mathbf{x}) \approx \frac{1}{Z}\exp(\sum_i \alpha_i f_i(x)) \qquad (5.9)$$

where Z is the normalization constant, f_i(**x**) is the i-th feature for the vector **x** and α_i is the corresponding weight. By adjusting the weights for different features, the exponential model is able to take into account the correlation between features. For the case of estimating P({tw∈T}), we can treat each word as a feature and the whole set of words {tw∈T} can be viewed as a feature vector. Then, similar to Equation (5.9), we can have P({tw∈T}) expressed as follows:

$$P(\{tw \in T\}) \approx \frac{1}{Z}\exp(\sum_{tw'} \alpha_{tw'} f_{tw'}(\{tw \in T\})) \tag{5.10}$$

where $\alpha_{tw'}$ stands for the weight for word 'tw′' and $f_{tw'}$({tw∈T}) is a indicator function which will give 1 if the word 'tw′' belongs to the set {tw∈T} and zero otherwise. However, if we allow a free parameter for each word 'tw′', we will run into a serious problem of data sparseness. Therefore, in this thesis, we assume that parameter α_{tw} has the following simple parametric form:

$$\alpha_{tw} = \alpha \ln P(tw) \tag{5.11}$$

In the above expression for α_{tw}, α is the only parameter to be decided and the difference between weights is determined by the unigram probability P(tw). Interestingly, if we substitute the expression for α_{tw} (in Equation (5.11)) into the expression for P({tw∈T}) (Equation (5.10)), we will have

$$P(\{tw \in T\}) \approx \frac{1}{Z}\exp(\sum_{\{tw \in T\}} \alpha \ln P(tw)) \propto \prod_{tw \in T} P^{\alpha}(tw) \tag{5.12}$$

Different from the simple unigram expansion in (5.8), Equation (5.12) weights each unigram probability to power α. When α is set to be zero, the probability P({tw∈T}) becomes a constant, which means that the factor P({tw∈T}) is ignored. On the other hand, when α is set to be one, we go back to the simple unigram expansion as in Equation (5.8). To determine the optimal α, we simply do the exhaustive search over different values of α and find the one that results in the best performance for title generation.

In order to see the effectiveness of this exponential model, similar to previous experiments, we tested the exponential model in Equation (5.12) against the CD of 1997 broadcast news with 40,000 training document-titles and 10,000 test document-titles. The titles are generated using the new model specified in Equation (5.3'), where term P({tw∈T}|D) is estimated using the statistical translation model and term P(T) is estimated using the expanded title language model as described in the last section. The F1 scores of the machine-generated titles for four different α values are listed in Table 5.3.

As indicated in Table 5.3, when α is set to be one, namely a simple unigram expansion for estimating P({tw∈T}) is used, we have the worst performance, which indicates that the naïve independence assumption is not appropriate for the estimation of P({tw∈T}). Meanwhile, when α is set to be zero, term P({tw∈T}) is simply a constant according to Equation (5.12) and therefore we go back to the previous framework for title generation. As seen from Table 5.3, the best performance is achieved with F1=0.261 when α is set to be 0.3.

Table 5.3: F1 scores for different α values

α	0	0.3	0.6	1
F1	0.247	0.261	0.240	0.190

5.4.3 Order Expansion

As discussed in the previous subsection, by introducing the weight constant α, we are able to relax the independence assumption a little bit. In this subsection, we would like to consider another way of estimating P({tw∈T}) that is able to alleviate the problem of the independence assumption even more.

The n-gram statistical language model has been proved to be an effective tool for estimating the probability for a word sequence, i.e. P(T). In order to apply the n-gram statistical language model to estimate the probability for a set of words, i.e. P({tw}), we need to bridge the gap between a set of words {tw} and a word sequence T. Since a set of words {tw} can be transformed into a word sequence T by imposing a particular word order, we can expand P({tw}) over all possible word orders for {tw} as shown in Equation (5.13) and then a n-gram statistical language model can be used to estimate each P(S) within the sum.

$$P(\{tw\}) = \sum_{O} P(\{tw\} \wedge O) = \sum_{\{S | \{tw' \in S\} = \{tw\}\}} P(S) \qquad (5.13)$$

In the above equation, symbol ‘O’ stands for a particular word order. As indicated in Equation (5.13), P({tw}) is expanded as the sum of probabilities of word sequences and each word sequence ‘S’ is created from the set of words {tw} with a particular word order ‘O’ . By applying a n-gram statistical language model to estimate every P(S) in the right side of Equation (5.13), we can obtain the estimation for P({tw}).

In order to see why the order expansion can alleviate the problem with the word correlation, let’s consider two extreme cases: first consider the case when the set of words {tw} are loosely correlated, namely the set of words {tw} should not be used together. According to Equation (5.13), the P({tw}) is computed as sum of P(S) and each P(S) is computed as $P(s) \propto \prod_{i} P(tw_i \mid tw_{i-1})$ using a bigram language model. Since the set of words {tw} are loosely correlated, most conditional probabilities $P(tw_i \mid tw_{i-1})$ will be small and thus the estimation of all P(S) will also be small. As a result, the estimation of P({tw}) is small. On the other hand, let’s consider the best case, i.e., words {tw}are strongly correlated with each other. For example, consider the case when the set of words {tw} are actually extracted from a common sentence.

Then, if we put the set of words {tw} into the word order of the original sentence from which {tw} are extracted, the corresponding P(S) should be quite large, which will result in a large value for the estimation of P({tw}). In order to quantitatively show that the order expansion is able to take into account the word correlation better than the unigram expansion, for the same set of top 100 popular two-words title phrases as used in 5.4.1, we compute the ratio of P({tw, tw'}) to the estimation based on order expansion. The averaged ratio is only 1.03, which is significantly better than the averaged ratio achieved by the unigram expansion (i.e., 2090).

However, one problem with using Equation (5.13) directly for estimating P({tw}) is the computational complexity in the sum. As a simple example, if a word set consists of ten different words, the total number of possible word orders exceeds 3 million. Therefore, computing the sum in Equation (5.13) directly is infeasible. However, we can apply the technique similar to the stack search (Jelinek, 1999) to eliminate majority of terms from the sum in Equation (5.13) using the assumption that there will only be a small number of word orders that can contribute significantly to the final value of P({tw}). In practice, for a given set of words, we start with a sequence S of zero length and add one word each time. Clearly, the number of different word sequences will grow exponentially if we keep on adding more words. To avoid this situation, with the assumption in mind that only a small number of word sequences are important to the estimation of P({tw}), we will only keep the top 1000 word sequences with the highest probabilities and throw away other less likely word sequences. When we add in the last title word, the probability of the top 1000 word sequences will be summed together as the approximation for the probability P({tw}). In this way, we are able to avoid computing the exponential number of terms within the sum but still achieve good estimation for the probability P({tw}). Furthermore, we have tried out different numbers of word sequences that are kept for the estimation of P({tw}), ranging from 2000 to 5000, and found that their F1 scores are almost identical to the case when only the top 1000 word sequences are kept. More details about stack search can be found in (Jelinek, 1999).

In order to see whether or not the order expansion method is effective for estimating P({tw∈T}), we conducted an experiment similar to what is stated in the last subsection with 40,000 training document-titles and 10,000 test document-titles. The titles are generated using the new model specified in Equation (5.3'), with term P({tw∈T}|D) being estimated by the statistical translation model, term P(T) being estimated by the expanded title language model and term P({tw}) being estimated by the order expansion method. F1 scores of machine-generated titles using both the exponential model and the order expansion method are listed in Table 5.4. According to Table 5.4, the estimation of P({tw∈T}) using the order expansion method achieves slightly better F1 score than the exponential model.

Table 5.4: F1 score for the order expansion method and the exponential model

	Exponential Model (α=0.3)	Order Expansion
F1	0.261	0.270

5.4.4 Conclusion

In this section, we examined three different methods for estimating P({tw∈T}), namely the unigram expansion method, the exponential model and the order expansion method. First, the empirical studies show that by including term P({tw∈T}), we are able to improve the F1 scores from 0.247 to 0.27 by using the order expansion method. Secondly, based on the comparison of the three methods for estimating P({tw∈T}), we find that the method of order expansion is able to take into account the word correlation better than both the unigram expansion and the exponential model. The empirical result also indicates that the order expansion method achieves a better F1 score than the other two methods. Thus, we conclude that the introduction of P({tw∈T}) is useful for the title generation model and the method of order expansion is an effective method for estimating P({tw∈T}) compared to others. For the later part of this thesis, unless explicitly specified, we will always use the order expansion method for estimating P({tw∈T}).

5.5 Title Word Selection Problem

In the previous sections, we mainly focused on the phase of title word ordering. In this section, we come back to the issue of title word selection and discuss how to improve the quality of title word selection, namely how to estimate the likelihood P({tw∈T}|D) in a better way.

In the previous discussion of estimating P({tw∈T}|D) in Chapter 4, we focused on the idea of how to combine all the words within document D together as evidence for finding appropriate title words, which results in the general formula (4.5). The flaw within that reasoning is that not every word in the document can be used as evidence for selecting title words. Particularly, a document may contain many common words, which usually only have linguistic functions and don't deliver the semantic meaning of documents. Therefore, a better strategy for title word selection should be to first sample out the important content words from the document and then determine the appropriate title words based on the sampled document words. Therefore to accomplish the title word selection task, we need two noisy channels: the first noisy channel distills important content words out of the original document and the second channel chooses the title words according to the 'distilled' content. Since the previous work on title word selection only considers one single channel that transforms a document into a title, we will call the new model the ‘dual noisy channel model’ and the previous model for title word selection the ‘single channel model’. A simple diagram in Figure 5.2 illustrates the difference between the ‘dual noisy channel model’ and the previous work on title word selection as stated in Chapter 4.

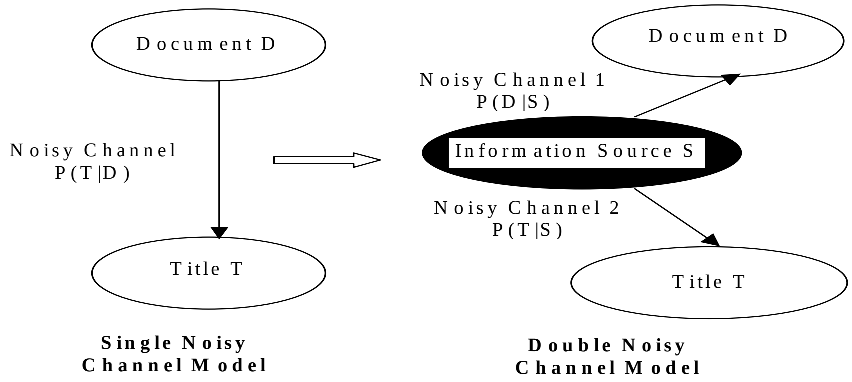

Figure 5.2: The graphic representation for the ‘single noisy channel model’ and the ‘dual noisy channel model’.

As indicated in Figure 5.2, for the previous work on automatic title generation, there is only one noisy channel P(T|D) which creates a title T directly from the document D. In the new model, both the document and the titles are generated from the same ‘information source’ hidden in the author’s mind. Importantly, the processes for creating a document and a title from an ‘information source’ are through two different noisy channels, namely the noisy channel P(D|S) and the noisy channel P(T|S). The ‘information source’ essentially is a distilled document with the trivial word removed and only the important content words kept. Therefore, in order to generate a title for the document, we need to first recover the hidden ‘information source’ by reversing the noisy channel P(D|S) and then create a title from the recovered ‘information source’.

To allow titles being generated from the distilled ‘information source’ instead of the original document, we can expand the probability P(T|D) as the sum of the probabilities P(T|S) over all the possible ‘information sources’ S. Formally, this idea can be expressed as:

$$P(T \mid D) = \sum_{S} P(T \mid S) P(S \mid D) \tag{5.14}$$

In Equation (5.14), term P(T|S)P(S|D) represents the idea of two noisy channels, with term P(S|D) corresponding to the first channel that samples ‘information source’ S out of original document D and term P(T|S) corresponding to the second noisy channel that creates title T from the distilled ‘information source’ S. Since the first noisy channel, i.e., P(S|D), is new to the old framework for automatic title generation, in the following discussion, we will focus on the discussion of the noisy channel P(S|D).

Because the motivation of introducing the hidden state ‘information source’ S is to strip off the common words and have important content words kept, we want the noisy channel P(S|D) to be a sampling process where important content words have higher chances to be selected than common words. Let function g(dw,D) represent the importance of word ‘dw’ related to the document D. Then, the word sampling distribution should be proportional to the word importance function g(dw,D). Therefore, to be consistent with this intuition, following the spirit of exponential model, we can assume that the probability P(S|D) is written as

$$P(S \mid D) = \frac{1}{Z(D)} \prod_{dw \in S} g(dw, D) \propto \prod_{dw \in S} g(dw, D) \tag{5.15}$$

where Z(D) stands for the normalization constant for the document D. As indicated by Equation (5.15), the probability for ‘information source’ S to represent the content of the document D, i.e., P(S|D), depends on whether or not the words selected by S are important to the content of the document D, which is expressed as the product of the importance function for all the words selected by ‘information source’ S.

By putting Equations (5.2), (5.14) and (5.15) together, we will have

$$P(T \mid D) \propto \frac{P(T)}{P(\{tw \in T\})} \sum_{S} P(\{tw \in T\} \mid S) \prod_{dw \in S} g(dw, D) \qquad (5.16)$$

In the above Equation, we ignore the normalization constant Z(D), which is independent from title T. Because the number of different ‘information source’ S for document D is on the order of 2^|D|, computing the sum in Equation (5.16) is almost impossible. In the following part, we will discuss an approximation of Equation (5.16) that makes the computation of the sum possible.

First, let’s only consider the ‘information source’ S that has the same number of unique words as title T. In other words, we ignore the case when there are either too few or too many unique words in S. Secondly, let’s further assume that the process of creating title T out of the ‘information source’ S can be divided into the following two steps: first align every title word position with a different word in the ‘information source’ S and then create a title word ‘tw’ for each position according to the aligned document word ‘dw’ and the probability distribution P(tw|dw). With these two assumptions, Equation (5.15) can be simplified as

$$
\begin{aligned}
P(T \mid D) &\propto \frac{P(T)}{P(\{tw \in T\})} \sum_{S} P(\{tw \in T\} \mid S) \prod_{dw \in S} g(dw, D) \\
&\approx \frac{P(T)}{P(\{tw \in T\})} \sum_{S} \sum_{a} \prod_{dw \in S} P(a(dw) \mid dw) \prod_{dw \in S} g(dw, D) \\
&= \frac{P(T)}{P(\{tw \in T\})} \sum_{S} \sum_{a} \prod_{dw \in S} P(a(dw) \mid dw) g(dw, D) \\
&\propto \frac{P(T)}{P(\{tw \in T\})} \sum_{a'} \prod_{tw \in T} P(tw \mid a'(dw)) g(dw, D) \\
&\approx \frac{P(T)}{P(\{tw \in T\})} \prod_{tw \in T} \sum_{dw \in D} P(tw \mid dw) g(dw, D)
\end{aligned}
\tag{5.17}
$$

where the variable ‘a’ in the above equation corresponds to the alignment between the title T and the ‘information source’ S. Symbol a(dw) stands for the title word that is aligned with the document ‘dw’ and a(tw) stands for the document word that is aligned with title word ‘tw’. The last step uses the fact that the operation of sum and product can be switched, which is also used in the proof of IBM statistical model I. As indicated from Equation (5.17), with the constraint that the number of unique words in S equals the number of words in the title and the assumption that every word in the title is aligned with a different word in S, we are able to change the order between the sum operation and the product operation, and therefore have a simple expression in Equation (5.16). In the annotation of big O, we have computation complexity O(|D||T|+|T|) for Equation (5.17) and O(|T|^|D|*|T|) for Equation (5.16). Clearly, the simplification in Equation (5.17) makes the computation much more efficient. Furthermore, compared to the formula (4.5), where the title word selection probability is expressed as $\frac{1}{|D|} \sum_{dw \in D} P(tw \mid dw) tf(dw, D)$, the difference between the previous model (e.g., single noisy channel model) and the ‘double noisy channel model’ is that, the new model uses the importance function g(dw, D) to weight the document word ‘dw’ while the old model simply weights word ‘dw’ based on its term frequency.

Of course, one key issue with the ‘dual noisy channel model’ is to find the appropriate importance function g(dw,D) that is able to indicate whether or not a word ‘dw’ is important to document D. Unfortunately, there is no training data with documents on one hand and their ‘information sources’ on the other hand. In the following subsections, we will discuss two candidates for importance function g(dw,D), namely the TF.IDF value and the content-based language model.

5.5.1 A TF.IDF Value for Importance Function

The TF.IDF value has been broadly used in information retrieval. The TF value of a word indicates how frequently that word has been used in the document and the IDF value indicates how rarely a word is used in different documents. A high TF.IDF value for a word usually means that the word appears frequently in a document but is rarely used by other documents. Therefore, words with high TF.IDF are usually important content words to a document.

A TF.IDF value can also be interpreted as the mutual information between document ‘D’ and word ‘w’ (Berger & Lafferty, 1999). For a given collection C, the mutual information between word ‘w’ and document ‘D’, i.e., I(w,D|C), is defined as:

$$I(w,D\,|\,C) = P(w\,|\,D)\log\frac{P(w\,|\,D)}{P(w\,|\,C)} = P(w\,|\,D)\left(\log\{1/P(w\,|\,C)\} - \log\{1/P(w\,|\,D)\}\right) \qquad (5.18)$$

First, if we are only interested in those not-very-common words, usually 1/P(w|D) (O(|D|)) is considerably smaller than 1/P(w|C) (O(|C|)). Therefore, for those words, mutual information I(w,D|C) can be approximated as P(w|D)log{1/P(w|C)}. Since the IDF value for a word ‘w’ is usually defined as the logarithm of the ratio of the number of documents in a collection to the number of documents containing word ‘w’, it equals term log{1/P(w|C)}. Furthermore, since the probability P(w|D) is defined as the ratio of TF value for the word ‘w’ to the document length, the mutual information I(w,D|C) can be simply approximated as |D|×TF.IDF(w,D). Therefore, TF.IDF measurements basically correspond to the mutual information between words and documents.

5.5.2 Content-based Language Model for Importance Function

As pointed out before, the difference between the new model and the previous model for title word selection is that the old model uses tf(w,D)/|D| to weight different words while the new model uses a importance function g(dw,D). Since tf(w,D)/|D| is simply the unigram language model for document D, in order to improve from the old model we need a better language model for documents. The problem with the simple unigram language model is that it mixes together the word distribution for the document content and the word distribution for general English. In general, to create an English article, we need two parts of knowledge, i.e. the general knowledge about how to write an English paper and the knowledge about the story that you are telling. In other words, a document is generated from the combination of two language models, namely the general English model and the content-based language model. Therefore, instead of using the simple unigram language model for documents, which is the combination of two different language models, we should use the content-based language model to create titles. In order to extract the content-based language model, we can first compute the simple unigram language model for the document and a general English language model by collapsing all documents together. Then the content-based language model is simply computed by subtracting the unigram language model from the general English language model. Formally, we have the content-based language model expressed as follows:

$$P'(w \mid D) = \begin{cases} P(w \mid D) - \lambda P(w \mid C) \quad if & P(w \mid D) - \lambda P(w \mid C) > 0.0001 \\ 0.0001 & o.w. \end{cases} \qquad (5.18)$$

where λ is a weight constant which determines how much of the general English language model needs to be removed. To avoid the sparse data issue, we set the minimum value for the content-based language model to be 0.0001.

5.5.3 Empirical Study

To determine the effectiveness of the two proposed methods, we conducted an experiment similar to the last section. The new title word selection model is tested against the CD of 1997 broadcast news with 40,000 training document-titles and 10,000 test document-titles. The machine-generated titles are compared to the reference titles using the F1 metric. The titles are generated using the probabilistic model specified in Equation (5.17), where term P(T) is estimated using the expanded title language model, term P({tw∈T}) is estimated using the order expansion method. Probability P(tw|dw) is estimated using the statistical translation model as described before. For the content-based language model, the smoothing constant λ is set to be 0.9 (Other values for λ, ranging from 0.8 to 0.95, have been tried out with almost identical F1 scores.). Both the results of using TF.IDF values and the content-based language model as the importance function, together with the result for the model without using the dual noisy channel, are listed in Table 5.5.

Table 5.5: F1 scores for the duel noisy channel model using the TF.IDF values (or mutual information) and the content-based language model, together with the similar model but with a single noisy channel.

	TF.IDF	Content-Based LM	No dual noisy channel
F1	0.280	0.271	0.270

As shown in Table 5.5, using the TF.IDF value as the importance function gives rise to a better F1 score compared to the content-based language model approach, and using the content-based language model as the importance function doesn't improve F1 score at all compared to the similar model but only with single noisy channel. Therefore, TF.IDF value based dual noisy channel (or mutual information based dual noisy channel model) appears to be a better choice for the importance function. In the later part of this thesis, we will always use TF.IDF value as the second noisy channel.

5.6 Summary of The New Probabilistic Model for Title Generation

In the previous sections from 5.1 to 5.5, we describe the main features of the proposed new model for title generation. In section 5.1 and 5.2, we discuss the correct formalization for automatic title generation problem, which is expressed in Equation (5.3'). The new feature of this formalization is the introduction of term P({tw∈T}). As indicated from Equation (5.3'), compared to P(T), using the ratio of P(T) to P({tw∈T}) for title word ordering alleviates the problem that P(T) is influenced not only by the quality of the word order in sequence T but also by whether or not the words inside T are common words. In section 5.3, 5.4 and 5.5, we discuss how to estimate the three components in Equation (5.3'), namely P(T), P({tw∈T}) and P(tw|D). For the estimation of P(T), the empirical results indicate that the expanded language model with the mixture of a title language model and the language model of the test document achieves the best performance. For P({tw∈T}), the order expansion method is able to handle the word correlation better than the other two methods and achieves the best F1 score. For P(tw|D), the TF.IDF based dual noisy channel model (or mutual information based dual noisy channel model) obtains the best performance. For the later experiments, unless explicitly specified, the implementation of the proposed model will always use the expanded title language model for estimating P(T), the order expansion method for estimating P({tw∈T}), the dual noisy channel model with the TF.IDF value (or mutual information) as the importance function g(dw,D) and a statistical translation model for estimating P(tw|dw).

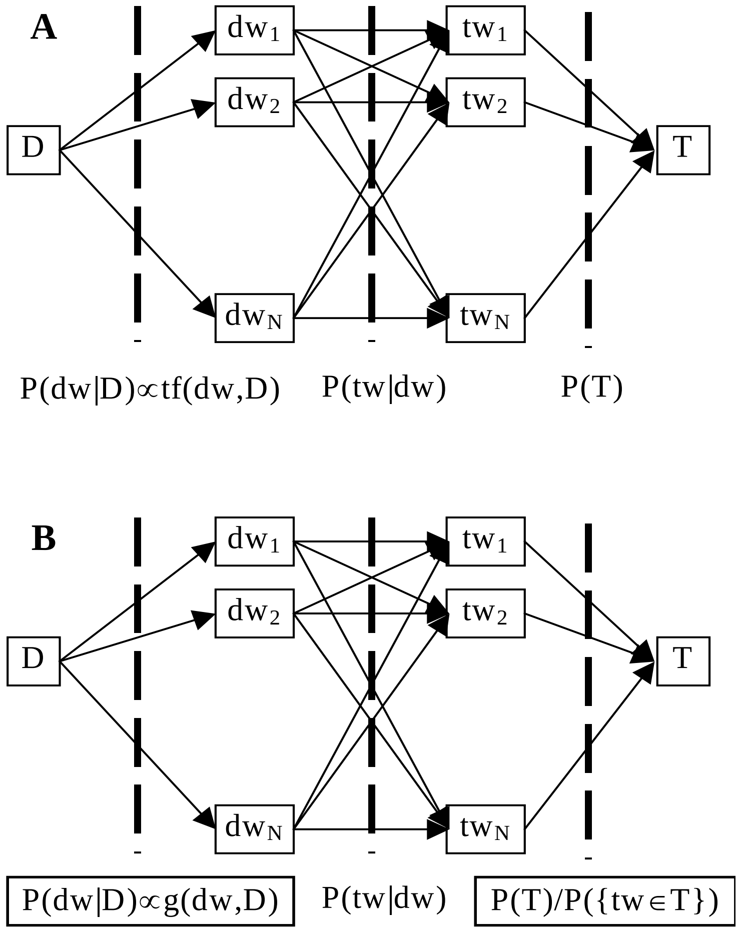

Figure 5.3: Comparison of the old framework for title generation to the ‘direct model with dual noisy channels’. Diagram ‘A’ represents the scheme for the old framework and ‘B’ represents the scheme for the new framework. The difference between them has been highlighted by the rectangles with bold lines.

The schemes for the new model and the previous model are illustrated in Figure 5.3. The contributions of the new models are:

- As illustrated in Figure 5.3, the new model introduces a new sampling distribution g(dw,D) other than the unigram distribution P(dw|D), which is implemented as TF.IDF values of words (or mutual information). Furthermore, for the procedure of ordering title words, the new model uses term $P(T)/P(\{tw \in T\})$ instead of P(T), which is able to alleviate the problem of P(T) in favoring common title words.

- For the new model, we examine various kinds of methods to estimate the sampling distribution g(dw,D), document-word-title-word correlation P(tw|dw), title language model P(T) and probabilities for word sets $P(\{tw \in T\})$. Empirical results suggest that using the TF.IDF values (or mutual information) for g(dw,D), the statistical translation model for P(tw|dw), the expanded title language model for P(T) and the order expansion method for $P(\{tw \in T\})$ achieves the best performance.

Now let’s compare the best configuration of the previous model to the best configuration of the new model. The setup of the experiment is similar to the previous section. As stated in chapter 4, the best configuration of the old model is to use a statistical translation model for title word selection and a bigram language model for title word ordering. The best configuration of the new model has already been discussed in the previous paragraphs. Similar to the previous experiments, both models are trained over 40,000 document-titles from the 1997 broadcast news (Primary Media Source, 1997) and tested against the 10,000 document-titles from the same broadcast news corpus. Titles with six words are generated for each method. F1 score is used as the automatic evaluation metric. Furthermore, out of the 10,000 test documents, 200 documents are randomly selected and titles of those selected documents sent for human judgments. The F1 results and the averaged scores of human judgments for both models are listed in Table 5.6. As shown from Table 5.6, the new model is able to outperform the old model substantially in terms of both F1 scores and human judgments.

Table 5.6: F1 scores and human judgments for the best configuration of the old model and the new model

	Old Model	New Model
F1	0.223	0.280
Human Judgments	2.5	2.9

In order to further illustrate the advantage of the proposed method versus the old method, we list a set of titles generate by the best configuration of the new model in Table 5.7. The corresponding titles generated by the best configuration of the old framework have already been listed in Table 4.13. The corresponding reference titles are listed in Table 4.14.

Table 5.7 Example titles generated by the direct model with dual noisy channels (DM) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	**A Direct Model with Dual Noisy Channels (DM)**
1	green cards illegal immigration card fraud
2	mossad scrutiny classics violinist classical music
3	jury selection oklahoma city bombing trial

4	stock market news morning business report
5	solving mir repair space station computer

5.7 Why not Use the Model for Automatic Speech Recognition?

One interesting question that many people may ask is why not directly use the model for automatic speech recognition (ASR) because both tasks try to estimate similar types of information, namely the most likely word sequence based on some observed evidence. The task of automatic speech recognition is to find the most likely word sequence based on audio signals while the task of automatic title generation is to find the most likely title word sequence based on the observed document. Therefore, the goal of these two tasks is to estimate P(T|E), where T stands for a word sequence and E stands for observed evidence. The only difference between them is in the evidence E, which are the audio signals for automatic speech recognition and documents for title generation.

But, despite the similarity between these two problems, we took an approach for automatic title generation different from that of automatic speech recognition. For automatic speech recognition, instead of estimating P(T|E) directly, people usually use the Bayesian rule to convert P(T|E) to P(E|T)P(T) and estimate the first part of the product, i.e., P(E|T), with an acoustic model and the second part of the product , i.e., P(T), with a statistical language model. If we apply the same idea to title generation, we will need to estimate P(D|T)P(T). Unfortunately, this simple approach is not appropriate for automatic title generation due to the difficulty in estimating P(D|T). The reason can be explained as follows: since a document contains much more detailed information than its title, inferring a title out of a document will be much more certain than inferring a document out of a title. In other words, the variance in estimating P(D|T) will be considerably larger than the variance in estimating P(T|D). This issue can be seen more clearly if we expand the conditional probability P(D|T) as a product of P(dw|T) for all the words ‘dw’ in the documents, i.e., $P(D\,|\,T) \propto \prod_{dw \in D} P(dw\,|\,T)$. Then, the number of terms P(dw|T) used for estimating logP(D|T) is equal to the length of the document D, i.e., |D|. On the other hand, for the proposed model for automatic title generation, probability P(T|D) is used and the number of terms P(tw|D) involved in estimating logP(T|D) is equal to the length of the title, i.e., |T|. If we assume that the variances in estimating P(tw|D) and P(dw|T) are similar and each estimation is independent from others, we can see that the variance in estimating logP(D|T) will be around |D|/|T| times larger than that of logP(T|D). Because the document length is usually much longer than the title length, the variance in the estimation of logP(D|T) will be much larger than that of logP(T|D). As a result, inferring document D out of title T will be much more uncertain than inferring title T out of document D. Of course, this simple illustration may be a little bit naive. But at least it indicates the danger of applying the model of automatic speech recognition to automatic title generation.

In order to prove the above point, we implemented the automatic speech recognition model for title generation and compared to the proposed model over the same dataset as previous experiments. For easy reference we refer to the model for title generation based on the idea of speech recognition as the ‘ASR-based model’ and the proposed model as the ‘**direct model with dual noisy channels**’ because the proposed model doesn’t apply the Bayesian rule to reverse the direction of inference and a dual noisy channel is used to represent the process of generating titles from documents. For the ASR-based title generation model, the problem of estimating P(T|D) is converted into the estimation of P(D|T)P(T). For implementation, we choose the statistical translation model for estimating P(D|T) and the expanded title language model for estimating P(T). Both models are asked to generate titles with six words. Similar to the set-up of previous experiment, 200 documents are randomly selected from the 10,000 test documents and titles of those selected documents are sent for human judgments. The F1 results and the averaged score of human judgments are listed in Table 5.8.

Table 5.8: Results for the ASR based model for automatic title generation compared to the results for 'direct model with dual noisy channels'.

	Direct Model with Dual Noisy Channels	ASR based Model for Title Generation
F1	0.280	0.230
Human Judgment	2.9	2.0

As indicated in Table 5.8, even though the ASR-based model for title generation appears to be effective in terms of the F1 score, it does poorly from the viewpoint of human judgments. A good F1 score implies that the ASR based title generation model is able to capture the important content words while the poor performance in human judgments implies that the sequences generated by the ASR based model may have a very poor readability. The reason can be explained by the large variance in estimating P(D|T). The large variance in estimating P(D|T) causes the scores of title words, i.e., P(D|tw), to have a skewed distribution. As a result, term P(D|T) is more influential than P(T) in composing titles, and therefore the generated titles are less readable. In order to illustrate the difference between these two models visually, we list samples of titles created by the ASR-based approach in Table 5.9. The corresponding titles generated by the 'direct model with dual noisy channels' are listed in Table 5.7 and the reference titles for the same set of documents are listed in Table 4.14.

Table 5.9 Example titles generated by the ASR based model (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	ASR based Model
1	illegal immigration cards immigrants texas card
2	tango music violinist tang kreme kremes
3	oklahoma bombing trial mcveigh's jurors mcveigh
4	stock market stocks tax dow moneyline
5	russian mir space morning computer versace

5.8 How to Determine the Length of Title

In the previous discussion of automatic title generation, we always assume the length of generated titles is fixed to be the average length of training titles. This is not desirable since some documents can be summarized into titles with only one word and other documents may need titles of many words. Particularly, in order to fill out the predefined title length, the model may have to come up with words that do not stick with other selected title words smoothly, which can degrade the quality of the generated titles.

One simple idea for automatically determining the title length is to weight titles of different lengths using the distribution of title length P(m), and a title that maximizes both P(T|D) and P(|T|) is chosen as the final title. Unfortunately, this simple idea could not work due to the flat distribution of title length. We fit the length distribution of training titles using a Gaussian distribution. The resulting mean is 5.8 words per title and the resulting standard deviation is 3.1 words. With this Gaussian distribution, titles with the length ranging from 3 to 9 are almost equally likely.

Here, we proposed another simple strategy for determining the title length which appears to work well in our empirical study. The task of automatically determining title length is equivalent to the task of determining where we should put the end of sentence mark ‘</s>’ into the title. Therefore, we can treat the end of sentence mark ‘</s>’ as another word token and allow ‘s>’ to be added into titles as a normal title word. For the title word selection phase, we always set the conditional probability P(‘</s>’|D) to be 1 and therefore the end of sentence symbol will always be used as a title word candidate. For the title word ordering phase, a bigram title language model including the end of sentence symbol will be trained based on the training corpus. To generate a title for a document, we apply the same algorithm for title word order as before until we reach the maximum title length (notice that not only the normal title words will be ordered by the bigram title language model, but also the end of sentence symbol). Then, the title with highest likelihood will be selected and words after the end of sentence symbol is removed. The leftover word sequence becomes the final title. As can be seen from the above description, with the introduction of the end of sentence symbol, we are able to find the appropriate position for the generated title.

We test the effectiveness of this simple algorithm over an experiment similar to the ones in the previous sections. The ‘direct model with dual noisy channels’ with the modification described as above is used for creating titles of variable length. The maximum title length in this experiment is set to be 10. The F1 score and the averaged human judgments for titles of fixed length and titles of variable length are listed in Table 5.10.

Table 5.10: F1 scores and human judgments for titles of variable length (Variable Title Length) and titles of fixed length (i.e., Title Length = 6)

	Title Length = 6	Variable Title Length
F1	0.280	0.281
Human Judgment	2.9	3.1

According to Table 5.10, even though the proposed method for automatically determining title length doesn't improve the F1 score from the previous algorithm, it does improve the human judgments noticeably from 2.9 to 3.1. The reason can be explained as follows: by automatically determining the title length, the algorithm doesn't improve its ability on title word selection and ordering. However, it does improve the readability by avoid unnecessary long titles. Cases such as including words only for the purpose of filling extra title word slots have been eliminated substantially. Actually, the average length of the generated titles using the method of automatically determining title length is only about 4, which is significantly less than the predetermined length, i.e., 6. In order to better illustrate this point, the sample titles generated by this method are listed in Table 5.11. Titles of fixed title length (equal to 6) are listed in Table 5.7 and the reference titles for the same set of documents are listed in Table 4.14. From these sample titles, we can see that some titles generated by the method with a fixed title length contain title words that do not stick with other title words smoothly and may be chosen only for the purpose of filling out the predefined title length. In contrast, the corresponding titles generated by the method that allows for flexible title length do not seem to have such a problem. For example, the last title created by the method with fixed title length is 'solving mir repair space station computer'. The last word 'computer' appears to be an unnecessary extra word. As a contrast, for the method that allows for variable title length, the created title 'russian space station mir problems' appears to be smoother.

Table 5.11 Example titles generated by the direct model with dual noisy channels (DM) with variable title length (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 4.14 and the original documents are included in the Appendix.

ID	A Direct Model with Dual Noisy Channels for Variable Title Length
1	stopping illegal immigration

2	classical music
3	continuing coverage timothy mcveigh trial jury selection oklahoma city bombing
4	stock market strategist discusses wall street news business report
5	Russian space station mir problems

5.9 Empirical Studies with the AP88 Collection

In the previous sections, all the experiments are tested against a single dataset, i.e., the CD of 1997 broadcast news. In this section, we will compare the proposed title generation model to other methods for automatic title generation over another dataset, the AP88 dataset (Associated Press, 1998). The AP88 dataset consists of 75,000 document-titles. Similar to the split used for 1997 broadcast news corpus, for every five documents we randomly select one as the test document and the leftover four as the training documents. As a result, we have 60,000 training documents and 15,000 test documents. Each method of automatic title generation is asked to create titles with six words. Stopwords are removed from all generated titles.

Four methods are compared in this experiment. They are the nearest neighbor approach (NN), the Naïve Bayes method with a limited vocabulary (NBL), the best configuration of the old framework for title generation (i.e., using the statistical translation model for estimating P(tw|dw) and a bigram statistical language model for estimating P(T)) (ST), and the direct model with dual noisy channels (DM). The reasons for choosing these three methods for comparison are:

1) The NBL method is the first statistical method proposed for automatic title generation. It is always worthwhile to compare the new model to the NBL method.

2) According to the previous study, the Nearest Neighbor method is a simple and effective method for automatic title generation even though it cannot create new titles. Unlike other methods for automatic title generation, the NN method doesn’t rely on the statistical n-gram language model to order title words because the title of the training document that is most similar to the test document is chosen as the generated title for the test document.

3) By comparing the direct model with dual noisy channels to the best configuration of the old framework, we can see whether the introduced new components, such as P({tw∈T}) and the dual noisy channel, are actually effective or not.

The F1 scores and human judgments for the four methods are listed in Table 5.12. As indicated from Table 5.12, the proposed model ‘DM’ is still able to achieve the best performance compared to the other methods, with the F1 score as 0.263 and the human judgment as 2.8. This fact indicates that the proposed model is in general a better model compared to the old framework for title generation. To get a sense of machine-generated titles, we listed five title examples created by each method in Table 5.13 to Table 5.16. The reference titles for the same set of documents are listed in Table 5.17.

Table 5.12: F1 scores and human judgments for the nearest neighbor approach (NN), the Naïve Bayes approach with a limited vocabulary (NBL), the best configuration for the old framework of title generation (ST) and the proposed model (DM) over the AP88 collection.

	NBL	NN	ST	DM
F1	0.175	0.201	0.216	0.263
Human Judgments	2.1	2.6	2.3	2.8

Table 5.13 Example titles generated by the Naïve Bayes method with a limited vocabulary (NBL) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 5.17.

ID	NBL
1	takeover bid black decker buyout offer
2	car bomb attack leftist guerrillas kidnap
3	construction workers injured fire high winds
4	court orders delay mecham impeachment trial
5	stocks finish york stock index decline

Table 5.14 Example titles generated by the nearest neighbor approach (NN) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 5.17.

ID	Nearest Neighbor (NN)
1	black & decker rebuffed american standard takeover bid
2	guerrillas kidnap mayors
3	boarding house fire kills six, injures canary islands
4	impeachment trial process embroiled controversy
5	york: advance estimates.

Table 5.15 Example titles generated by the statistical translation model (ST) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 5.17.

ID	Statistical Translation Model (ST)
1	black decker buyout offer american standard
2	mayors caller leftist guerrillas kidnap bjt
3	construction workers injured building collapses bjt
4	mecham criminal trial delay impeachment bjt
5	london shares close higher stocks Tokyo

Table 5.16 Example titles generated by the direct model with dual noisy channels (DM) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 5.17.

ID	DM
1	black & decker financing american standard bid
2	leftist guerrillas kidnap mayors
3	construction workers injured building collapses
4	arizona governor mecham criminal trial delay impeachment
5	york stock market

Table 5.17 Examples of reference titles for the AP88 documents (Notice that all the stopwords are removed.)

ID	Reference Titles
1	black decker financing american standard bid
2	mayors reporters kidnapped guerrillas
3	wind blows wall injuring 10 workers
4	governor's lawyer seeks delay impeachment trial
5	york market

5.10 Empirical Studies with Heterogeneous Datasets

In the previous experiments, we intentionally let training data be similar to testing data by randomly selecting four out of every five consecutive documents as training ones and the leftover one as a testing document. As a result, a simple nearest neighbor approach performs fairly well even compared to other sophisticated learning algorithms. In this subsection, we will consider another type of setup of experiments, i.e., testing data are heterogeneous from training data. More specifically, for the AP88 collection, we use the news stories of the first ten months of 1988 as the training data and the news stories of last two months as the testing data. Because there are many news stories that only appeared in November and December of 1988 and were not mentioned at all in the first ten months of 1988, we expect that the nearest neighbor approach would work poorly for the heterogeneous testing data. Unlike the nearest neighbor approach which simply reuses one of the training titles as the title for the test document, the proposed algorithm relies the title word statistics to select appropriate title words for the test document and order the selected title words into the title sequence. Therefore, it should be robust even for the heterogeneous testing datasets. The total number of training documents is 63,000 and the number of testing documents is 12,000. Both the nearest neighbor approach (NN) and the direct model with dual noisy channels are examined in this experiment. The F1 scores of both approaches are listed in Table 5.18. Meanwhile, the F1 scores with the homogeneous testing dataset over AP88 collection (e.g., the F1 scores of the previous experiment in Section 5.9) are also listed in Table 5.18. As indicated from Table 5.18, the F1 score of the nearest neighbor approach degrades substantially, dropping from 0.201 for the homogeneous case to 0.119 for the heterogeneous case. On the other hand, the proposed model is quite robust to the heterogeneous testing dataset, whose F1 score changes much less substantially than that of the nearest neighbor approach, only from 0.263 to 0.221.

Table 5.18: F1 scores for the nearest neighbor approach (NN) and the direct model with dual noisy channels (DM) for the case of homogeneous testing data and the case of heterogeneous testing data.

	NN	DM
Heterogeneous Case	0.119	0.221
Homogeneous Case	0.201	0.263

Chapter 6

AUTOMATIC TITLE GENERATION FOR NOISY DOCUMENTS

In the previous chapters, we have examined the effectiveness of automatic title generation model for documents written by people. In this chapter, we are considering another type of documents, i.e., machine-generated documents, such as speech-recognized documents, machine-translated documents and optical character recognized (OCR) documents. Compared to the 'true' documents written by human subjects, these machine-generated documents are considerably more erroneous. For example, using the Sphinx system, the word error rates of speech recognized documents amount to 30% for broadcast news if the system is not specially trained to fit the voices of anchors and contents of the news. These noises within the machine-generated documents will impose a great challenge on the automatic title generation task. In this chapter, we will examine the robustness of the statistical model for title generation model for speech-recognized documents and machine-translated documents.

6.1 Title Generation for Speech Recognized Documents

The goal of this task is to automatically generate titles for speeches. To accomplish it, we need to first 'translate' a speech into a sequence of words using speech recognition technology, and then apply the automatic title generation model to create a title for the recognized speech transcripts. Due to the limited size of vocabulary, the ambiguity in the voice and tone of the speaker and the limited amount of training data, the speech recognized transcripts usually contain a large number of word errors. In this section, we are going to examine how robust the statistical models for automatic title generation are to the noises introduced by the speech recognition process.

6.1.1 Experiment Design

For this experiment, we use a different corpus since we don't have the speech data for the broadcast news of 1997. Our training set, consisting of 21190 perfectly transcribed documents, is obtained from the CNN.com web site during 1999. Included with each training document text is a human-assigned title. The test set, consisting of 1006 CNN TV news story documents for the same year (1999), are randomly selected from the Informedia Digital Video Library (http://www.informedia.cs.cmu.edu). Each document has a closed captioned transcript, an alternative transcript generated with the CMU Sphinx speech recognition system using a 64000-word broadcast news language model and an original title provided by CNN. The word error rate of the speech recognition transcripts is around 35%. In order to see how the word errors in the spoken documents affect the performance of our title generation methods, we train the title generation models over the 21190 perfectly transcribed documents and their titles, and apply the models to create titles for both close captioned transcripts and speech-recognized transcripts.

The title generation methods that we will examine in this experiment are: The term frequency and inverse document frequency approach (TF.IDF), the nearest neighbor approach (NN), the Naïve Bayes approach with the limited vocabulary (NBL) and the proposed 'direct model with dual noisy channels' (DM). The description of each algorithm can be found in previous chapters. The reasons for choosing these four methods are due to their different characteristics:

- Compared to the other three approaches, the TF.IDF approach is the only approach that has no consideration of the correlation between titles and documents. It simply selects words with high TF.IDF scores and organizes them into a sequence.

- Compared to the other three approaches the nearest neighbor approach (NN) is the only approach that doesn't rely on the title language model for ordering words. Instead, it simply finds the document within the training corpus that is most similar to the testing one and uses its title as the title for the testing document.

- Both the Naïve Bayes approach with limited vocabulary (NBL) and the 'direct model with dual noisy channels' (DM) use the statistics on the title_word-document_word correlation as the basis for selecting title words, and rely on the title language model to organize the word sequence. In the previous study of automatic title generation, we have shown that the proposed model outperforms the NBL approach significantly. In this experiment, we will examine how the word errors in the spoken documents can influence the performances of these two learning approaches.

Each method is asked to generate a title with six words. F1 is used as the evaluation metric.

6.1.2 Results and Discussion

F1 results for the four different title generation methods are shown in Figure 6.1. For each method, example titles generated for both the original documents and the spoken documents are listed in Table 6.1 to Table 6.4. The reference titles are listed in Table 6.5. According to the results shown in Figure 6.1, we can draw the following conclusions:

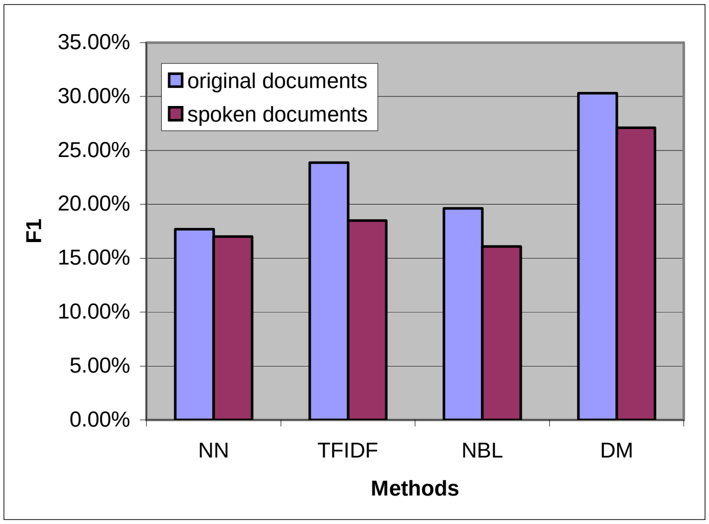

Figure 6.1: F1 scores of machine-generated titles for both close captioned transcripts and speech recognized transcripts. Results for four different methods are listed. They are: a nearest neighbor approach (NN), a term frequency and inverse document frequency approach (TF.IDF), a Naïve Bayes approach with limited vocabulary (NBL) and the direct model with dual noisy channels (DM).

- **The nearest neighbor approach (NN) is quite robust**. Compared to other approaches, the nearest neighbor approach (NN) achieves the least degradation, going from 0.177 for close captioned transcripts to 0.172 for the speech recognized transcripts. We believe it is due to the fact that the document-document similarity is computed based on the overall word matches between two documents. Therefore, even though the word error rate of the spoken documents is about 35%, the majority of the words in most documents are still correct and thus, the document similarity measurement can be robust to even quite large word error rates.

- **The term frequency and inverse document frequency approach (TF.IDF) obtained the largest degradation**. For the case of perfectly transcribed documents, the simple TF.IDF approach performs quite well, only secondary to the direct model with dual noisy channels (DM). However, for the case of spoken documents, the TF.IDF approach suffers the largest loss in F1, going from 0.239 to 0.185. Since the TF.IDF approach is an extraction-based approach and can only use the words in the test document to form the title, it will surely be influenced by the word errors in spoken documents. Particularly, words with high TF.IDF values tend to be not common words and therefore have more chance to be misrecognized than common words. This fact makes the situation even harder for the TF.IDF approach.

- **The Naïve Bayes approach with limited vocabulary approach (NBL) achieves the worst performance for the speech recognized transcripts**. This approach doesn't perform well for the spoken documents. As a matter of fact, it has the worst performance among the four different methods for the spoken documents, with F1 of 0.162. Again, similar to the analysis for TF.IDF, we think it is due to the fact that the NBL approach limits the choice of title words to be the words appearing in the test document and therefore is essentially an extraction-based approach. When the important content words are mis-recognized, the extraction-based approach will have difficult time in finding good candidates for title words.

- **The new model performs well in both cases**. The direct model with dual noisy channels achieves the best performance for both closed captioned transcripts and the speech recognized transcripts. Similar to the analysis for the nearest neighbor approach, the new model for title generation is able to select good title words based on the overall opinions of all the words within the documents, and therefore is quite resilient to large word error rates as long as the majority of the words within the document are still correct. Meanwhile, unlike both the TF.IDF and the NBL approach where the title words are chosen from the words in the documents, the new model allows any word in the vocabulary to be a title word candidate. Therefore, it is robust to the case even when some important content words in a document are misrecognized.

Table 6.1 Example titles generated by the nearest neighbor approach (NN) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 6.5.

ID	Original Docs	Spoken Docs
1	experts warn dolphins dangerous humans death linked seaworld's tillikum golf courses safe havens wildlife	experts warn dolphins dangerous humans death linked seaworld's tillikum golf courses safe havens wildlife
2	snowfall midwest delivers dream white christmas hampers travel	snowfall midwest delivers dream white christmas hampers travel
3	colorado school shooting teacher three more students buried today	colorado school shooting michael thompson discusses boys bullies school violence
4	adoption tennessee massachusetts cases open records increase contact biological parents	adoption tennessee massachusetts cases open records increase contact biological parents
5	columbine students start school year positive note	san francisco teenager delivers $18,000 columbine victim

Table 6.2 Example titles generated by the TFIDF approach (TFIDF) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 6.5.

ID	Original Docs	Spoken Docs
1	experts warn dolphins dangerous humans wild	san francisco dolphins dangerous animals wild
2	canceled flights chicago aviation o'hare airport	airline delays air travel flights chicago
3	federal investigators world church shooting spree	federal authorities confirm received shooting spree
4	human rights supreme court judge rule	supreme court judge york senate candidate
5	san francisco airport security threats violence	san francisco workplace violence schools safe

Table 6.3 Example titles generated by the Naïve Bayes approach with a limited vocabulary (NBL) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 6.5.

ID	Original Docs	Spoken Docs
1	killer whale dolphins dangerous san Francisco	health care bill clinton signs life
2	chicago airports airlines flight delays expected	travel delays air force airlines flight
3	federal investigators world church shooting spree	murder case federal investigation shooting spree
4	supreme court judge rules parents children	state supreme court judge york senate
5	san francisco airport high school shooting	san francisco airport high school shooting

Table 6.4 Example titles generated by the direct model with dual noisy channels (DM) (Notice that all the stopwords are removed). The reference titles for the same set of documents are listed in Table 6.5.

ID	Original Docs	Spoken Docs
1	animal experts warn dolphins dangerous humans	health experts warn dolphins dangerous humans
2	winter storm american airlines flight delays	winter storm american airlines flight delays
3	authorities arrest high school shooting spree	high school shooting gun control legislation
4	supreme court hear boy custody case	supreme court decision federal judge rules
5	columbine high school shooting gun control	columbine high school shooting gun control

Table 6.5 Examples of reference titles for the CNN TV news (Notice that all the stopwords are removed.)

ID	Reference Titles
1	experts warn dolphins dangerous humans
2	storm causes travel delays ohio
3	schoolchildren attack look patterns school shooting incidents
4	legal fray begins custody elian Gonzalez
5	columbine students prepare take back school

6.2 Title Generation for Machine Translated Documents

Similar to speech recognized transcripts, machine translated documents are also quite noisy. The errors in a machine-translated document include incorrect word translations, syntactic errors and ungrammatical structures. In this section, we will examine how the errors in the machine-translated documents influence the performance of different title generation models.

6.2.1 Experiment Design

The experimental dataset comes from a CD of 1997 broadcast news transcriptions published by Primary Source Media (1997), as do most other experiments conducted in this thesis. There were a total of roughly 50,000 news documents and corresponding titles in the dataset. The training dataset was formed by randomly picking four documents-title pairs from every five pairs in the original dataset. The size of training corpus was therefore 40,000 documents and their titles. We fixed the size of the test collection at 1000 items from the unused document-title pairs.

Since we did not have a large set of documents with titles in multiple parallel languages, we approximated this by creating machine-translated documents from the original 1000 documents with titles as follows: Each of the 1000 test documents was submitted to the SYSTRAN machine translation system (http://babelfish.altavista.com) and translated into French. The French translation was again submitted to the SYSTRAN translation system and translated back into English. This final retranslation resulted in our French machine translation data. The procedure was repeated on the original documents for translation into Portuguese and German to obtain two more machine translated sets of identical documents. For all languages, the average word overlap between the translated documents and the original documents was around 70%.

Similar to the experiment done for the spoken documents, we choose the same four title generation methods, namely the term frequency and inverse document frequency approach (TF.IDF), the nearest neighbor approach (NN), the Naïve Bayes approach with a limited vocabulary (NBL) and the direct model with dual noisy channels (DM). Each method is asked to create titles of six words. F1 is used as the evaluation metric.

6.2.2 Results and Discussions

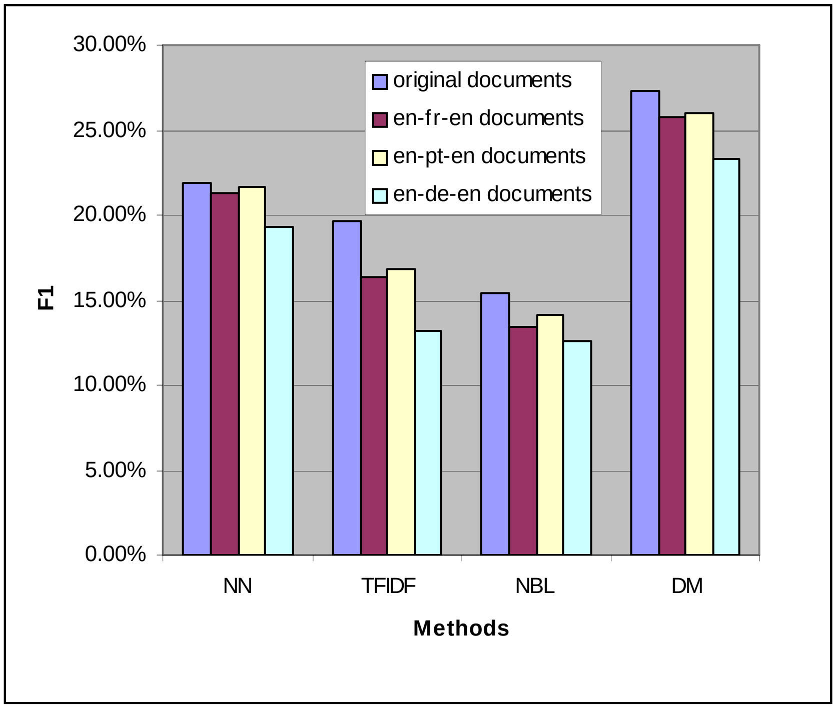

Figure 6.2: F1 scores for the machine-generated titles. For each method, there are four bars representing the F1 scores for the titles generated from the original documents and the translated documents. The legends en-fr-en, en-pt-en and en-de-en represent the documents translated from English document to French, Portuguese and German and back in English, respectively.

F1 scores of machine-generated titles for all four methods are shown in Figure 6.2. The legend 'en-fr-en documents' stands for the documents that are first translated from English to French and then re-translated from French to English. Similarly, the legends 'en-pt-en documents' and 'en-de-en documents', stand for the translated documents through Portuguese and German, respectively. For each method, the sample titles for the original documents, the 'en-pt-en documents', the 'en-de-en documents' and the 'en-fr-en documents', are listed in Table 6.6 to Table 6.9. The reference titles for the same set of documents are listed in Table 6.10.

According to Figure 6.2, we can draw the following conclusions:

- **Performance of French and Portuguese corpus is substantially better than the corpus of German.** For all four different methods, according to Figure 6.2, the F1 scores for the corpuses that are translated from English to French and Portuguese and back to English are considerably better than the corpus that is translated from English to German and back to English. This phenomenon can be explained by the fact that French, Portuguese and English are quite similar languages while German is significantly different from these three languages. For example, German contains many compound words and each of them corresponds to the meaning of several individual words in other languages. With this difference in mind, we would expect the German corpus to contain more translation mistakes than the corpuses of the other two languages. As a result, the quality of machine-generated titles for the German corpus is worse than the corpuses of either French or Portuguese.

- **The Nearest Neighbor approach (NN) is quite robust to the machine-translation errors.** According to Figure 6.2, for the nearest neighbor approach, the F1 scores for French and Portuguese corpuses are only slightly worse than the F1 scores for the original English documents, degrading from 21.93% to 21.65% for the Portuguese corpus and 21.25% for the French corpus. Similar to the analysis for the case of spoken documents, we believe the robustness of the nearest neighbor approach for machine-translated documents is due to the fact that the document-document similarity is determined by the overall word matches bewteen two documents. Therefore, even though 30% of the English words are missed during the machine translation process, the majority of English words are still being kept, which makes the similarity measurement reasonably reliable even for machine-translated documents.

- **The TF.IDF approach suffers the largest degradation**. For the TF.IDF approach, the F1 scores for the Portuguese corpus and the French corpus are much worse than the F1 score for the original English corpus, dropping from 19.87% to 16.83% for the Portuguese corpus and to 16.30% for the French corpus. Again, a similar analysis applied to the spoken documents can be used here, namely the TF.IDF approach is an extraction-based approach and therefore is very sensitive to the word errors in the documents, particularly when the important content words are mistranslated.

- **The NBL has the worst performance among the four methods**. Slightly different from the case of spoken documents, where the NBL approach performs slightly better than the nearest neighbor approach for the perfectly transcribed documents, the NBL approach has the worst F1 scores for both the original English documents and the documents that are translated through Portuguese, French and German. Similar to the analysis for spoken documents, the poor performance of the NBL approach can be attributed to the fact that the NBL approach only considers the words in the documents as the candidates for title words and therefore is very sensitive to the word errors in documents.

- **The new model significantly outperforms the other title generation approaches**. According to Figure 6.2, the direct model with dual noisy channels achieves the best performance for both the original English corpus and the three different translated corpuses. A similar explanation for speech recognition transcripts can be applied here, i.e., the new model is a non-extraction based approach and allows any word to be used as title words. Furthermore, the new model is able to take into account all the document words as evidence in deciding title words and therefore is less sensitive to the word errors in documents.

6.3 Conclusions

According to the previous analysis, we can see that the four different methods show similar patterns for speech recognized documents and machine-translated documents. Usually, the nearest neighbor approach is quite robust to both the translation errors and the speech recognition errors because the document-document similarity depends on the overall word matches between documents and therefore is resilient to even large word error rates in documents. On the other hand, both the TF.IDF method and the NBL method are quite sensitive to word errors due to the extraction nature of these two approaches. Finally, the new model is able to substantially outperform the other three methods for both the speech recognized documents and machine-translated documents because the new statistical model is able to take into account all the document words for selecting title words and the candidates of title words are not restricted to the words appearing in the tested document. Therefore, we conclude that the proposed model for title generation, i.e., ‘direct model with dual noisy channels’, is resilient to the word errors in the documents.

Table 6.6 Example titles generated by the nearest neighbor approach (NN) for the original documents, the ‘en-fr-en’ documents, the ‘en-pt-en’ documents and the ‘en-de-en’ documents. Notice that all the stopwords are removed.

ID	**Titles for Original Documents**
1	pen phen warning
2	meeting israeli palestinian leaders
3	world balloon quest
4	oil food deal rescue iraq's stagnant economy
5	astronomists continue probe life mars
	Titles for ‘en-fr-en Documents’
1	fda issues warning herbal fen-phen
2	secret meeting arafat netanyahu unsuccessful
3	british tycoon's balloon launch thwarted
4	oil food deal rescue iraq's stagnant economy
5	astronomists continue to probe life on mars
	Titles for ‘en-pt-en Documents’
1	pen phen warning
2	meeting israeli palestinian leaders
3	british tycoon's balloon launch thwarted
4	oil food deal rescue iraq's stagnant economy
5	astronomists continue to probe life on mars
	Titles for ‘en-de-en Documents’
1	pen phen warning
2	hebron
3	british tycoon's balloon launch thwarted
4	oil food deal rescue iraq's stagnant economy
5	astronomists continue to probe life on mars

Table 6.7 Example titles generated by the TFIDF method (TFIDF) for the original documents, the ‘en-fr-en’ documents, the ‘en-pt-en’ documents and the ‘en-de-en’ documents. Notice that all the stopwords are removed.

ID	**Titles for Original Documents**
1	phen fen drug effects drugs disease
2	palestinian cabinet minister netanyahu yasser arafat
3	swiss balloon quest landed safely desert
4	iraq sanctions iraqi oil arab nations
5	life galaxy earth stars pounds universe
	Titles for ‘en-fr-en Documents’
1	drug effects mark drugs disease capacity
2	yasser arafat hebron agreement israel palestinians
3	search mark algerian switzerland balloon quest
4	iraq iraqi sanctions food companies usa
5	life ground east stars usa universe
	Titles for ‘en-pt-en Documents’
1	phen fen effects drugs doctor illness
2	yasser arafat hebron agreement israel palestinians
3	algerian balloon quest teams de stops
4	iraq iraqi sanctions food stands commentary
5	life distant galaxy stars land universe
	Titles for ‘en-de-en Documents’
1	doctor effects drugs illness robbery breath
2	israeli cabinet hebron agreement israel palestinians
3	algerian message places balloon branson continuous
4	iraq iraqi sanctions medicine arab enterprises
5	life span credit mass possibly universe

Table 6.8 Example titles generated by the Naïve Bayes method with a limited vocabulary (NBL) for the original documents, the 'en-fr-en' documents, the 'en-pt-en' documents and the 'en-de-en' documents. Notice that all the stopwords are removed.

ID	**Titles for Original Documents**
1	diet drugs food health news medicine
2	yasser arafat netanyahu ross talks hebron
3	algerian man balloonist richard branson week
4	iraq oil food deal iraqi crisis
5	life mars scientists religious civil earth
	Titles for 'en-fr-en Documents'
1	health news doctors weight rubin hypertension
2	israelis palestinians yasser arafat hebron agreement
3	world news morning branson balloon flight
4	iraq food medicine coverage iraqi crisis
5	scientists world science olympic astronomers galactic
	Titles for 'en-pt-en Documents'
1	doctors diet drugs food health medicine
2	minister netanyahu arafat hebron agreement israel
3	balloonist richard branson world balloon week
4	iraq oil food medicine business report
5	life mars landing religious civil land
	Titles for 'en-de-en Documents'
1	food health people dead man hearings
2	israeli palestinian cabinet hebron agreement israel
3	algerian man news week house industry
4	russian space pair murder man strike
5	oil food medicine market economy companies

Table 6.9 Example titles generated by the direct model with dual noisy channels (DM) for the original documents, the 'en-fr-en' documents, the 'en-pt-en' documents and the 'en-de-en' documents. Notice that all the stopwords are removed.

ID	**Titles for Original Documents**
1	fda diet drug fen phen redux fenfluramine heart disease
2	israeli palestinian peace middle east jerusalem
3	world ballon quest
4	iraq oil food deal
5	life mars probe
	Titles for 'en-fr-en Documents'
1	fda diet drug fen phen redux heart disease
2	israeli palestinian peace talks middle east jerusalem
3	flight world balloon quest
4	iraqi crisis
5	scientists discusses universe mars probe
	Titles for 'en-pt-en Documents'
1	diet drug fen phen health medicine
2	israeli palestinian peace middle east
3	branson's beautiful balloon quest
4	iraqi sanctions iraq oil food deal
5	scientists universe life mars probe
	Titles for 'en-de-en Documents'
1	risks diet drugs heart treatment
2	middle east peace process west bank
3	balloon race reno testimony
4	iraqi crisis iraq
5	scientists discusses universe mars

Table 6.10 Examples of reference titles for the experiment of automatic title generation with machine-translated documents. Notice that all the stopwords are removed.

ID	**Reference Titles**
1	fen phen primary pulmonary hypertension
2	update night's netanyahu arafat meeting
3	world balloon quest incomplete
4	iraq dealing international sanctions

5	scientists search life planets

Chapter 7

A TITLE GENERATION MODEL FOR INFORMATION RETRIEVAL

In this chapter, we will discuss how to apply the title generation model to information retrieval. The basic idea is to use the title generation model for exploiting the correlation between document words and title words. Unlike the traditional language model, where a document language model is built for each document, this model tries to build an approximated query language model for every document. Furthermore, at the end of this chapter, we will briefly study another application of the title generation model, namely applying the automatic title generation model to text categorization.

7.1 Introduction

Using language models for information retrieval has been studied extensively recently (Lafferty & Zhai, 2001a; Zhai & Lafferty, 2001b; Berger & Lafferty, 1999; Hiemstra & Kraaij, 1999; Miller et al., 1999; Ponte & Croft, 1998; Voorhees & Harman, 1996). The basic idea is to compute the conditional probability $P(Q|D)$, i.e., the probability of generating a query Q given the observation of a document D. Several different methods have been applied to compute this conditional probability. In most approaches, the computation is conceptually decomposed into two distinct steps: (1) Estimating a document language model; (2) Computing the query likelihood using the estimated document model based on some query model. For example, Ponte and Croft (1998) emphasized the first step, and used several heuristics to smooth the Maximum Likelihood Estimate (MLE) of the document language model, and assumed that the query is generated under a multivariate Bernoulli model. The

BBN method (Miller et al., 1999) emphasized the second step and used a two-state hidden Markov model as the basis for generating queries, which, in effect, is to smooth the MLE with linear interpolation, a strategy also adopted in Hiemstra and Kraaij (1999). In Zhai and Lafferty (2001b), it has been found that the retrieval performance is affected by both the estimation accuracy of document language models and the appropriate modeling of the query, and a two-stage smoothing method was suggested to explicitly address these two distinct steps.

A common deficiency in these approaches is that they all apply an estimated document language model directly to generating queries, but presumably queries and documents should be generated through different stochastic processes, since they have quite different characteristics. Therefore, there exists a "gap" between a document language model and a query language model. Indeed, such a gap has been well recognized in (Lafferty & Zhai, 2001a), where separate models are proposed to model queries and documents respectively. The gap has also been recognized in (Lavrenko & Croft, 2001), where a document model is estimated based on a query through averaging over document models based on how well they explain the query. In most existing approaches using the query likelihood for scoring, this gap has been implicitly addressed through smoothing. Indeed, in (Zhai & Lafferty, 2001b) it has been found that the optimal setting of smoothing parameters is actually query-dependent , which suggests that smoothing may have helped bridge this gap.

Although filling the gap by simple smoothing has been shown to be empirically effective, ideally we should estimate a *query language model* directly based on the observation of a *document*, and apply the estimated query language model, instead of the document language model, to generate queries.

The question then is, “What evidence do we have for estimating a query language model given a document?”. This is a very challenging question, since the information available to us in a typical ad hoc retrieval setting includes no more than a database of documents and queries.

In this section, we propose to use the titles of documents as the evidence for estimating a query language model for a given document -- essentially to approximate the query language model given a document by the title language model for that document, which is easier to estimate. The motivation of this work is based on the observation that queries are more like titles than documents in many aspects. For example, both titles and queries tend to be very short and concise description of information. The reasoning process in the author’s mind when making up the title for a document is similar to what is in a user’s mind when formulating a query based on some “ideal document” -- both would be trying to capture what the document is *about*. Therefore, it is reasonable to assume that the titles and queries are created through a similar generation process. The title information has been exploited previously for improving information retrieval, but so far, only heuristic methods, such as increasing the weight of title words, have been tried (e.g., Lam-Adesina & Jones, 2001; Voorhees & Harm, 1996). Here we use the title information in a more principled way by treating a title as an observation from a document-title statistical translation model.

Technically, the title language model approach falls into the general source-channel framework proposed in Berger and Lafferty (1999), where the difference between a query and a document is explicitly addressed by treating query formulation as a “corruption” of the “ideal document” in the information theoretic sense. Conceptually, however, the title language model is different from the synthetic query translation model explored in (1999). The use of synthesized queries provides an interesting way to train a statistical translation model that can address important issues such as synonymy and polysemy, whereas the title language model is meant to directly approximate queries with titles. Moreover, training with the titles poses special difficulties due to data sparseness, which we discuss below.

A document can potentially have many different titles, but the author only provides one title for each document. Thus, if we estimate title language models only based on the observation of the author-given titles, it will suffer severely from the problem of sparse data. The use of a statistical translation model can alleviate this problem. The basic idea is to treat the document-title pairs as ‘translation’ pairs observed from some translation model that captures the intrinsic document to query translation patterns. This means we would train the statistical ‘translation’ model based on the document-title pairs in the whole collection. Once we have this general translation model in hand, we can estimate the title language model for a particular document by applying the learned translation model to the document.

Even if we pool all the document-title pairs together, the training data is still quite sparse given the large number of parameters involved. Since titles are typically much shorter than documents, we would expect that most words in a document would never occur in any of the titles in the collection. To address this problem, we extend the standard learning algorithms of the translation models by adding special parameters to model the “self-translation” probabilities of words. We propose two such techniques: One assumes that all words have the same self-translation probability and the other assumes that each title has an extra unobserved null word slot that can only be filled by a word generated through self-translation.

7.2 Title Language Model for IR

The basic idea of the title language model approach is to estimate the title language model for a document and then to compute the likelihood that the query would have been generated from the estimated model. Therefore, the key issue is how to estimate the title language model for a document based on the observation of a collection of documents.

A simple approach would be to estimate the title language model for a document using only the title of that document. However, because of the flexibility in choosing different titles and the fact that each document has only one title given by the author(s), it would be almost impossible to obtain a good estimation of title language model directly from the titles.

Our approach is to exploit statistical translation models to find the title language model based on the observation of a document. More specifically, we use a statistical translation model to "convert" the language model of a document to the title language model for that document. To accomplish this conversion process, we need to answer two questions:

1. How to estimate such a statistical translation model?

2. How to apply the estimated statistical translation model to convert a document language model to a title language model and use the estimated title language model to score documents with respect to a query?

In the following section, we are going to address these two questions respectively.

7.2.1 Learning a Statistical Title Translation Model

The key component in a statistical title translation model is the word translation probability P(tw|dw), i.e., the probability of using word tw in the title, given that word dw appears in the document. Once we have the set of word translation probabilities P(tw|dw), we can easily calculate the title language model for a document based on the observation of that document.

To learn the set of word translation probabilities, we can take advantage of the document-title pairs in the collection. By viewing documents as samples of a 'verbose' language and titles as samples of a 'concise' language, we can treat each document-title pair as a translation pair, i.e., a pair of texts written in the 'verbose' language and the 'concise' language respectively.

Formally, let {<ti, di>, i = 1, 2, …, N} be the title-document pairs in the collection. According to the standard statistical translation model (Brown et al., 1990) we can find the optimal model M* by maximizing the probability of generating titles from documents, or

$$M^* = \arg\max_{M} \prod_{i=1}^{N} P(t_i \mid d_i, M) \qquad (7.1)$$

Based on the model 1 for the statistical translation model (Brown et al., 1990), Equation (7.1) can be expanded as

$$\begin{aligned} M^* &= \arg\max_{M} \prod_{i=1}^{N} P(t_i \mid d_i, M) \\ &\approx \arg\max_{M} \prod_{i=1}^{N} \prod_{tw \in t_i} \left\{ \frac{\varepsilon}{|d_i|+1} \left(P(tw \mid \phi, M) + \sum_{dw \in d_i} P(tw \mid dw, M) c(dw, d_i) \right) \right\} \\ &\approx \arg\max_{M} \prod_{i=1}^{N} \prod_{tw \in t_i} \left\{ \frac{P(tw \mid \phi, M)}{|d_i|+1} + \sum_{dw \in d_i} P(tw \mid dw, M) P(dw \mid d_i) \right\} \end{aligned} \qquad (7.2)$$

where ε is a constant, ϕ stands for the null word, $|d_i|$ is the length of document d_i, $c(dw, d_i)$ is the number of times that word dw appears in document d. In the last step of Equation (7.2), we throw out the constant ε and use the approximation that $P(dw \mid d) \approx c(dw, d)/(|d|+1)$. To find the optimal word translation probabilities $P(tw|dw,M^*)$, we can use the EM algorithm. The details of the algorithm can be found in the literature for statistical translation models, such as (Brown et al., 1993). We call this model "model 1" for easy reference.

7.2.1.1 The problem of under-estimating self-translation probabilities

There is a serious problem with using model 1 described above directly to learn the correlation between the words in documents and titles. In particular, the self-translation probability of a word (i.e., P(w'=w|w)) will be under-estimated significantly. A document can potentially have many different titles, but authors generally only give one title for every document. Because titles are usually much shorter than documents, only an extremely small portion of the words in a document can be expected to actually appear in the title. We measured the vocabulary overlapping between titles and documents on three different TREC collections: AP(1988), WSJ(1990-1992) and SJM(1991), and found that, on average, only 5% of the words in a document also appear in its title. This means that most of the document words would never appear in any title, which will result in a zero self-translation probability for most of the words. Therefore, if we follow the learning algorithm for the statistical translation model directly, the following scenario may occur: For some documents, even though they contain every single query word, the probability P(Q|D) can still be very low due to the zero self-translation probability. In the following subsections, we propose two different learning algorithms that can address this problem. As will be shown later, both algorithms improve the retrieval performance significantly over the model 1, indicating that the proposed methods for modeling the self-translation probabilities are effective.

7.2.1.2 Tying self-translation probabilities (Model 2)

One way to avoid the problem of zero self translation probability is to tie all the self translation probabilities P(w'=w|w) with a single parameter Pself. Essentially, we assume that all the self-translation probabilities have approximately the same value, and so can be replace with a single parameter. Since there are always some title words actually coming from the body of documents, the unified self-translation probability P_{self} will not be zero. We call the corresponding model Model 2.

We can also apply the EM algorithm to estimate all the word translation probabilities, including the smoothing parameter P_{self}. The updating Equations are as follows:

Let P(w'|w) and P_{self} stand for the parameters obtained from the previous iteration, P'(w|w) and P'_{self} stand for the updated values of the parameters in the current iteration. According to the EM algorithm, the updating equation for the self-translation probability P'_{self}, will be

$$P'_{self} = \frac{1}{Z_{self}} \sum_i \sum_w \frac{P_{self} C(w, d_i) C(w, t_i)}{P_{self} C(w, d_i) + \sum_{w' \in d_i \wedge w \neq w'} P(w \mid w') C(w', d_i)} \qquad (7.3)$$

where variable Z_{self} is the normalization constant and is defined as

$$Z_{self} = \sum_i \left(\sum_w \sum_{w' \neq w} \frac{P(w \mid w')C(w, t_i)C(w', d_i)}{P_{self} C(w, d_i) + \sum_{w'' \in d_i \wedge w'' \neq w} P(w \mid w'')C(w'', d_i)} + \sum_w \frac{P_{self} C(w, d_i)C(w, t_i)}{P_{self} C(w, d_i) + \sum_{w' \in d_i \wedge w \neq w'} P(w \mid w')C(w', d_i)} \right) \quad (7.4)$$

For those non-self-translation probabilities, i.e., $P(w' \neq w | w)$, the EM updating equations are identical to the ones used for the standard learning algorithm of a statistical translation model except that in the normalization equations, the self-translation probability should be replaced with P_{self}, or

$$\forall w \quad \sum_{w' \neq w} P'(w' \mid w) = 1 - P'_{self} \quad (7.5)$$

7.2.1.3 Adding a Null Title Word Slot (Model 3)

One problem with tying all the self-translation probabilities for different words with a single unified self-translation probability is that we lose some information about the relative importance of words. Specifically, those words with a higher probability in the titles should have a higher self-translation probability than those with a lower probability in the titles. Tying them would cause under-estimation of the former and over-estimation of the latter. As a result, the self-translation probability may be less than the translation probability for other words, which is not desirable.

In this subsection, we propose a better smoothing model that is able to discriminate the self-translation probabilities for different document words. It is based on the idea of introducing an extra NULL word slot in the title. An interesting property of this model is that the self-translation probability is guaranteed to be no less than the translation probability for any other word, i.e., $P(w|w)^3P(w'^1w|w)$. We call this model ‘Model 3’.

Titles are typically very short and therefore only provide us with very limited data. Now, suppose we had sampled more title words from the title language model of a given document, what kinds of words would we expect to have seen? Given no other information, it would be reasonable to assume that we will more likely observe a word that occurs in the document. To capture this intuition, we assume that there is an extra NULL, unobserved, word slot in each title, that can only be filled in by self-translating any word in the body of the document. Use e_t to stand for the extra word slot in the title t. With the count of this extra word slot, the standard statistical translation model between the document d and title t will be modified as

$$\begin{aligned} P(t \mid d, M) &\approx P(e_t \mid d, M) \prod_{tw \in t} P(tw \mid d, M) \\ &\approx \left(\sum_{dw \in d} P(dw \mid dw, M) P(dw \mid d) \right) \times \\ &\prod_{tw \in t} \left(\frac{P(tw \mid \phi, M)}{|d| + 1} + \sum_{dw \in d} P(tw \mid dw, M) P(dw \mid d) \right) \end{aligned} \tag{7.6}$$

To find the optimal statistical translation model, we will still maximize the translation probability from documents to titles. Substituting the document-title translation probability $P(t|d,M)$ with equation (7.6), the optimization goal (Equation (7.1)) can be written as

$$M^* = \underset{M}{\operatorname{argmax}} \prod_{i=1}^{N} \left\{ \begin{array}{l} \sum_{dw \in d_i} P(dw \mid dw, M) P(dw \mid d_i) \times \\ \prod_{tw \in t_i} \left(\frac{P(tw \mid \phi, M)}{|d_i| + 1} + \sum_{dw \in d_i} P(tw \mid dw, M) P(dw \mid d_i) \right) \end{array} \right\} \tag{7.7}$$

Because the extra word slot in every title provides a chance for any word in the document to appear in the title through the self-translation process, it is not difficult to prove that this model will ensure that the self-translation probability *P*(*w*|*w*) will be no less than *P*(*w'*≠*w*|*w*) for any word *w*. The EM algorithm can again be applied to maximize Equation (7.7) and learn the word translation probabilities. The updating equations for the word translation probabilities are essentially the same as what are used for the standard learning algorithm for statistical translation models, except for the inclusion of the extra counts due to the null word slot.

7.2.2 Computing Document Query Similarity

In this section, we discuss how to apply the learned statistical translation model to find the title language model for a document and use the estimated title language model to compute the relevance value of a document with respect to a query. To accomplish this, we define the conditional probability *P*(*Q*|*D*) as the probability of using query *Q* as the title for document *D*, or, the probability of translating document *D* into query *Q* using the statistical title translation model, which is given below.

$$
\begin{aligned}
P(Q \mid D, M) &= \prod_{qw \in Q} \left\{ \frac{\varepsilon}{|d|+1} \left(P(qw \mid \phi, M) + \sum_{dw \in d} P(qw \mid dw, M) c(dw, D) \right) \right\} \\
&\approx \varepsilon \prod_{qw \in Q} \left\{ \frac{P(qw \mid \phi, M)}{|D|+1} + \sum_{dw \in D} P(qw \mid dw, M) P(dw \mid D) \right\}
\end{aligned} \qquad (7.8)
$$

As can be seen from Equation (7.8), the document language model *P*(*dw*|*D*) is not directly used to compute the probability of a query term. Instead, it is "converted" into a title language model through using word translation probabilities *P*(*qw*|*dw*). Such conversion also happens in the model proposed in (Berger & Lafferty, 1999), but there the translation model is meant to capture synonym and polysemy relations, and is trained with synthetic queries. Similar to the traditional language modeling approach, to deal with the query words that can't be generated from the title language model we need to do further smoothing, i.e.,

$$\begin{aligned} P(Q\,|\,D,M) &= \prod_{qw\in Q} \left\{ \begin{array}{l} \dfrac{\lambda\varepsilon}{|\,d\,|+1}\left(P(qw\,|\,\phi,M) + \sum\limits_{dw\in d} P(qw\,|\,dw,M)c(dw,D) \right) + \\ (1-\lambda)P(qw\,|\,GE) \end{array} \right\} \\ &\approx \varepsilon \prod_{qw\in Q} \left\{ \begin{array}{l} \lambda\left(\dfrac{P(qw\,|\,\phi,M)}{|\,D\,|+1} + \sum\limits_{dw\in D} P(qw\,|\,dw,M)P(dw\,|\,D) \right) + \\ (1-\lambda)P(qw\,|\,GE) \end{array} \right\} \end{aligned} \qquad (7.8')$$

where constant l is the smoothing constant and P(qw|GE) is the general English language model which can be easily estimated from the collection. In our experiment, we set the smoothing constant l to be 0.5 for all different models and all different collections.

Equation (7.8') is the general formula that can be used to score a document with respect to a query with any specific translation model. A different translation model would thus result in a different retrieval formula. In the next section, we will compare the retrieval performance using different statistical title translation models, including Model 1, Model 2 and Model 3.

7.3 Experiment

7.3.1 Experiment Design

The goal of our experiments is to answer the following three questions:

1. Will the title language model be effective for information retrieval? To answer this question, we will compare the performance of the title language model to that of the state-of-art information retrieval methods,

including the Okapi method and the traditional language model for information retrieval. Furthermore, since our method explores the correlation between words in titles and words in documents, it can be treated as a method of query expansion using title information instead of document information. Therefore, we also compare the performance of title language model to the performance of query expansion using titles of top retrieved documents.

2. How general is the trained statistical title translation model? Can a model estimated on one collection be applied to another? To answer this question, we conduct an experiment that applies the statistical title translation model learned from one collection to other collections. We then compare the performance of using a “foreign” translation model with that of using no translation model.

3. How important is the smoothing of self-translation in the title language model approach for information retrieval? To answer this question, we can compare the results for title language model 1 with model 2 and model 3.

We used three different TREC testing collections for evaluation: AP88 (Associated Press, 1988), WSJ90-92 (Wall Street Journal from 1990 to 1992) and SJM (San Jose Mercury News, 1991). Furthermore, because these three collections contain only news stories and therefore may be too homogeneous, we create a fourth heterogeneous collection, which consists of documents sampled from seven different document collections, namely AP88 (Associated Press, 1988), AP90 (Associated Press, 1990), WSJ90-92 (Wall Street Journal from 1990 to 1992), FR88 (Federal Register, 1998), ZIFF (Computer Select Disk, 1989 & 1990), SJM (San Jose Mercury News, 1991), ZIFF2 (Computer Select Disk, 1991 & 1992). For each collection, we sampled 10% of the documents out of the original collection and the total number of documents for this 'created collection' is 49765. Finally, all the relevant documents are appended to the synthesized collection.

We used TREC4 queries (201-250) and their relevance judgments for evaluation. The average length of the titles in these collections is four to five words. The different characteristics of the three databases allow us to check the robustness of our models.

7.3.2 Baseline Models

The two baseline methods are the Okapi method (Robertson et al., 1993) and the traditional language modeling approach. The exact formula for the Okapi method is shown in Equation (7.9)

$$Sim(Q,D) = \sum_{qw \in Q} \left\{ \frac{tf(qw,D)\log(\frac{N - df(qw) + 0.5}{df(qw) + 0.5})}{0.5 + 1.5\frac{|D|}{avg\_dl} + tf(qw,D)} \right\} \qquad (7.9)$$

where *tf*(*qw*,*D*) is the term frequency of word *qw* in document *D*, *df*(*qw*) is the document frequency for the word *qw*, and *avg_dl* is the average document length for all the documents in the collection. The query expansion for the Okapi method is implemented as follows: we select titles of the top 10 retrieved documents. The top 15 most popular title words among the selected titles are chosen as expanded query words with a discount weight 0.3. Parameters used in the query expansion are roughly tuned in order to get a reasonable performance. Notice that in this experiment, we use the title words of top retrieved documents for query expansion, instead of the document words. The reason for that is because the proposed retrieval model takes advantage of the correlation between title words and document words to enhance the retrieval performance. Therefore, as a fair comparison, the method of query expansion should use similar information.

The exact equation used for the traditional language modeling approach is shown in Equation (7.10).

$$P(Q\,|\,D) = \prod_{qw \in Q} ((1-\lambda)P(qw\,|\,GE) + \lambda P(dw\,|\,D)) \qquad (7.10)$$

The constant λ is the smoothing constant (similar to the λ in Equation (7.8')), and *P*(*qw*|*GE*) is the general English language model estimated from the collection. To make the comparison fair, the smoothing constant for the traditional language model is set to be 0.5, which is same as for the title language model.

7.3.3 Experiment Results for Homogeneous Datasets

The results on AP88, WSJ and SJM are shown in Table 7.1, Table7.2, and Table 7.3, respectively. In each table, we include the precisions at different recall points and precisions for different numbers of top retrieved documents. Meanwhile, the averaged precision and the precision after the number of relevant documents retrieved are also listed in those tables. For easy comparison, the results of average precision are also shown in Figure 7.1.

Several interesting observations can be made on these results:

First, let us compare the results between different title language models, namely model 1, model 2 and model 3. As seen from Table 7.1, 7.2 and 7.3, for all the three collections, model 1 is inferior to model 2, which is inferior to model 3, in terms of both average precision and precisions at different recall points. In particular, on the WSJ collection, title language model 1 performs extremely poorly compared with the other two methods. This result indicates that title language model 1 may fail to find relevant documents in some cases due to the problem of zero self-translation probability, as we discussed in Section 7.2. We computed the percentage of title words that cannot be found in their documents. This number is 25% for AP88 collection, 34% for SJM collection and 45% for WSJ collection. This high percentage of “missing” title words strongly suggests that the smoothing of self-translation probability will be critical. Indeed, for the WSJ collection, which has the highest percentage of missing title words, title language model 1, without any smoothing of self-translation probability, degrades the performance more dramatically than for collections AP88 and SJM, where more title words can be found in the documents, and the smoothing of self-translation probability is not as critical.

The second dimension of comparison is to compare title language models with traditional language model. As already pointed out by Berger and Lafferty (1999), the traditional language model can be viewed as a special case of the translation language model, i.e., all the translation probability $P(w'|w)$ become delta functions $\delta(w,w')$. Therefore, the comparison along this dimension can indicate if the translation probabilities learned from the correlation between titles and documents are effective in improving retrieval accuracy. As seen from Table 7.1, Table 7.2, and Table 7.3, title language model 3 performs better than the traditional language model over all the three collections in terms of all the performance measures. Thus, we can conclude that the translation probability learned from title-document pairs appears to be helpful for finding relevant documents.

Lastly, we compare the performance of the title language model approach with the Okapi method (Robertson et al., 1993). For all the three collections the title language model 3 outperforms Okapi in terms of most performance measures, except the precision at 0.0 recall point for AP88 collection, precisions at 0.0

and 0.1 recall point for WSJ collection, and the precision at the top 5 retrieved documents for SJM collection. For these four points, Model 3 performs worse than the Okapi method. Furthermore, we compare the performance of the proposed model to that of the query expansion. Surprisingly, for all three collections, the method of query expansion doesn’t improve the performance of original Okapi method. As already mentioned before, the query expansion used here is different from the traditional query expansion. For the traditional query expansion, the most popular *document words* of the top retrieved documents are added to the original query, while for this experiment, the most popular *title words* of the top retrieved documents are added to the original query. In effect, we conducted the same experiments but with query expansion on document words and the results are substantially better than the original Okapi method. The reason why the query expansion with title words doesn’t improve the performance can be explained by the fact that words used for titles are significantly different from words used for constructing documents. This difference is indicated by the fact that a large portion of words used in titles doesn’t appear in the corresponding documents. It is also confirmed by the comparison of a title language model to a document language model in Section 5.3.1.4 when we tried to explain why using documents as extra training data for creating title language model is inappropriate. Because of this difference, the title words of top retrieved documents may not necessary be important words to those relevant documents. Thus, expanding queries with title words doesn’t help improve the performance of information retrieval.

7.3.4 Experiments on the Generality of the Proposed Model

To test the generality of the estimated translation model, we applied the statistical title translation model leaned from the AP88 collection to the AP90 collection. We hypothesize that if two collections are ‘similar’, the statistical title translation model learned from one collection should be able to give a good approximation of the correlation between documents and titles of the other collection. Therefore, it would make sense to apply the translation model learned from one collection to another ‘similar’ collection.

Table 7.4 gives the results of applying the translation model learned from AP88 to AP90, together with the results of applying the translation model learned from AP90. Since title language model 3 already demonstrated its superiority to model 1 and model 2, we only considered model 3 in this experiment. From Table 7.4, we see that title generation model 3 trained on both AP88 and AP90 is able to outperform the traditional language model and Okapi method substantially in terms of most measures. Surprisingly, model 3 trained over the AP88 collection performs slightly better than the same model trained over the AP90 collection. This appears to contradict common sense, which usually believes that a model trained over the test collection should be superior to models trained over other collections. However, since the difference between these two models is small (e.g., the improvement is less than 5%), the improvement cannot be deemed as a reliable observation. We also applied the statistical title translation model learned from AP88 to WSJ to further examine the generality of the model and our learning method. This time, the performance of title language model 3 with the statistical title translation model learned from AP88 is only about the same as the traditional language model and the Okapi method for the collection WSJ. Since the statistical title translation model learned from AP88 can be expected to be a much better approximation of the correlation between documents and titles for AP90 than for WSJ, these results suggest that applying the translation model learned from a “foreign” database is helpful only when the “foreign” database is similar to the “native” one.

7.3.5 Experiment Results for the Heterogeneous Dataset

In this experiment, we test the proposed model 3 over the heterogeneous dataset that is formed by the documents sampled from seven different document collections. The results for the proposed model 3, the traditional language model, the Okapi method and the query expansion method are listed in Table 7.5. For easy comparison, the results of average precision are also shown in Figure 7.1. This time, the proposed model (i.e., 'Model 3',) doesn't improve the performance over the traditional language model. In contrast, the averaged precision of the proposed retrieval model degrades from 0.2842 to 0.2720. The reason why the title language model doesn't work for the heterogeneous datasets can be attributed to the diversity of title words. In order to see the situation clearly, let's consider the word 'bank', which can mean either a financial institute or the bank of a river. When the word 'bank' means a financial institute, we would expect words such as 'investment' to appear in titles. When it means the bank of a river, we may see words such as 'flood' appearing in titles. If we have mixture of financial stories and stories about weather, the title language model will be able to link the word 'bank' strongly with the title words 'investment' and 'flood'. Therefore, for a query with word 'bank', even if it is intended to find financial news stories, the title language query will push both the documents of financial news and the documents of floods toward the top of the retrieval list. Clearly, such an adjustment can actually degrade the performance. With this simple illustration, we can see that for the heterogeneous datasets, document words with multiple meanings can be linked to many more different title words than the homogeneous datasets. When a query contains such a document word, the title language model may push documents in several different topics toward the top of the retrieval list, even though the query only asks for documents in a single topic. As a result, the title language model can degrade the performance of information retrieval for the heterogeneous datasets. In order to see the diversity of title words that are correlated with document words, we compute the averaged number of unique title words that are linked with each document word by the title language model. It is 170.9 for the synthesized heterogeneous dataset and only around 130 for both AP90 and AP88 collection. This number indicates that for the heterogeneous dataset, each document word is linked with many more unique title words than the homogeneous datasets, which provides more opportunities for the title language model to move irrelevant documents toward the top of the retrieval list. Furthermore, for each document word, we compute the ten most correlated title words for both the synthesized dataset and the AP88 dataset, and found that in average less than one word is shared by both collections. This fact again indicates that the set of title words correlated with each document word in the synthesized dataset is much more diverse than the homogeneous datasets. In conclusion, the above analysis suggests that heterogeneous datasets can lead each document to correlate with diverse title words and therefore the proposed title language model for information retrieval may not be effective for heterogeneous datasets.

Furthermore, different from the experiments with the homogeneous datasets, for this dataset the query expansion method does improve the performance of the Okapi method substantially in terms of most measures. It can be explained by the fact that for this synthesized dataset, the average retrieval precision for the top 10 documents is 0.64 while for the homogeneous datasets, the averaged retrieval precision for the top 10 documents is no more than 0.35. Therefore, the top retrieved documents for the heterogeneous dataset are much more relevant to the query than those homogeneous datasets, which provides a better chance for the query expansion method to improve the performance even using the title words of the top retrieved documents. To further confirm this hypothesis, we run the query expansion method over the synthesized dataset again using top 100 documents (the retrieval accuracy at 100 documents is around 0.3460). All the other parameters, including the number of expanded query words and the weights used for expanded query words, are kept the same. The resulted averaged precision is 0.2812, which is only as good as the Okapi method (e.g., 0.2782).

7.4 Conclusions For the Title Language Model of IR

Bridging the "gap" between a query language model and document language model is an important issue when applying language models to information retrieval. In this thesis, we propose bridging this gap by exploiting titles to estimate a title language model, which can be regarded as an approximate query language model. The essence of our work is to approximate the query language model for a document with the title language model for the document. Operationally, we first estimate such a translation model by using all the document-title pairs in a collection. The translation model can then be used to "convert" a regular document language model into a title language model. Finally, the title language model estimated for each document is used to compute the query likelihood. Intuitively, the scoring is based on the likelihood that the query could have been a title for a document.

From the experiment results, we can draw the following conclusions:

- Based on the comparison between the title language model, the traditional language model and the Okapi method, we can conclude that the title language model for information retrieval is an effective retrieval method for homogeneous datasets but may not be effective for heterogeneous datasets due to the fact that the correlation between titles and documents varies from one collection to another.

- Based on the comparison between three different title language models for information retrieval, we can conclude that title generation model 2 and 3 are superior to model 1, and model 3 is superior to model 2. Since the difference between the three different title language models is on how they handle the self-translation probability, we can conclude that: First it is crucial to smooth the self-translation probability to avoid the zero self-translation probability. Secondly, a better smoothing method for self-translation probability can improve the performance. Results show that adding an extra null word slot to the title is a reasonable smoothing method for the self-translation probabilities.

- The success of applying the title language model learned from AP88 to AP90 appears to indicate that in the case when the two collections are similar, the correlation between documents and titles in one collection also tend to be similar to that in the other. Therefore, it would seem to be appropriate to apply the statistical title translation model learned from one collection to the retrieval task of another similar collection.

Table 7.1: Results for AP88 collection. 'LM' stands for the traditional language model, 'Okapi' stands for the Okapi formula, 'QE' stands for the query expansion based titles of the top retrieved documents, and model-1, model-2 and model-3 stand for the title language model 1, model 2 and model 3.

Collection	**LM**	**Okapi**	**Model 1**	**Model 2**	**Model 3**	**QE**
Recall 0.0	0.5986	0.6459	0.5708	0.6028	0.6224	0.6429
Recall 0.1	0.4398	0.4798	0.2061	0.4885	0.5062	0.4717
Recall 0.2	0.3490	0.3789	0.1409	0.4082	0.4024	0.3749
Recall 0.3	0.3035	0.3286	0.1154	0.3417	0.3572	0.3280
Recall 0.4	0.2492	0.2889	0.0680	0.2830	0.3133	0.2904
Recall 0.5	0.2114	0.2352	0.0525	0.2399	0.2668	0.2351
Recall 0.6	0.1689	0.2011	0.0277	0.1856	0.2107	0.2034
Recall 0.7	0.1369	0.1596	0.0174	0.1460	0.1742	0.1548
Recall 0.8	0.0811	0.0833	0.0174	0.0897	0.1184	0.0889
Recall 0.9	0.0617	0.0611	0.0115	0.0651	0.0738	0.0615
Recall 1.0	0.0580	0.0582	0.0115	0.0618	0.0639	0.0584
Avg. Prec.	**0.2238**	**0.2463**	**0.2108**	**0.2516**	**0.2677**	**0.2454**
5 docs	0.3600	0.3720	0.3240	0.376	0.4080	0.3640
10 docs	0.2820	0.3080	0.2560	0.312	0.3380	0.3080
15 docs	0.2520	0.2787	0.2330	0.268	0.3000	0.2707
20 docs	0.2290	0.2480	0.2100	0.236	0.2700	0.2450
30 docs	0.1930	0.2080	0.1787	0.2067	0.2270	0.2067
100 docs	0.0974	0.1094	0.0940	0.1086	0.1200	0.1804
200 docs	0.0487	0.0547	0.0470	0.0543	0.0600	0.0542
500 docs	0.0195	0.0219	0.0188	0.0217	0.0240	0.0217
1000 docs	0.0097	0.0109	0.0094	0.0109	0.0120	0.0108
R-precision	**0.2511**	**0.2668**	**0.2422**	**0.2994**	**0.2937**	**0.2676**

Table 7.2: Results for WSJ collection. 'LM' stands for the traditional language model, 'Okapi' stands for the Okapi formula, 'QE' stands for the query expansion based titles of the top retrieved documents, and model-1, model-2 and model-3 stand for the title language model 1, model 2 and model 3.

Collection	**LM**	**Okapi**	**Model 1**	**Model 2**	**Model 3**	**QE**
Recall 0.0	0.5127	0.5564	0.2710	0.4895	0.5136	0.5619
Recall 0.1	0.4308	0.4539	0.2061	0.4055	0.4271	0.4499
Recall 0.2	0.3587	0.3546	0.1409	0.3449	0.3681	0.3543
Recall 0.3	0.2721	0.2724	0.1154	0.2674	0.2878	0.2698
Recall 0.4	0.2272	0.1817	0.0680	0.2305	0.2432	0.1804
Recall 0.5	0.1812	0.1265	0.0525	0.1723	0.1874	0.1277
Recall 0.6	0.1133	0.0840	0.0277	0.1172	0.1369	0.0855
Recall 0.7	0.0525	0.0308	0.0174	0.0764	0.0652	0.0312
Recall 0.8	0.0328	0.0218	0.0174	0.0528	0.0465	0.0223
Recall 0.9	0.0153	0.0106	0.0115	0.0350	0.0204	0.0111
Recall 1.0	0.0153	0.0106	0.0115	0.0321	0.0204	0.0111
Avg. Prec.	**0.1844**	**0.1719**	**0.0761**	**0.1851**	**0.1950**	**0.1722**
5 docs	0.3200	0.3200	0.1440	0.2960	0.3160	0.3120
10 docs	0.2580	0.2660	0.1140	0.2560	0.2600	0.264
15 docs	0.2253	0.2320	0.1053	0.2200	0.2290	0.2307
20 docs	0.2060	0.2170	0.1020	0.1970	0.2130	0.2170
30 docs	0.1747	0.1813	0.0860	0.1660	0.1813	0.1820
100 docs	0.1036	0.1018	0.0514	0.1010	0.1050	0.1020
200 docs	0.0518	0.0509	0.0257	0.0505	0.0525	0.0510
500 docs	0.0207	0.0204	0.0103	0.0202	0.0210	0.0204
1000 docs	0.0104	0.0102	0.0051	0.1010	0.0105	0.0102

R-precision	**0.2137**	**0.2154**	**0.0962**	**0.2188**	**0.231**	**0.2142**

Table 7.3: Results for SJM collection. 'LM' stands for the traditional language model, 'Okapi' stands for the Okapi formula, 'QE' stands for the query expansion based titles of the top retrieved documents, and model-1, model-2 and model-3 stand for the title language model 1, model 2 and model 3.

Collection	**LM**	**Okapi**	**Model 1**	**Model 2**	**Model 3**	**QE**
Recall 0.0	0.5093	0.5352	0.5470	0.5126	0.5480	0.5619
Recall 0.1	0.4009	0.4054	0.4226	0.4249	0.4339	0.4499
Recall 0.2	0.3345	0.3232	0.3281	0.3650	0.3638	0.3543
Recall 0.3	0.2813	0.2348	0.2712	0.2890	0.3019	0.2698
Recall 0.4	0.2076	0.1692	0.1991	0.2236	0.2296	0.1804
Recall 0.5	0.1815	0.1378	0.1670	0.1874	0.1919	0.1277
Recall 0.6	0.1046	0.0986	0.1095	0.1393	0.1431	0.0855
Recall 0.7	0.0816	0.0571	0.0782	0.0862	0.0974	0.0312
Recall 0.8	0.0460	0.0312	0.0688	0.0591	0.0788	0.0223
Recall 0.9	0.0375	0.0312	0.0524	0.0386	0.0456	0.0111
Recall 1.0	0.0375	0.0312	0.0524	0.0386	0.0456	0.0111
Avg. Prec.	**0.1845**	**0.1727**	**0.1910**	**0.1983**	**0.2081**	**0.1722**
5 docs	0.2880	0.3120	0.2720	0.2640	0.3040	0.3120
10 docs	0.2460	0.2460	0.2100	0.2580	0.2520	0.264
15 docs	0.2147	0.2360	0.1880	0.2280	0.2227	0.2307
20 docs	0.1990	0.2140	0.1740	0.2110	0.2220	0.2170
30 docs	0.1760	0.1773	0.1487	0.1830	0.1913	0.1820
100 docs	0.0920	0.0984	0.0850	0.0996	0.1018	0.1020
200 docs	0.0496	0.0492	0.0425	0.0498	0.0509	0.0510
500 docs	0.0198	0.0197	0.0170	0.0199	0.0204	0.0204
1000 docs	0.0099	0.0098	0.0085	0.0100	0.0102	0.0102
R-precision	**0.2250**	**0.2030**	**0.224**	**0.2203**	**0.2243**	**0.2142**

Table 7.4: Results for AP90. 'LM' stands for the traditional language model, 'Okapi' stands for the Okapi formula, 'QE' stands for the query expansion based on the titles of the top retrieved documents, model3 (AP88) stands for the title language model 3 using AP88 data for training, and model3 (AP90) stands for the title language model3 using AP90 data for training. Different from the previous experiments in which the translation model is learned from the retrieved collection itself, this experiment applies the translation model learned from AP88 to retrieve relevant document in AP90 collection. For the method of query expansion, we also use the titles of top retrieved documents from collection AP88.

Collection	**LM**	**Okapi**	**Model 3 (AP88)**	**Model3 (AP90)**	**QE**
Recall 0.0	0.5904	0.5888	0.6598	0.6471	0.5480
Recall 0.1	0.4775	0.4951	0.5137	0.5034	0.5075
Recall 0.2	0.4118	0.4308	0.4454	0.4230	0.4225
Recall 0.3	0.3124	0.3374	0.3628	0.3500	0.3583
Recall 0.4	0.2700	0.2894	0.3248	0.3061	0.3306
Recall 0.5	0.2280	0.2567	0.2665	0.2643	0.2854
Recall 0.6	0.1733	0.2123	0.2222	0.2027	0.2153
Recall 0.7	0.1294	0.1230	0.1372	0.1409	0.1365
Recall 0.8	0.0991	0.0969	0.1136	0.1252	0.1082
Recall 0.9	0.0782	0.0659	0.0963	0.0959	0.0657
Recall 1.0	0.0614	0.0550	0.0733	0.0744	0.0359
Avg. Prec.	**0.2411**	**0.2511**	**0.2771**	**0.2675**	**0.2567**
5 docs	0.3640	0.364	0.3880	0.3880	0.3480
10 docs	0.3020	0.3060	0.3280	0.2940	0.3180
15 docs	0.2693	0.2787	0.2960	0.2653	0.2920
20 docs	0.245	0.2590	0.2780	0.2480	0.2750
30 docs	0.2027	0.2193	0.2387	0.2172	0.2413
100 docs	0.1142	0.1198	0.1306	0.1196	0.1304
200 docs	0.0571	0.0599	0.0653	0.0598	0.0652

500 docs	0.0228	0.0240	0.0261	0.0239	0.0261
1000 docs	0.0131	0.0120	0.0131	0.0120	0.0130
R-precision	**0.2746**	**0.2920**	**0.3117**	**0.2826**	**0.2925**

Table 7.5: Results for the synthesized heterogeneous dataset. The synthesized heterogeneous dataset consists of documents sampled from seven different document collections. ‘LM’ stands for the traditional language model, ‘Okapi’ stands for the Okapi formula, ‘QE’ stands for the query expansion based on the titles of the top retrieved documents, ‘Model3’ stands for the title language model 3.

Collection	**LM**	**Okapi**	**Model 3**	**QE**
Recall 0.0	0.8281	0.8690	0.8126	0.8592
Recall 0.1	0.6701	0.6645	0.6529	0.6933
Recall 0.2	0.5402	0.5449	0.5371	0.5942
Recall 0.3	0.4411	0.4300	0.4150	0.4465
Recall 0.4	0.3135	0.3135	0.3235	0.3223
Recall 0.5	0.2677	0.2339	0.2367	0.2649
Recall 0.6	0.1440	0.1055	0.1150	0.1398
Recall 0.7	0.0880	0.0839	0.0740	0.0865
Recall 0.8	0.0565	0.0470	0.0502	0.0519
Recall 0.9	0.0189	0.0249	0.0207	0.0253
Recall 1.0	0.0138	0.0092	0.0127	0.0114
Avg. Prec.	**0.2842**	**0.2782**	**0.2720**	**0.2931**
5 docs	0.6800	0.6960	0.6760	0.7120
10 docs	0.6160	0.6420	0.6140	0.6620
15 docs	0.5760	0.5947	0.5733	0.6160
20 docs	0.5480	0.5580	0.5470	0.5730
30 docs	0.5033	0.5000	0.4920	0.5200
100 docs	0.3428	0.3460	0.3354	0.3612
200 docs	0.1714	0.1730	0.1677	0.1806
500 docs	0.0686	0.0692	0.0671	0.0722
1000 docs	0.0343	0.0346	0.0335	0.0361

R-precision	**0.3417**	**0.3322**	**0.3332**	**0.3467**

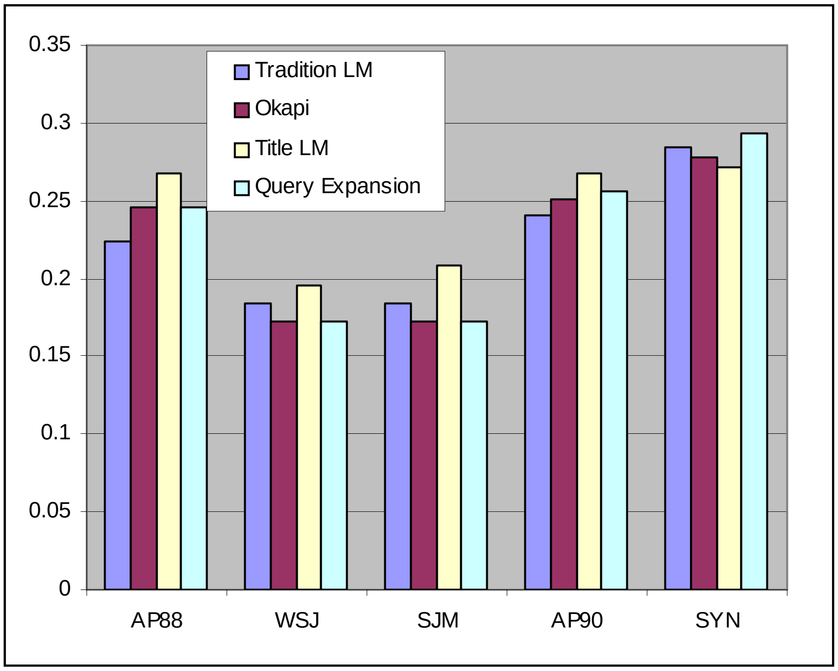

Figure 7.1: Average precision over 11 recall points for the tradition language model (Tradition LM), the Okapi method (Okapi), the title language model (Title LM), and the query expansion method (Query Expansion). Results for five different datasets are listed. They are: AP88, WSJ, SJM, AP90 and the synthesized dataset.

Chapter 8

CONCLUSIONS

In this thesis, we proposed a new statistical framework for automatic title generation named the ‘direct model with dual noisy channels’. Unlike the previous framework where there is only a single noisy channel that transforms documents into titles, in the new framework a hidden ‘information source’ is assumed and a title and a document are created from the same hidden ‘information source’ but through different noisy channels. In order to create a title for a document, we need to first recover the hidden ‘information source’ from the observed document and then create a title from the estimated ‘information source’. Two noisy channels are suggested for this process: the first noisy channel distills the ‘information source’ out of the document and the second noisy channel will then create a title out of the distilled ‘information source’.

Various methods have been examined for each component of the proposed model, namely the component for distilling important words out of documents, the component for computing the correlation between title words and document words, and the component for ordering selected title words. The best configuration uses the TF.IDF value (or mutual information) for distilling important content words, a statistical translation model for computing document-word-title-word correlation, an expanded title language model for estimating the likelihood of word sequences, and an order expansion method for estimating the likelihood for a set of words. Compared to both the Naïve Bayes method with a limited vocabulary and the best implementation of the framework suggested by Witbrock and Mittal, the proposed model has demonstrated substantially better performance over two large datasets in terms of the F1 metric and human judgments. Experiments with speech recognition transcripts and machine-translated documents also indicated that the proposed model is more resilient to word errors than the old framework in terms of the F1 metric. Furthermore, experiments with heterogeneous testing documents suggested that the proposed model could be effective even when testing documents are substantially different from training documents.

In addition to examining automatic title generation, in this thesis we also extend the title generation model to information retrieval and automatic text categorization. For information retrieval, we treat queries as variants of titles, and then the similarity of a query to a document can be estimated as the likelihood of using the query as a title for this document. For automatic text categorization, we treat each text category label as a different title word, and the problem of finding appropriate category labels for a document can be cast as the problem of finding appropriate title words for the document. Empirical studies on information retrieval have shown that the title language model for information retrieval is able to achieve better performance than the traditional language model for homogeneous collections, but not for heterogeneous collections. Preliminary studies with text classification have suggested that the title language model for text categorization has potential to achieve performance comparable to other commonly used text categorization methods, particularly for common categories.

Other technical contributions of this thesis include:

- **Investigation of evaluation metrics**. In this thesis, we investigated three metrics that can be used for evaluating the quality of machine-generated titles:

two automatic ones and a manual one. They are: the F1 metric, the edit distance metric and human judgments. According to the experiments, both the F1 metric and the edit distance metric are positively correlated with human judgments, but the F1 metric correlates with the human judgments more strongly than the edit distance metric and is more sensitive to changes in the quality of titles.

- **Investigation of different title word selection methods.** In this thesis we examined various approaches for title word selection, including two variations of Naïve Bayes approaches, a text categorization approach, a statistical machine translation approach, a reverse information retrieval approach, a K nearest neighbor approach and a TF.IDF approach. The general conclusion is that methods that are able to take into account all the words in the test document appear to be more effective than methods that only examine a subset of words in the test document.

- **Different title word ordering methods.** Unlike the old framework for automatic title generation, where likelihood P(T) is used to order selected title words, a new metric $P(T)/P(\{tw \in T\})$ is introduced for title word ordering. The advantage of this new metric is that it avoids the problem of P(T) favoring word sequences with common words. Empirical studies have shown that with the help of this new metric for title word ordering, we are able to substantially improve the F1 scores.

- **Automatically determining title length**. Instead of relying on the distribution of title lengths in the training data to find titles with appropriate lengths, in this thesis we introduce the ‘end_of_sentence’ symbol to help find the appropriate title length. The basic idea is to treat the ‘end_of_sentence’ symbol as a normal title word candidate. The title with highest score will be selected but only the word sequence before the ‘end_of_sentence’ symbol is used as the final title for the document. By generating titles in this way, the resulting title will have a ‘smooth’ ending.

Of course, there are still some open issues with the proposed model in this thesis. The biggest deficiency in the current implementation of the ‘direct model with dual noisy channels’ is with the realization of the second noisy channel, namely the noisy channel that distills the ‘information source’ out of the original documents. Due to the lack of training data with documents and descriptions of corresponding ‘information sources’, we cannot learn an appropriate model for such a noisy channel. Instead, we simply resorted to some well-known functions, such as the TF.IDF value (or mutual information). More studies are needed in the future to understand how to learn such a model, particularly in an unsupervised manner. Secondly, since this thesis mainly focuses on the statistical model for automatic title generation, we only take advantage of the statistics of title words, document words and their co-occurrence patterns. The structure of documents and titles are not considered in this thesis. As a next step, we can take advantage of the structure of documents and titles, both semantic and syntactic, in order to further improve the quality of machine-generated titles. One way to include semantic knowledge into the statistical model is to enrich the ‘word translation’ probability P(tw|dw). In this dissertation, P(tw|dw) is estimated based only on the appearance of title word ‘tw’ and document word ‘dw’. By including the semantic knowledge of words, we can expand P(tw|dw) as P(s(tw,dw), tw|dw), where s(tw,dw) is a vector or a number representing our knowledge about the relationship between words ‘tw’ and ‘dw’. As an example, with WordNet we can use s(tw,dw) to represent the length of the minimum path between word ‘tw’ and ‘dw’ within the tree of ontology. In this case, s(tw,dw) measures how closely these two words are related to each other. Thirdly, in this thesis we completely ignore the stopwords in generated titles. Removing stopwords from documents is a standard practice in information retrieval and has been proved to be effective. The reason for excluding stopwords from generated titles is because stopwords are not supposed to reflect the content of the test document and therefore title word selection will be ineffective in choosing appropriate stopwords. Instead of choosing stopwords based on the document content, appropriate stopwords should be chosen to connect the content words that are used for titles. In practice, because stopwords appear much more frequently than other title words, the resulting 'word translation' probability P(tw|dw) for most document words 'dw' will be nonzero for stopwords. Moreover, because the prediction of title words is based on the sum of P(tw|dw), the title word selection procedure will always favor stopwords as title words than other words. As a result, if we allow stopwords to be included in machine-generated titles, it is very likely that many machine-generated titles will contain multiple stopwords and be less informative to the content of the test document. Based on the above analysis, in order to include stopwords in generated title, we need an extra noisy channel. This extra noisy channel will examine the already selected title words and search for the appropriate stopwords that are able to connect those selected title words together smoothly. Finally, the proposed statistical model for title generation is computationally expensive, not only on the training phase but also on the testing phase. For the corpus used in the experiment, which has 40,000 training documents and 10,000 testing documents, we needed 3 days to train the statistical model and 3.5 days to generate 10,000 documents. This is unacceptable if we want to apply the proposed model to real applications. As future work, we need to improve the efficiency of the proposed model significantly.

Compared to automatic text summarization and automatic text categorization, the key advantage of automatic title generation is the flexibility in creating titles. Unlike automatic text categorization, which can only assign a document to one of the predefined category labels, automatic title generation is able to create a flexible title for the unseen document which may not be any title in the training corpus. Different from automatic text summarization, where summaries are formed by sentences extracted from the original document, automatic title generation can create titles consisting of words that don't belong to the original document. It is this flexibility that makes automatic title generation useful and unique to many tasks that both automatic text summarization and automatic text categorization cannot be applied to.

One useful application for automatic title generation is to summarize multiple documents, which may not belong to any of the predefined categories and cannot be summarized by one or two sentences extracted from original documents. The idea of being able to summarize multiple documents is useful to many applications. For example, for the problem of document clustering, it will be useful to automatically create a headline for each cluster of documents so that users can quickly find the cluster that they want without having to read any of the documents inside the cluster. Another interesting application is to apply automatic title generation model to summarize foreign documents into English headlines as first suggested in the paper (Witbrock and Mittal, 1999). Most text summarization approaches are based on extractions from the original documents and therefore will fail to create English summaries for foreign documents, while for statistical title generation models, words that can be used for titles are not limited to the words in the test document. The idea of cross-lingual title generation can be useful to cross-lingual information retrieval. In this case, not only will foreign documents be retrieved but also an English title provided for each retrieved document. With the help of English titles, users may be able to quickly find interesting foreign documents without having to translate them into English.

More interestingly, the idea of automatic title generation, particularly the idea of dual noisy channels, can be extended to many other problems. In this thesis we have demonstrated the generality of the model by applying it to information retrieval and automatic text categorization. The title generation model can also be extended to the problem of automatically annotating images, where a system is asked to find a set of words that appropriately describe the content of images. In the previous research (Duygulu et al., 2002), this task has been treated as a translation problem with images and texts being viewed as information written in different languages, and the process of annotating images becomes a process of 'translating' images into text. However, similar to the problem described in automatic title generation, the translation view of image annotation essentially consists of a single noisy channel transforming images into text, which may lead to the problem of transcribing everything appearing in an image into text. In many cases, not everything in an image needs to be transcribed into text. For example, consider an image that contains a beautiful mountain scene. Even though there is a portion of blue sky in the back of the image, we usually don't use the word 'sky' in the description of the image since it is mainly "about" the mountain scene. Continuing in the direction of dual noisy channels, we can introduce an extra noisy channel that distills useful image information from the original image, and then the description text will be created from the 'distilled' images instead of the original images. As a further extension, we can apply the same idea to create titles for image collections and video sequences. For the case of creating titles for videos, it becomes even more interesting because both text (obtained from speech recognized transcripts) and images can be used as hints for finding appropriate title sequences, and how to combine these two types of evidence together is an interesting research topic. In the extreme case, we may be able to create a reasonable name for movies.

The other application of the title generation model is to apply it to automatic text categorization. By viewing each category label as a title word, the problem of finding correct category labels for documents becomes creating appropriate titles for a given document. The preliminary experiments with the title generation model for text categorization have shown that for common categories, the title generation model is able to perform as well as the state-of-art models in text categorization, such as the K nearest neighbor approach or the Naïve Bayes approach.

APPENDIX

In this appendix, we include the five documents that have been used to show the examples of machine-generated titles.

Document 1:

ILLEGAL IMMIGRATION HAS INCREASED IN THE LAST FOUR YEARS AND SO HAS THE BUDGET OF THE IMMIGRATION AND NATURALIZATION SERVICE .CONGRESS HAS PUT THE MONEY INTO STOPPING ILLEGAL ALIENS AT THE BORDER .BORDER SECURITY FUNDS HAVE MORE THAN DOUBLED .PREVENTING THOSE WHO SNEAK IN FROM WORKING HAS BEEN LESS OF A PRIORITY .THE I. N. S. SLOWLY HAS BEEN ADDRESSING THE ISSUE ANYWAY .TO LIMIT THE USE OF PHONY WORK PAPERS BY ILLEGAL IMMIGRANTS THE AGENCY IS RELEASING A NEW I. D. CARD THAT'S HARDER TO COUNTERFEIT .NPR'S PETER KENYON REPORTS .FIRST OF ALL THE I. N. S. IS ANXIOUS THAT PEOPLE UNDERSTAND WHAT THIS NEW CARD IS NOT .IT'S NOT A NATIONAL I. D. CARD .IT'S NOT EVEN A NEW GREEN CARD .THOSE ARE FOR PERMANENT NONRESIDENT WORKERS .LAST YEAR NEARLY ONE POINT SEVEN MILLION GREEN CARDS WERE ISSUED .ANOTHER SEVEN HUNDRED THOUSAND CARDS WERE ISSUED TO PEOPLE IN THIS COUNTRY ON TEMPORARY WORK VISAS OR TO THOSE WORKING HERE WHILE THEY HAVE AN ASYLUM PETITION PENDING WITH THE GOVERNMENT .THOSE ARE THE CARDS THAT ARE CHANGING .THE I. N. S. IS USING ADVANCE TECHNOLOGY TO COMBAT DOCUMENT FRAUD A CHRONIC PROBLEM FACING THOSE WHO WANT TO REDUCE THE JOBS MAGNET DRAWING PEOPLE ACROSS THE BORDER .IN GOVERNMENT JARGON THE CARD IS CALLED AN EAD OR EMPLOYMENT AUTHORIZATION DOCUMENT .COMMISSIONER DORIS MEISSNER CALLS IT A LANDMARK IN DOCUMENT SECURITY .THE NEW EAD CARD IS STATE OF THE ART IT IS AN IMPORTANT TECHNOLOGICAL ADVANCE FOR THE IMMIGRATION SERVICE .INSTEAD OF THE OLD LAMINATED SOFT CARDS THE NEW ONES ARE MADE OF HARD PLASTIC LIKE CREDIT CARDS .BUT UNLIKE A CREDIT CARD THIS ONE

INCLUDES A FINGERPRINT A PHOTOGRAPH AT LEAST THREE HOLOGRAMS MICROPRINTING AND OTHER SECURITY FEATURES .I. N. S. SAYS THEIR FORENSIC EXPERTS CALL THIS CARD THE MOST COUNTERFEIT RESISTANT DOCUMENT OF ITS KIND .MOST TEMPORARY WORK AUTHORIZATIONS EXPIRE AFTER SIX MONTHS OR A YEAR SO MEISSNER SAYS THE NEW CARDS WILL QUICKLY GET INTO CIRCULATION .AND SO IN THE NEXT PERIOD WE WILL BE SYSTEMATICALLY REPLACING EAD CARDS THAT ARE OUT THERE WITH ONE THAT WILL BE MUCH EASIER FOR EMPLOYERS TO WORK WITH MUCH SAFER AND MORE SECURE FOR THOSE PEOPLE WHO ACTUALLY HAVE AND SHOULD HAVE THE EAD CARD AND MUCH MORE DIFFICULT TO REPLICATE .TEXAS REPUBLICAN LAMAR SMITH CHAIRMAN OF THE HOUSE IMMIGRATION SUBCOMMITTEE APPLAUDED THE NEW CARD .IT'S A GOOD STEP FORWARD BECAUSE THERE HAS BEEN SUCH ABUSE THERE IS SUCH A PROLIFERATION OF FRAUDULENT DOCUMENTS THAT THIS EFFORT TO TRY TO REDUCE THE NUMBER OF FRAUDULENT DOCUMENTS IS A GOOD ONE .WELL OF COURSE IT'S LEGITIMATE AND APPROPRIATE FOR THE GOVERNMENT TO MAKE A DOCUMENT MORE TAMPER RESISTANT .LUCAS GUTTENTAG WITH THE A. C. L. U.'S IMMIGRANT RIGHTS PROJECT SAYS HIS CONCERN IS THAT TOO OFTEN THE I. N. S. REACHES FOR A TECHNOLOGICAL FIX RATHER THAN ADDRESS THE LONGSTANDING IN HOUSE PROBLEMS THAT PLAGUE LEGITIMATE NONRESIDENT WORKERS .TOO OFTEN INDIVIDUALS WHO ARE ENTITLED TO GET WORK AUTHORIZATION CARDS ARE REJECTED BY THE I. N. S. BECAUSE OF AGENCY ERRORS MISTAKES AND INEFFICIENCIES IN THE I. N. S. COMPUTER AND RECORDS .AFTER PUTTING THESE NEW CARDS INTO CIRCULATION THE I. N. S. PLANS TO APPLY THE SAME TECHNOLOGY TO GREEN CARDS .SO FAR THERE'S NO TIMETABLE FOR THAT PROJECT .MEANWHILE THE AGENCY IS MOVING AT HIGH SPEED ON THE ENFORCEMENT FRONT .COMMISSIONER MEISSNER YESTERDAY ANNOUNCED THE DEPLOYMENT OF MORE THAN SEVEN HUNDRED NEW BORDER PATROL AGENTS PRIMARILY IN TEXAS ARIZONA NEW MEXICO AND CALIFORNIA .AN ADDITIONAL THREE HUNDRED AGENTS ARE BEING HELD IN RESERVE .MEISSNER SAYS THIS WINTER WILL TELL IF THIS LATEST INCREASE IN ENFORCEMENT IS ENOUGH OR IF THE BATTLE TO SECURE THE BORDERS MUST BE ESCALATED YET AGAIN .I'M PETER KENYON IN WASHINGTON .

Document 2:

MUSIC MAKING THAT BEGAN IN ARGENTINA AND LATVIA IS TOPPING THE C. D. CHARTS IN GERMANY AND JAPAN. THE ARGENTINIAN IS ASTOR PIAZZOLLA THE TANGO MASTER WHO DIED IN NINETEEN NINETY TWO. LATVIAN VIOLINIST GIDON KREMER HAS RECORDED SOME OF PIAZZOLLA'S COMPOSITIONS A REAL DEPARTURE FOR THIS CLASSICAL VIRTUOSO. BUT KREMER NOW FIFTY IS KNOWN AS AN ICONOCLAST. SO WAS PIAZZOLLA. NPR'S SUSAN STAMBERG REPORTS. THIS IS ASTOR PIAZZOLLA RECORDED IN NINETEEN EIGHTY SIX PLAYING HIS BANDONIAN A SOUTH AMERICAN VERSION OF THE ACCORDION . AND THIS IS GIDON KREMER PLAYING TODAY THE SAME PIAZZOLLA COMPOSITION CONCERTO FOR A QUINTET . THERE'S A BANDONIAN HERE TOO . BUT KREMER'S VIOLIN IS THE LEAD INSTRUMENT. WHEN YOU WERE A LITTLE BOY IN RIGA STUDYING VIOLIN WITH YOUR FATHER AND YOUR GRANDFATHER MR. KREMER DID YOU EVER IMAGINE THAT ONE DAY YOU WOULD BE PLAYING THIS MUSIC. NO I NEVER NEVER IMAGINED THAT I WOULD EVER PLAY PIAZZOLLA'S MUSIC. HOWEVER I LEARNED THROUGH MY LIFE TO APPRECIATE GREAT COMPOSERS. AND PIAZZOLLA IS ONE OF THEM. GIDON KREMER WAS JUST FOUR WHEN HE STARTED TO PLAY VIOLIN TCHAIKOVSKY PROKOFIEV THE RUSSIAN MASTERS FOUR TO FIVE HOURS A DAY OF PRACTICE. HIS FATHER INSISTED NO PLAYGROUNDS NO STICK BALL. IN RIGA LATVIA IN THE KREMER FAMILY THE VIOLIN ALWAYS CAME FIRST THE CLASSICS. ASTOR PIAZZOLLA WHO INVENTED THE NEW TANGO NUEVO TANGO ALSO GREW UP WITH THE CLASSICS IN ARGENTINA. IN NINETEEN EIGHTY EIGHT HE TOLD PUBLIC RADIO'S TERI GROSS ABOUT HIS EARLIEST MUSICAL INFLUENCES. I WAS BORN PLAYING JOHANN SEBASTIAN BACH ON THE BANDONIAN BECAUSE IT SOUNDS MORE LIKE AN ORGAN AS THE INSTRUMENT. MY TEACHER TAUGHT ME TO PLAY ALL CLASSICAL MUSIC ON THE BANDONIAN. I NEVER PLAYED A TANGO WHEN I WAS SMALL. IN HIS THIRTIES IN THE NINETEEN FIFTIES PIAZZOLLA INTRODUCED HIS NEW TANGO A MIX OF TRADITIONAL TANGO CLASSICAL MUSIC JAZZ AND POP. PURISTS WERE HORRIFIED. HE'D TAKEN THEIR MOST BELOVED ART FORM THEIR FAVORITE DANCE MUSIC AND DARED TO MAKE IT UNDANCEABLE. THEY SAID PIAZZOLLA WAS THE ASSASSIN OF THE TANGO.

THEY USED TO CALL ME ON THE PHONE AND SAY THEY WOULD WAIT FOR ME IN THE STREETS . AND THEY WOULD BE BEATEN . WHY JUST BECAUSE I CHANGED THE TANGO . BUT I DIDN'T CHANGE THE TANGO . WHAT I CHANGED IS THE WAY OF PLAYING AND THE WAY OF WRITING . I MEAN I I GAVE IT A A NEW DIMENSION . AND NOW GIDON KREMER HAS TAKEN UP PIAZZOLLA'S REVOLUTION WITH HIS VIOLIN . THE FORM IS NEW FOR KREMER . THE SPIRIT OF ADVENTURE IS NOT . THE LATVIAN VIRTUOSO HAS ALWAYS BEEN AN EXPLORER SEARCHING FOR NEW MUSICAL CHALLENGES . ABOUT FOURTEEN YEARS AGO KREMER WENT TO A PIAZZOLLA CONCERT IN PARIS . HE BECAME A FAN . BUT GIDON KREMER WHO HAS BEEN CALLED AN INTELLECTUAL MUSICIAN AND BY HERBERT FONTARIAN THE GREATEST VIOLINIST IN THE WORLD WASN'T SURE HE COULD MASTER THE NUEVO TANGO . TO PLAY PIAZZOLLA'S MUSIC YOU NOT ONLY NEED TO FEEL HIS IDIOM TO BE SOMEHOW TURNED ON ON TANGOS . BUT IT SEEMS LIKE YOU HAVE TO MOVE FREELY LIKE BEING A JAZZ MUSICIAN BETWEEN THE BARS BETWEEN THE LINES . YOU HAVE NOT JUST TO PLAY ONLY THE NOTES THAT ARE PUT INTO THE SCORE . BUT YOU HAVE TO GET THE FEEL FOR IT . SLOWLY SLOWLY OVER SOME YEARS GIDON KREMER TOOK UP PIAZZOLLA'S TANGO . FIRST IN MOSCOW LATER AT A SMALL CONCERT IN GERMANY THEN AT A MUSIC FESTIVAL IN AUSTRIA HE DID TWO FULL EVENINGS OF PIAZZOLLA . AUDIENCES LOVED IT . KREMER INTENSE CURIOUS ALWAYS PROBING FELT HE HAD FOUND HIS OWN VOICE IN PIAZZOLLA'S MUSIC . ASTOR PIAZZOLLA USED A VIOLINIST IN HIS GROUP BUT DIDN'T DESIGN HIS MUSIC ESPECIALLY FOR THE VIOLIN . GIDON KREMER MADE HIS OWN ARRANGEMENTS AND PUT HIS VIOLIN AT THE CENTER OF THE PIAZZOLLA MUSIC HE MADE . PIAZZOLLA BECAME PART OF MY HEARTBEAT . SO NATURAL SO MUCH IN HIS SPIRIT BECAUSE IN ITS SERIOUSNESS ITS AMBITION PIAZZOLLA'S NEW TANGO WAS LIKE THE MUSIC KREMER HAD GROWN UP WITH . QUOTE CELOS QUOTE FOR EXAMPLE QUOTE JEALOUSY QUOTE AS IT WOULD BE IN ENGLISH IS A TUNE OF SCHUBERTIAN CHANGES OF HARMONY . I WOULD NOT KNOW ANOTHER COMPOSER THAT WOULD HAVE THE COURAGE TO JUST TO BREAK A HARMONY IN THE MIDDLE OF A TUNE AND AND START A NEW CHAPTER IN SUCH A SIMPLE BUT AT THE SAME TIME INCREDIBLY SENSUAL AND INTELLIGENT WAY . I DISCOVERED THAT SOMETHING I I NEVER KNEW THAT

MY MUSIC WAS SENSUAL MUSIC . IT WAS SEXUAL . AGAIN THE LATE COMPOSER ASTOR PIAZZOLLA. I NEVER INTENDED TO DO SENSUAL MUSIC . BUT MY ONLY INTENTION WAS TO WRITE THE MUSIC I FEEL THAT I I THINK IT'S VERY RELIGIOUS . IT'S VERY MELANCHOLIC . IT'S SAD. IT COULD BE VERY SOPHISTICATED . IT COULD BE VERY AGGRESSIVE . THE ONLY THING THAT MY MUSIC DOESN'T HAVE IS THE WORD HAPPY . THERE'S NO HAPPINESS . BUT ISN'T HAPPINESS A PART OF RELIGION OF RAPTURE OF THESE RAVISHING NOTES AND CHORDS . ISN'T THERE HAPPINESS IN BEING TOUCHED BY MUSIC DRAWN TO IT CAPTURED IN IT IN BEING DEEPLY MOVED. GIDEON KREMER SAYS WE ARE BOMBARDED BY SOUND. BUT NONE OF IT SPEAKS TO OUR SOULS. WE ALL ARE VICTIMS OF NOISES IN OUR CENTURY OR THE END OF THIS CENTURY. VICTIMS OF EVERYTHING THAT IS IMPOSED ON US . WE ADAPT THE LOUDNESS OF THE STREET OF THE DISCO . WE ADAPT OURSELVES TO THE MEDIA TO THE TELEVISION RUNNING AROUND THE CLOCK . AND WHAT IS THE MOST IMPORTANT THING IN FACT FOR A HUMAN BEING OR FOR A HUMAN LIFE IS TO BE SENSITIVE TO SILENCE . AND I FEEL PIAZZOLLA DESPITE HIS EXTREMELY AGGRESSIVE TURNS OCCASIONALLY OR HIS POWER OR HIS ENERGY HE IS ABLE ALSO TO CONCENTRATE US WITHIN OURSELVES . AND VERY FEW COMPOSERS IN OUR CENTURY WERE ABLE TO DO SO . GIDON KREMER'S QUOTE HOMMAGE TO PIAZZOLLA QUOTE IS THE FIRST OF WHAT WILL BE THREE PIAZZOLLA ALBUMS . IT SEEMS THAT IN PAYING RECORDED TRIBUTES TO THE TANGO MASTER GIDON KREMER THE CLASSICAL VIOLIN VIRTUOSO HAS DISCOVERED A PART OF HIMSELF . I FEEL I NEVER COULD DANCE THE TANGO BUT I ALWAYS DREAMT ABOUT IT . THERE WAS SOMETHING QUITE SENSUAL WHICH I NEVER COULD MAYBE EXPRESS IN LIFE . BUT I SOMEHOW SUCCEEDED TO EXPRESS IN MUSIC . SO PIAZZOLLA HELPED ME ALSO TO FIND MYSELF AND I'M QUITE GRATEFUL TO HIM . IN WASHINGTON I'M SUSAN STAMBERG .

Document 3:

OUR TOP STORY TONIGHT THE FIRST DAY OF THE OKLAHOMA CITY BOMBING TRIAL . JURY SELECTION IS UNDERWAY IN DENVER COLORADO FOR THE MEN AND WOMEN WHO WILL HEAR THE GOVERNMENT'S CASE AGAINST TIMOTHY

MCVEIGH . HO WEVER IT'S SLOW GOING TODAY .C. N. N.'S TONY CLARK JOINS US LIVE FROM DENVER WITH A REPORT .TONY HOW MANY PROSPECTIVE JURORS HAVE THEY BEEN ABLE TO QUESTION SO FAR .LINDEN SO FAR ONLY SIX PROSPECTIVE JURORS HAVE BEEN QUESTIONED BY JUDGE RICHARD MATSCH AND DEFENSE ATTORNEYS AND PROSECUTORS. THE SIXTH PROSPECTIVE JUROR WILL BE CALLED BACK TOMORROW TO FINISH THE QUESTIONING WHICH GIVES YOU SOME IDEA THAT THIS JURY SELECTION PROCESS WILL TAKE A LONG TIME. AS HE LED HIS DEFENSE TEAM INTO THE FEDERAL COURTHOUSE IN DENVER MONDAY DEFENSE ATTORNEY STEPHEN JONES WAS ALL SMILES . WE'RE READY . IT HAS BEEN NEARLY TWO YEARS SINCE A BOMB RIPPED THROUGH THE FEDERAL BUILDING IN OKLAHOMA CITY KILLING ONE HUNDRED SIXTY EIGHT PEOPLE . NOW AFTER ONE OF THE LARGEST CRIMINAL INVESTIGATIONS IN U. S. HISTORY FORMER SOLDIER TIMOTHY MCVEIGH IS ON TRIAL FOR HIS LIFE ACCUSED ALONG WITH CODEFENDANT TERRY NICHOLS OF PUTTING TOGETHER THE BOMB AND DETONATING IT . INSIDE THE SECOND FLOOR COURTROOM MCVEIGH SAT QUIETLY DRESSED IN A BLUE SHIRT KHAKI PANTS AND SPORTING A CLOSE BURR HAIRCUT . HE LOOKED STRAIGHT AT EACH PROSPECTIVE JUROR AS HE OR SHE WAS BROUGHT IN TO BE QUESTIONED BY THE JUDGE AND TRIAL ATTORNEYS . THE QUESTIONS DEAL WITH THREE PRIMARY AREAS WHAT DOES THE PROSPECTIVE JUROR KNOW ABOUT THE BOMBING WHAT NEWS STORIES HAVE THEY HEARD ABOUT MCVEIGH AND PERHAPS MOST IMPORTANTLY HOW DO THEY FEEL ABOUT THE DEATH PENALTY . IF YOU ARE OPPOSED TO THE DEATH PENALTY YOU MAY NOT SIT ON THIS CASE . THE POTENTIAL JURORS ARE BROUGHT INTO THE COURTROOM INDIVIDUALLY . THEY ARE KNOWN ONLY BY NUMBERS AND ARE HIDDEN FROM VIEW FROM MOST OF THE REPORTERS AND SPECTATORS . THE FIRST JUROR NUMBER EIGHT HUNDRED FIFTY EIGHT LIVED IN TULSA OKLAHOMA AT THE TIME OF THE BOMBING AND VISITED THE BOMBING SITE THREE WEEKS AFTER THE BLAST . HE SAID HE WAS QUOTE QUITE MOVED AND STIRRED QUOTE BY WHAT HE SAW . HE SAID HE PRAYED AND CRIED . NUMBER EIGHT HUNDRED FIFTY EIGHT ALSO SAID HE KNEW ABOUT MCVEIGH'S ALLEGED . CONFESSION BUT DIDN'T NECESSARILY BELIEVE IT AND HE SAID HE THINKS THE BOMBING QUOTE RUNS FURTHER AND DEEPER THAN ONE INDIVIDUAL QUOTE . A

SECOND PROSPECTIVE JUROR REMEMBERED SEEING MCVEIGH BEING LED OUT OF THE NOBLE COUNTY COURTHOUSE . SHE SAID QUOTE I FELT VERY SORRY FOR HIM FOR SUCH A YOUNG MAN TO WASTE HIS LIFE QUOTE . MARSHA KIGHT WHOSE DAUGHTER WAS KILLED IN THE BOMBING WAS SITTING IN THE COURTROOM MONDAY . I THINK THE FAMILY MEMBERS ARE DEFINITELY LOOKING . THEY WANT THE RIGHT PERSON THEY WANT THE TRUTH . ABOUT SIXTY OTHER BOMBING SURVIVORS AND VICTIMS' FAMILY MEMBERS WATCHED THE PROCEEDINGS BY CLOSED CIRCUIT IN OKLAHOMA CITY . IT'S A LOT MORE EMOTIONAL THAN SOME OF THE PEOPLE THOUGHT . THE LAST JUROR THAT WAS QUESTIONED JUROR NUMBER EIGHT HUNDRED FIFTY ONE SAID SHE HAD PLAYED CLOSE ATTENTION TO THE BOMBING AND ALSO TO THE PREVIOUS FIRE AT THE BRANCH DAVIDIAN COMPOUND BECAUSE BOTH HAD HAPPENED ON HER BIRTHDAY APRIL NINETEENTH . IN THE COURTROOM TODAY WAS MCVEIGH'S FATHER BILL MCVEIGH AND ALSO SOME OF THE VICTIMS' FAMILIES . TOM KIGHT WHO LOST HIS STEP DAUGHTER IN THE BOMBING SAID HE HAD SYMPATHY FOR MCVEIGH'S FATHER . HE'S FIGHTING FOR A YOUNG MAN'S LIFE AND MY HEART DID GO OUT TO HIM LIKE I SAID EARLIER MCVEIGH'S FATHER BECAUSE HE HAD TO LISTEN TO SOME OF THE TESTIMONY THAT THE PROSECUTION WAS SAYING ONE HUNDRED SIXTY EIGHT PEOPLE AND A LOT OF CHILDREN AND YOU COULD SEE HIS HEAD SLUMPED . NONE OF US AS PARENTS WANT OUR CHILDREN TO BE THIS . WE ALL HAVE SOME THAT TURN INTO BAD APPLES .

I DON'T BELIEVE THAT MAN CAN . AND FROM ONE PARENT TO ANOTHER I HAVE A LOT OF COMPASSION . PROSECUTORS AND DEFENSE ATTORNEYS WILL MEET WITH JUDGE RICHARD MATSCH BEFORE COURT BEGINS SESSION TUESDAY MORNING . THEY WILL GO OVER THE INDIVIDUALS WHO WERE QUESTIONED ON MONDAY TO DECIDE WHO WILL REMAIN IN THE JURY POOL THOUGH THAT INFORMATION WILL NOT BE GIVEN TO US . THE ONE THING THAT APPEARS CERTAIN NOW IS THAT JURY SELECTION WILL BE TOUGH AND MAY TAKE A LONG TIME . LINDEN . TONY DID YOU NOTICE ANY VISIBLE REACTION FROM THE POTENTIAL JURORS OR FROM ANY OF THE FAMILY MEMBERS OR THE SURVIVORS IN THE COURTROOM TO THE PRESENCE OF TIMOTHY MCVEIGH . WE WERE UNABLE TO SEE THE POTENTIAL JURORS OR

SEE VERY MUCH OF THEM . THE WAY THE COURTROOM HAS BEEN REDESIGNED THERE IS A PARTITION THAT HIDES THE POTENTIAL JUROR THE PROSPECTIVE JUROR FROM VIEW THROUGHOUT MUCH OF THE COURTROOM AND SO OTHER THAN WHEN THEY ARE STANDING TO BE SWORN IN WE'RE UNABLE TO SEE THEM . WE DO SEE REACTIONS FROM MCVEIGH AND FAMILY MEMBERS THROUGHOUT THE DAY AND MCVEIGH AT FIRST WAS VERY QUIZZICAL IN THE COURTROOM BECAUSE THAT WAS THE FIRST TIME HE WAS IN THERE . HE WAS LOOKING AROUND A LOT AND SMILES A LITTLE BIT . MORE THAN ANYTHING ELSE HE SEEMS TO BE PAYING CLOSE ATTENTION TO THE PROCEEDINGS . ALL RIGHT . C. N. N.'S TONY CLARK IN DENVER . THANK YOU VERY MUCH . PREPARED BY FEDERAL DOCUMENT CLEARING HOUSE INCORPORATED . NO LICENSE IS GRANTED TO THE USER OF THIS MATERIAL OTHER THAN FOR RESEARCH . USER MAY NOT REPRODUCE OR REDISTRIBUTE THE MATERIAL EXCEPT FOR USER'S PERSONAL OR INTERNAL USE AND IN SUCH CASE ONLY ONE COPY MAY BE PRINTED NOR SHALL USER USE ANY MATERIAL FOR COMMERCIAL PURPOSES OR IN ANY FASHION THAT MAY INFRINGE UPON CABLE NEWS NETWORK INCORPORATED'S COPYRIGHT OR OTHER PROPRIETARY RIGHTS OR INTERESTS IN THE MATERIAL PROVIDED HOWEVER THAT MEMBERS OF THE NEWS MEDIA MAY REDISTRIBUTE LIMITED PORTIONS OF THIS MATERIAL WITHOUT A SPECIFIC LICENSE FROM C. N. N. SO LONG AS THEY PROVIDE CONSPICUOUS ATTRIBUTION TO C. N. N. AS THE ORIGINATOR AND COPYRIGHT HOLDER OF SUCH MATERIAL . THIS IS NOT A LEGAL TRANSCRIPT FOR PURPOSES OF LITIGATION .

Document 4:

ON WALL STREET THE PERFECT CLIMATE FOR A BLOCKBUSTER RALLY . A BALANCED BUDGET DEAL COUPLED WITH AN INVESTOR FRIENDLY EMPLOYMENT REPORT TOGETHER SENT STOCKS ROCKETING . THE DOW INDUSTRIALS TONIGHT AT SEVEN THOUSAND SEVENTY ONE A GAIN OF NEARLY NINETY FOUR AND THREE QUARTERS POINTS AND FOR THE WEEK THE BEST SHOWING IN TERMS OF POINTS EVER FOR THE DOW UP THREE HUNDRED THIRTY TWO POINTS . THE VOLUME WAS FIVE HUNDRED ONE MILLION SHARES AND ADVANCING ISSUES SWAMPING DECLINING ISSUES BY

MORE THAN FOUR TO ONE MARGIN . NEW YORK STOCK EXCHANGE COMPOSITE INDEX GAINED SEVEN AND ONE QUARTER POINTS . THE S. AND P. FIVE HUNDRED INDEX UP ALMOST FOURTEEN AND ONE HALF POINTS . THE DOW TRANSPORTATION INDEX GAINED SIXTY ONE AND TWO THIRDS POINTS TO CLOSE AT A RECORD TWO THOUSAND SIX HUNDRED FIVE . DOW UTILITIES GAINING FOUR AND ONE QUARTER POINTS . THE AMERICAN STOCK EXCHANGE INDEX GAINED TEN POINTS AND ON THE NASDAQ THE NASDAQ COMPOSITE GAINED THIRTY FOUR POINT EIGHT THREE ALMOST THIRTY FIVE POINTS THE BIGGEST ONE DAY GAIN EVER FOR THE NASDAQ . LET'S TAKE A LOOK AT SOME TECHNOLOGY STOCKS THAT GAINED TODAY . DELL COMPUTER UP MORE THAN FIVE DOLLARS GATEWAY TWO THOUSAND UP FOUR AND ONE QUARTER INTEL UP THREE AND SEVEN EIGHTHS MICROSOFT DOWN ONE QUARTER . IN TONIGHT'S MONEYLINE FOCUS WE TAKE A CLOSER LOOK AT THE MARKETS . JOINING ME NOW IS DEAN WITTER REYNOLDS CHIEF INVESTMENT STRATEGIST PETER CANELO . WELCOME BACK TO MONEYLINE . GOOD TO BE HERE . SO DOES IT GET ANY BETTER FROM HERE . I MEAN THIS WAS REALLY AN INCREDIBLE DAY AND WEEK . IT'S GOING TO BE A TOUGH WEEK TO FOLLOW BUT EVERYTHING WORKED OUT JUST FINE . WE'VE SEEN SIGNS OF SLOWING IN THE ECONOMY . NOTHING DANGEROUS BUT ENOUGH TO CREATE SOME DOUBT THAT THE FED IS GOING TO MOVE RATES UP AGGRESSIVELY . MAYBE THEY DON'T DO IT IN MAY BUT OBVIOUSLY THE KEY THING HERE IS THE CAPITAL GAINS TAX RATE REDUCTION AND THAT MEANS YOU GET TO KEEP NOT SEVENTY PERCENT OF WHAT YOU MAKE IN STOCKS . YOU MIGHT GET TO KEEP PERHAPS EIGHTY PERCENT OF WHAT YOU MAKE . IT'S INTERESTING THOUGH . SECRETARY RUBIN WAS SAYING THAT THAT'S NOT IN WRITING AT THIS POINT THE CAPITAL GAINS TAX CUT . YES BUT THE REPUBLICANS I THINK ARE STRONGLY IN FAVOR OF A FIFTY PERCENT EXCLUSION ON GAINS SO THAT WOULD PROBABLY BRING THE TAX RATE ON GAINS DOWN TO MAYBE NINETEEN PERCENT FROM THE CURRENT TWENTY EIGHT . SO SUDDENLY STOCKS ARE CHEAPER BECAUSE YOU'VE JUST YOU'RE GOING TO GET TO KEEP MORE OF YOUR RETURNS . BUT WHAT KIND OF EFFECT DOES IT HAVE INITIALLY IF IN FACT THIS CAPITAL GAINS TAX CUT ACTUALLY HAPPENS . PEOPLE DO FEAR THAT THERE COULD BE A LOT OF SELLING . MY EXPERIENCE IS THAT ANY TYPE OF SELLING WOULD BE

TEMPORARY AND NOT TERRIBLY SERIOUS TO THE MARKET . SUDDENLY YOUR LONG TERM INVESTOR FEELS A WHOLE LOT BETTER ABOUT BUYING STOCKS . HE'S NOT GOING TO BE SELLING STOCKS . I THINK THIS IS GOING TO HELP US TO GET TO NEW HIGHS IN THE WEEKS AHEAD . IS YOUR SENSE THAT THE FEDERAL RESERVE YOU SAY THE FEDERAL RESERVE COULD BE ON HOLD . I MEAN COULD YOU HAVE A SITUATION WHERE IN FACT RATES CAME DOWN AS A RESULT OF THE BALANCED BUDGET DEAL . WELL THE BOND YIELDS HAVE COME DOWN UNDER SEVEN PERCENT . THEY MAY COME DOWN A LITTLE MORE MAYBE DOWN TO SIX AND THREE QUARTERS BUT I THINK WE STILL HAVE A VERY GOOD ECONOMY . WE HAD A TERRIFIC G. D. P. IN THE FIRST QUARTER AND EVEN IF IT SLOWS A LITTLE BIT IN THE SPRING WE'RE NOT OUT OF THE WOODS YET . WE HAVE WE PROBABLY HAVE ONE OR TWO INTEREST RATE HIKES AHEAD OF US IN THE NEXT COUPLE OF QUARTERS SO I THINK THE MARKET HAS JUST FORGOTTEN ABOUT THAT FOR A WHILE . IT MAY COME BACK AND CREATE SOME PROBLEMS LATER ON DOWN THE ROAD . SO A CORRECTION STILL AHEAD . WELL I NO I THINK A MOVE TO NEW HIGHS IS IMMEDIATELY AHEAD . WE'LL TAKE IT ONE STEP AT A TIME . I DO THINK THE ECONOMY WOULD HAVE TO REALLY SLOW DOWN TO REALLY OBVIATE ANY INTEREST RATE INCREASES . WE STILL THINK THEY'RE GOING TO RAISE RATES BUT THEY THEY MAY GO A LITTLE MORE SLOWLY AND THAT'S GOING TO HELP THE BOND MARKET THE BANK STOCKS THE ENTIRE MARKET . WHAT IS THE BIG GAINER THOUGH IN ALL OF THIS . IS IT SMALL CAP STOCKS . YES I THINK SO FOR SEVERAL REASONS . YOU KNOW THEY'RE LITTLE COMPANIES ARE AFRAID OF HIGHER INTEREST RATES AT LEAST SOME OF THOSE FEARS MAY HAVE BEEN MUTED HERE THIS WEEK BUT SINCE THEY DON'T PAY DIVIDENDS LITTLE GROWTH STOCKS REALLY MAKE MONEY FOR YOU IN CAPITAL GAINS AND SUDDENLY YOU'RE GOING TO GET TO KEEP MORE OF THAT CAPITAL GAINS . IT'S GOT TO BE BULLISH . LITTLE STOCKS DID BETTER THAN BIG STOCKS THIS WEEK . WE MAY SEE A REVERSAL HERE FOR SOME TIME . THANKS VERY MUCH PETER CANELO CHIEF INVESTMENT STRATEGIST AT DEAN WITTER REYNOLDS . THANKS FOR JOINING US . PREPARED BY FEDERAL DOCUMENT CLEARING HOUSE INCORPORATED . NO LICENSE IS GRANTED TO THE USER OF THIS MATERIAL OTHER THAN FOR RESEARCH . USER MAY NOT REPRODUCE OR REDISTRIBUTE THE MATERIAL EXCEPT FOR

USER'S PERSONAL OR INTERNAL USE AND IN SUCH CASE ONLY ONE COPY MAY BE PRINTED NOR SHALL USER USE ANY MATERIAL FOR COMMERCIAL PURPOSES OR IN ANY FASHION THAT MAY INFRINGE UPON CABLE NEWS NETWORK INCORPORATED'S COPYRIGHT OR OTHER PROPRIETARY RIGHTS OR INTERESTS IN THE MATERIAL PROVIDED HOWEVER THAT MEMBERS OF THE NEWS MEDIA MAY REDISTRIBUTE LIMITED PORTIONS OF THIS MATERIAL WITHOUT A SPECIFIC LICENSE FROM C. N. N. SO LONG AS THEY PROVIDE CONSPICUOUS ATTRIBUTION TO C. N. N. AS THE ORIGINATOR AND COPYRIGHT HOLDER OF SUCH MATERIAL . THIS IS NOT A LEGAL TRANSCRIPT FOR PURPOSES OF LITIGATION .

Document 5:

GOOD MORNING . THIS IS THURSDAY JULY SEVENTEENTH . IN THE NEWS THIS MORNING ANOTHER SERIOUS PROBLEM ON THE RUSSIAN SPACE STATION MIR NEW EVIDENCE IN THE VERSACE MURDER INVESTIGATION . AND THE HOME RUN DERBY BETWEEN THE YANKEES' TINO MARTINEZ AND MARK MCGWIRE OF THE OAKLAND A'S . FROM A. B. C. THIS IS WORLD NEWS THIS MORNING WITH MARK MULLEN AND ASHA BLAKE . GOOD MORNING EVERYONE . THANKS FOR WAKING UP WITH US TODAY . GOOD MORNING MARK . GOOD MORNING TO YOU AND TO ALL OF YOU . WE HAVE A DEVELOPING STORY FROM THREE HUNDRED MILES ABOVE EARTH . ONCE AGAIN THERE IS A SERIOUS PROBLEM ONBOARD THE RUSSIAN SPACE STATION MIR . A. B. C.'S MIKE LEE JOINS US NOW FROM MOSCOW WITH DETAILS . MIKE . GOOD MORNING MARK . SOMETIME IN THE LAST TWELVE OR TWENTY HOURS SOMEONE WE BELIEVE IT HAD BEEN TSIBLIYEV THE MAN WHO HAD A HEART IRREGULARITY APPARENTLY DISCONNECTED AN ELECTRICAL CABLE THE WRONG CABLE ON THE SPACECRAFT WHILE THEY WERE PRACTICING FOR CABLE REPAIRS . THIS DISCONNECTED THE COMPUTER THAT CONTROLS THE GYRO . AND MIR IS MORE OR LESS SPINNING OUT OF CONTROL WHICH MEANS IT CANNOT ORIENT ITSELF TO THE SUN . THAT MEANS THE SOLAR PANELS CAN'T CHARGE . VIRTUALLY ALL OF THE POWER HAS DROPPED OUT OF MIR . THE CREW HAD TO GO TO THE SOYUZ ESCAPE CAPSULE TO MAKE A RADIO CONTACT . AND AS FAR AS I'M BEING TOLD AT THE MOMENT VIRTUALLY

NOTHING IS WORKING ON MIR BECAUSE IT IS IN THE SHADOW OF EARTH . WHAT THEY'RE TRYING TO DO IS GET ENOUGH SOLAR POWER THROUGH THE RANDOM ROTATION TODAY TO RECONNECT THE COMPUTER AND HAVE THE COMPUTER ORIENT THE GYRO AGAIN . IT'S DESCRIBED AS A PROBLEM THAT'S HAPPENED BEFORE . IT WAS A HUMAN ERROR ACCORDING TO SPACE OFFICIALS HERE BUT IT'S POTENTIALLY A SERIOUS ONE . HOW OPTIMISTIC ARE THEY AT THIS POINT MIKE THAT THEY CAN FIX IT . WELL THEY'RE STILL EXPRESSING OPTIMISM THAT WHEN THE SPACECRAFT IN A FEW HOURS COMES OUT OF THE SHADOW OF THE EARTH AND GETS SOME EXPOSURE TO THE SUN THAT THEY'LL HAVE ENOUGH BATTERY POWER AFTER THAT TO HAVE THE COMPUTER REORIENT THE GYRO WHICH WILL SOLVE THE LARGER PROBLEM . THERE IS NO GUARANTEE OF THAT . WORST CASE SCENARIO IF THEY CANNOT CORRECT THE PROBLEM THEN WHAT . WELL THEY'VE GOT ABOUT WITHIN THE MIR SPACECRAFT ITSELF THEY HAVE ABOUT TWO DAYS OF BATTERY SUPPLY BACKUP BATTERY SUPPLY . THE ULTIMATE BACKUP OF COURSE THEY GO INTO THE SOYUZ CAPSULE WHICH IS A SELF CONTAINED SPACE CAPSULE . IT'S CALLED THE DESCENT CAPSULE . THEY COULD COME BACK DOWN TO EARTH . BUT AS OF NOW THEY'RE SAYING IT IS NOT A LIKELY SCENARIO . THEY'RE STILL EXPRESSING OPTIMISM THEY CAN FIX IT . BUT OBVIOUSLY IT IS A SERIOUS SETBACK TO PLANS TO POSSIBLY TAKE THAT REPAIR MISSION NEXT WEEK . WITH AN UPDATE A. B. C.'S MIKE LEE IN MOSCOW . MIKE THANKS VERY MUCH . ASHA .

BIBLIOGRAPHY

M.-R. Amini & P. Gallinari (2002). The use of unlabeled data to improve supervised learning for text summarization. In *Proceedings of Annual ACM Conference on Research and Development in Information Retrieval*, pp. 105-112, 2002

C. Apte , F. Damerau & S.M. Weiss (1998). Text mining with decision trees and decision rules. In *Conference on Automated Learning and Discovery*, Carnegie-Mellon University, June 1998

M. Banko, V. Mittal, M. Kantrowitz & J. Goldstein (1999). Generating extraction-based summaries from hand-written summaries by aligning text spans. In *Proceedings of PACLING-99*, July 1999

M. Banko, V. O. Mittal & M. J. Witbrock (2000). Headline generation based on statistical translation. In the *Proceeding of Association for Computational Linguistics 2000*, 2000

R. Barzilay & M. Elhadad (1997). Using lexical chains for text summarization, In *Proceedings of the Workshop on Intelligent Scalable Text Summarization*, pages 10--17, Madrid, Spain, August 1997. Association for Computational Linguistics.

L.R. Baul, P.F. Brown, P.V. deSouza & R.L. Mercer (1989). A tree-based statistical language model for natural speech recognition. *IEEE Trans. On Acoustic, Speech, and Signal Processing (ICASSP)*, 36(7):1001-1008, 1989

J. R. Bellagarda (1998). A multispan language modelling framework for large vocabulary speech recognition. *IEEE Trans. Speech and Audio Proc., ASAP* 6(5):456--467, 1998.

A. Berger & J. D. Lafferty (1999). Information retrieval as statistical translation, In *Proceedings of SIGIR1999*, pp. 222-229, 1999

B. Boguraev & C. Kennedy (1997). Salience-based content characterisation of text documents. In *Proceedings of the ACL/EACL Workshop on Intellegent Scalable Text Summarization*, 1997.

P. Brown, S. Cocke, S.D. Pietra, V.D. Pietra, F. Jelinek, J. Lafferty, R. Mercer & R. Roossin (1990). A Statistical approach to machine translation, *Computational Linguistics* V. 16, No. 2, 1990

P. Brown, S.D. Pietra, V.D. Pietra, F. Jelinek & R. Mercer (1993). The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics* V. 16, No. 2, 1993

C. Chelba (1997). A structured language model. In *Proceedings of the Thirty-Fifth Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics*, 1997

P. R. Clarkson & R. Rosenfeld (1997). Statistical language modeling using the CMU-Cambridge Toolkit. In *Proceedings ESCA Eurospeech 1997*.

W.D. Climenson, N.H. Hardwick & S.N. Jacobson (1961). Automatic syntax analysis in machine indexing and abstracting. In *American Documentation*, 12(3):178-183, 1961

A.P. Dempster, N.M. Laird & D.B. Rubin (1977). Maximum likelihood from incomplete data via EM algorithm. *Journal Royal Statistical Society* Series B 39, 1-38

P. Duygulu, K. Barnard, J.F.G. d. Freitas, and D.A. Forsyth, Object recognition
as machine translation: Learning a lexicon for a fixed image vocabulary, Proc. of The Seventh European Conference on Computer Vision, Copenhagen, Denmark, pp. IV:97-112, 2002

H. P. Edmundson (1969). New methods in automatic extracting, *Jurnal of Association for Computing Machinery,* 16(2)264-285, 1969

E. Firmin & M.J. Chrzanowski (1999). An evaluation of automatic text summarization. In I. Mani and M. Maybury, editors. *Advances in Automatic Text Summarization*. MIT Press, Cambridge, Massachusetts, 1999

E. Frank, G.W. Paynter, I.H. Witten, C. Gutwin & C.G. Nevill-Manning (1999). domain-specific keyphrase extraction. In *Proceedings of the*

Sixteen International Joint Conference on Artificial Intelligence (IJCAI-99), pp. 668-673, California: Morgan Kaufmann.

J. H. Friedman (1994). Flexible metric nearest neighbor classification. *Technical Report (Nov. 1994)*.

R.P. Futrelle (1999). Summarization of diagrams in documents. In I. Mani & M. Maybury (Eds.), *Advances in Automated Text Summarization*. Cambridge, MA: MIT Press, 1999

J. Goldstein, M. Kantrowitz, V. Mittal, & J. Carbonell (1999). Summarizing text documents: sentence selection and evaluation metrics. In *Proceedings of SIGIR 99*, Berkeley, CA, August 1999.

F.B. Hildebrand (1952). *Methods of Applied Mathematics,* ch. 2, Prentice-Hall, Englewood Cliffs, NJ, 1952

D. Hiemstra & W. Kraaij (1999), Twenty-One at TREC-7: ad-hoc and cross-language track, In *Proceedings of the seventh Text Retrieval Conference TREC-7*, NIST Special Publication 500-242, pages 227-238, 1999

E. Hovy & C.Y. Lin (1997). Automated text summarization in SUMMARIST, In *Proceedings of the Workshop of Intelligent Scalable Text Summarization*, July 1997

F. Jelinek (1999). *Statistical methods for speech recognition*. MIT press, 1999.

R. Jin & A.G. Hauptmann (2000a). Title generation for spoken broadcast news using a Training Corpus, *Proceedings of the 6th International Conference on Speech and Language Processing (ICSLP 2000),* Beijing China, 2000

R. Jin & A.G. Hauptmann (2000b). Title generation using training corpus, *Proceedings of CICLING-2000*, Mexico City, Mexico, 2000

R. Jin and A. G. Hauptmann (2001). Learn to Select Good Title Word: A New Approach based on Reverse Information Retrieval. In *Proceedings of ICML 2001*.

H. Jing, R. Barzilay, K. McKeown, & M. Elhadad (1998). Summarization evaluation methods experiments and analysis. In *AAAI Intelligent Text Summarization Workshop*, pp. 60-68, Stanford, CA, Mar. 1998

T. Joachims (1998). Text categorization with support vector machines: learning with many relevant features. In *Proceeding of European Conference on Machine Learning (ECML) 1998*, Claire Nédellec and Céline Rouveirol (ed.), 1998.

K. S. Jones & P. Willett (1997). *Reading in Information Retrieval*. Morgan Kaufmann Publishers, 1997

D. Jurafsky & J. H. Martin (2000) Speech and Language, Processing Prentice Hall, 2000

S.M. Katz (1987). Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Trans. On Acoustic, Speech and Signal Proc., ASSP* 35(3):400-401, 1987

P. Kennedy & A.G. Hauptmann (2000). Automatic title generation for the Informedia Multimedia Digital Library. In *Proceeding of the ACM Digital Libraries, DL-2000*, San Antonio Texas, May 2000

K. Knight & D. Marcu (2000). Statistics-based summarization-step one: sentence compression. In *Proceedings of AAAI 2000*, 2000

J. Kupiec, J. Pedersen & F. Chen (1995). A trainable document summarizer. *Proceedings of the 18th Annual International ACM SIGIR*, pages 68-73, 1995

J. Lafferty & C. Zhai (2001a), Document language models, query models, and risk minimization for information retrieval, In *Proceedings of SIGIR 2001*, pp. 111-119, 2001

A. M. Lam-Adesina & G. J. F. Jones (2001), Applying summarization techniques for term selection in relevance feedback , In *Proceedings of SIGIR 2001*, pp. 1-9, 2001

V. Lavrenko & W. B. Croft (2001), Relevance-based language models, In *Proceedings of SIGIR 2001*, pp. 120-127, 2001

C. H. Leung & W.K. Kan (1997). A statistical learning approach to automatic indexing of controlled index terms. *Journal of the American Society for Information Science*, 48 (1), 55-66, 1997.

H.P. Luhn (1958). The automatic creation of literature abstracts. *I.B.M. Journal of Research and Development*, 2 (2), 159-165, 1958

I. Mani & M. Maybury (1999). *Advances in Automated Text Summarization*. Cambridge, MA: MIT Press, 1999

I. Mani, T. Firmin, D. House, G. Klein, B. Sundheim & L. Hirschman (1999). The TIPSTER SUMMAC text summrization evaluation, In *Proceedings of EACL 1999*, 1999

D. Marcu (1997). From discourse structures to text summaries. In *Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization,* pages 82--88, Madrid, Spain, August 1997. Association for Computational Linguistics.

A. McCallum & K. Nigam (1998). A comparison of event models for naive bayes text classification. *AAAI-98 Workshop on "Learning for Text Categorization"*, 1998

K. McKeown, J. Robin & K. Kukich (1995). Designing and evaluating a new revision-based model for summary generation. *Information Process and Management* 31, 5, 1995

D. Miller, T. Leek & R. M. Schwartz (1999). A hidden Markov model information retrieval system. In *Proceedings of SIGIR'1999,* pp. 214-222, 1999

V. Mittal , M. Kantrowitz , J. Goldstein & J. Carbonell (1999). Selecting text spans for document summaries: heuristics and metrics, In *Proceedings of the sixteenth national conference on artificial intelligence and eleventh innovation applications of AI conference on Artificial intelligence and innovative applications of artificial intelligence*, p.467-473, July 18-22, 1999, Orlando, Florida, United States

MUC-6 (1995), *Proceeding of The Sixth Message Understanding Conference*, 1995

T. Niesler (1997), Category-based statistical language models, PhD thesis, Dept. of Engineering, University of Cambridge, 1997

T.R. Niester & P.C. Woodland (1996). A variable-length category-based n-gram language model. In *Proceedings of ICASSP'96*, 1996

H. Nye (1984). The Use of a one stage dynamic programming algorithm for Connected Word Recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, Vol. AASP-32, No 2, pp. 262-271, April 1984.

C.D. Paice (1990). Constructing literature abstracts by computer: Techniques and prospects. *Information Processing and Management*, 26 (1), 171-186, 1990

C.D. Paice & P.A. Jones (1993). The identification of important concepts in highly structured technical papers. *In Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 69-78, 1993

S.D. Pietra, V.D. Pietra & J. Lafferty (1997). Inducing features of random. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 19(4), April, 1997, pp. 380-393

J. Ponte & W. B. Croft (1998). A language modeling approach to information retrieval. In *Proceedings of SIGIR' 1998*, pp. 275-281, 1998

M.F. Porter (1980). An algorithm for suffix stripping. *Automated Library and Information Systems*, 14 (3), 130-137, 1980

W.H. Press, S.A. Teukolsky, W.T. Vetterling, W. T. & B.P. Flannery (1993). *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1993

Primary Source Media (1997). Broadcast News CDROM, Woodbridge, CT, 1997

J. R. Quinlan (1986). Induction of decision trees. *Machine Learning*, vol. 1, pp. 81--106, 1986.

D. Radev & K. McKeown (1998). Generating natural language summaries from multiple online sources. *Computational Linguistics* 24, 3, pp. 469-501, September 1998

L. Rau, P.S. Jacobs & U. Zernick (1989). Information extraction and text summarization using linguistic knowledge acquisition. In *Information Processing and Management*, 25(4):419-428, 1989

U. Reimer & U. Hahn (1988). Text condensation as knowledge base abstraction. In *Proceedings of Fourth Conference on Artificial Intelligence Applications*, 338-344, 1988.

V. Rjiesbergen (1979). Information Retrieval. Chapter 7. Butterworths, London, 1979.

S.E. Robertson et al.(1993). Okapi at TREC-4. In *The Fourth Text Retrieval Conference (TREC-4)*. 1993

S.E. Roberson & S. Walker (1999). Okapi/Keenbow at TREC-8. In E.M. Voorhees and D.K. Harmann, editor, *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*, Gaithersburg, 1999

R. Rosenfeld (1994). Adaptive Statistical language modeling: a maximum entropy approach, PhD thesis, 1994, Carnegie Mellon University

G. Salton (1971). *The SMART Retrieval System: Experiments in Automatic Document Proceeding*, Prentice Hall, Englewood Cliffs, New Jersey, 1971

G. Salton & C. Buckley (1988). Term-weighting approaches in automatic text retrieval, *Information Processing and Management*, 24, 513—523, 1988

G. Salton, J. Allan, C. Buckley & A. Singhal (1994). Automatic analysis, theme generation, and summarization of machine-readable texts. *Science*, 264, 1421-1426, 1994

G. Salton, A. Singhal, M. Mitra & C. Buckley (1997). Automatic text structuring and summary. *Information Process And Management*, 33(2):193-207, March, 1997.

R. E. Schapire, Y. Singer & A. Singhal (1998). Boosting and Rocchio applied to text filtering. In *SIGIR '98: Proceedings of the 21st Annual International Conference on Research and Development in Information Retrieval*, 1998.

E.F. Skorokhodko (1972). Adaptive method of automatic abstracting and indexing. In *IFIP Congress*, Ljubljana, Yugoslavia 71, 1179-1182, 1972

T. Strzalkowski, J. Wang & B. Wise (1998). A robust practical text summarization system. In *AAAI Intelligent Text Summarization Workshop*, pp. 26-30, Stanford, CA, March, 1998.

S. Teufel & M. Moens (1998). Sentence extraction and rhetorical classification for flexible abstracts. In *AAAI Spring Symposium on Intelligent Text Summarization*, 1998.

P. D. Turney (1997). Extraction of keyphrases from text: evaluation of four Algorithms. *National Research Council, Institute for Information Technology, Technical Report ERB-1051*, 1997

P.D. Turney (2000). Learning algorithms for keyphrase extraction. *Information Retrieval*, 2(4): 303-336, 2000

T.A. vanDijk (1980). Macrostructures: An interdisciplinary study of global structures in discourse, interaction, and cognition. Lawrence Erlbaum, Hillsdale, NJ, 1980

E. Voorhees & D. Harman (ed.) (1996), *The Fifth Text REtrieval Conference (TREC-5)*, NIST Special Publication 500-238, 1996

M. Witbrock & V. Mittal (1999). Ultra-Summarization: A statistical approach to generating highly condensed non-extractive summaries. In *Proceedings of SIGIR 99*, Berkeley, CA, August 1999

Y. Yang & C.G. Chute (1994). An example-based mapping method for text classification and retrieval, *ACM Transactions on Information Systems (TOIS)*, 12(3): 252-77. 1994

Y. Yang (1997). An evaluation of statistical approaches to text categorization. Information Retrieval Vol. 1(2), pp. 69—90, 1997

Y. Yang (2001). A study on thresholding strategies for text categorization. In *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'01)*, pp 137-145, 2001

K. Zechner (2001). Automatic generation of concise summaries of spoken dialogues in unrestricted domains. *In Proceedings of SIGIR 2001*, pp. 199-207, 2001

C. Zhai & J. Lafferty (2001b). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceeding of SIGIR'01, 2001*, pp.334-342, 2001