RESEARCH PROPOSAL

Automatic Generation of Informative Headlines for Text and Speech

David M. Zajic
Department of Computer Science
University of Maryland College Park

Advisor: Dr. Bonnie Dorr
Committee: Dr. Douglas Oard, Dr. William Gasarch

Area: Natural Language Processing
Sub-discipline: Text Summarization

Abstract

This research proposal describes methods for automatically constructing very short informative summaries, or headlines, for documents. Two existing systems for constructing headlines, a statistical system (HMM Hedge) and a linguistic-heuristic system (Hedge Trimmer) are described. Extrinsic and intrinsic evaluations of these two systems, are described.

The proposed research will investigate the effectiveness of different methods for producing English headlines for English news articles, non-English news articles and transcribed broadcast news. I will conduct experiments to determine if headlines are more useful than topic lists for humans making relevance judgments, and whether query-specific headlines perform better than generic headlines in the context of Information Retrieval. User studies will discover whether features such as topic coverage, brevity, fluency and clarity contribute to the usefulness of a headline in various usage contexts. Another goal of the research is to determine whether summary evaluation measures must take into consideration the intended use for a summary in order to agree with extrinsic measures of human performance on a task using a summary.

The main contributions of this research will be (1) an approach to document summarization that combines linguistic and statistical methods in various ways to support different types of input documents and different summary usages and (2) a methodology for evaluating document summaries that takes into consideration the intended summary usage.

Contents

Introduction

Headline generation is a form of text summarization in which the summarization is required to be informative and extremely short, and to mimic the condensed language features of newspaper headlines. Informative summaries answer the questions "what happened in this document?" or "what is claimed in this document?," rather than the question "what is this document about?" The last question is best answered by an indicative summary, such as a topic list. Informative summaries are harder to create than indicative summaries, because they must conform to fluency, clarity and accuracy. It is also difficult to determine what is the primary event or situation described by a document. The intended use for the summary and the interests of the user affect the suitability of any particular headline.

In my work to date I have developed two approaches to headline generation. The first is primarily based on statistical methods, and the second is primarily based on linguistically-informed heuristics. These two systems represent a growing pool of techniques related to creating headlines that can be combined into systems with varying properties. I will explore the question of which techniques perform best with respect to document type and intended use.

The hypotheses to be tested are:

1. That it is possible to automatically construct informative headlines in English for various types of documents.
2. That the techniques for producing informative headlines will vary in their effectiveness depending on the type of input document (English news articles, non-English news articles; transcribed English broadcast news)
3. That informative summaries (headlines) are more useful than indicative summaries (topic lists) in the context of Information Retrieval.
4. That the usefulness of an informative summary will vary depending on certain features, including: topic coverage; brevity; fluency; clarity.
5. That the usefulness of an informative summary in the context of Information Retrieval will vary with the appropriateness of the summary to the specific query.
6. That the degree to which a summary evaluation technique agrees with an extrinsic measure of human performance on a task using a summary will vary with the purpose for which the summary is used.

In order to test hypotheses, I will refine and develop new techniques for constructing informative summaries for English news articles. These techniques will be extended to apply to non-English news articles and transcribed broadcast news, and new techniques developed as needed. I will conduct more experiments of the type described in section 2.4.4 on headlines constructed by various means to determine whether informative headlines are more useful in the context of Information Retrieval than topic lists or a baseline extract of comparable length from the front of the article.

The usefulness of a headline can depend as much on the interest of the user as on the content of the document. Consider the following news story from the Washington Post, June 10, 2003:

> A D.C. police officer and a man with a rifle exchanged gunfire on a Northeast Washington street in a bizarre confrontation yesterday afternoon that ended with the gunman stripping himself naked and a police dog biting another officer. ...
> The only injuries were scratches suffered by the alleged gunman, Damien J. Lee, 26, and a minor bite to the officer's knee. ...
> The man was blocking traffic, she said, and pointing the rifle – described by police as an M1 carbine with the stock sawed off – at nearby people and cars. He was yelling profanities, Williams said, and saying, "I'm God."

Is this a story about police dogs, traffic disruptions or people who claim to be God? A copy editor at the Post provided this article with two base-covering headlines:

> Gunman Says, 'I'm God,' Blocks Traffic, Fires, Strips
> Police Dog Bites Officer on the Knee In Struggle to Arrest Naked NE Man

This extreme example illustrates that a single article might be relevant to many diverse queries, and that a single all-purpose headline that represents every important facet may be impossible.

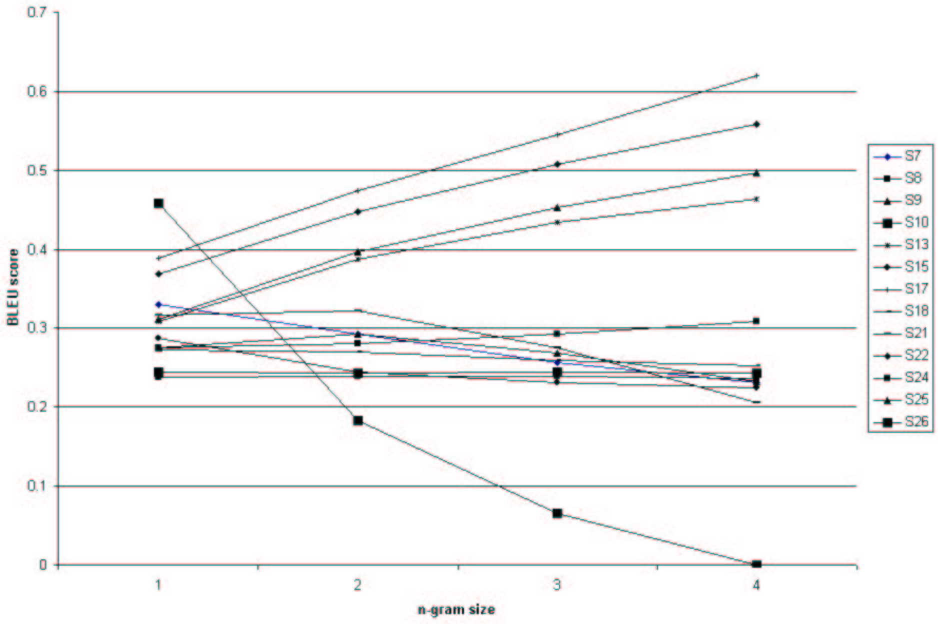

Figure 1: BLEU Score vs BLEU n-gram configuration for DUC2003 submissions

It is important for any evaluation technique to agree with human judgment, however in the case of the usefulness of summaries, humans may not be able to judge the usefulness of a headline in the abstract. The more important question is how well an evaluation method corresponds to improved human performance on a task using summaries. Different tasks may require different characteristics from a summary. For instance, in a situation where recall is more important than precision, one might favor short, fluent summaries. However, if precision is more important than recall, one would favor longer, topic-rich summaries. Different evaluation techniques can reveal different properties.

Figure 1 shows the BLEU scores of the systems that were submitted to the DUC 2003 Text Summarization evaluation as the BLEU evaluation is performed with different maximum n-gram sizes. The systems that rise in rank as the BLEU n-gram configuration rises are also the systems that provided fluent summaries. The system which sinks in rank as n-grams increase produced topic lists with no fluency. Changing the configuration of the BLEU evaluation can have a powerful effect on the the features that it will value.

The first section in this report presents a literature review of related work in three areas: Text Summarization, Statistical Machine Translation (MT) and Document Selection in Information Retrieval (IR). Section 2 describes the problem of headline generation in more detail and presents two prototype headline generation systems and an extrinsic evaluation of them in an IR context. Section 3 describes the proposed research and future work.

Chapter 1

Literature Review

1.1 Statistical Machine Translation

Statistical methods have been applied to many areas of Natural Language Processing (NLP) that support Machine Translation (MT), such as parsing (Booth and Thomson, 1973), word sense disambiguation (Gale, Church, and Yarowsky, 1992) and text alignment (Gale and Church, 1993). The first fully statistical approach to MT (Brown et al., 1990) uses an idea from Information Theory, the Noisy Channel Model. Suppose that an information signal that has been distorted by transmission through a noisy medium. If we know enough about the properties of the original signal and the nature of the distortion we can reconstruct the original signal. In (Brown et al., 1990) the source sentence in French is treated as a noisy version of an unseen sentence in English. There are three components to a Noisy Channel system: a source language model, a channel model and a decoder. The source language model gives the prior probability of a string of words as an English sentence, $P(e)$. This can be estimated with a n-gram model in which the probability of an English sentence $e = \{e_1, e_2, ..., e_n\}$ is

$$P(e) = P(e_1) \cdot P(e_2|e_1) \cdots P(e_n|e_{n-1})$$

The channel model, or translation model, gives the probability $P(f|e)$ that French sentence f is the translation of English sentence e. The goal of the decoder is to discover the English sentence that will maximize $P(e|f)$. By Bayes' rule

$$\operatorname*{argmax}_e P(e|f) = \operatorname*{argmax}_e \frac{P(f|e)P(e)}{P(f)}$$

Because $P(f)$ is a constant with respect to e, it is equivalent to find

$$\operatorname*{argmax}_e P(f|e)P(e)$$

Because there is not always a one-to-one relationship between words in English and French and the words are not always in the same order, (Brown et al., 1990) defines the notions of fertility and distortion. The fertility of an English word is the number of French words that an English word produces. Distortion is the extent to which a French word appears at a position in the sentence different from the English word which produced it. The parameters of the source model are the n-gram probabilities of the words in the vocabulary of the source language. There are three parameters to the translation model:

- Fertility probabilities, $P(n|e)$, the probability that word e generates n target words,
- Translation probabilities, $P(f|e)$, the probability that a f is the translation of e, and
- Distortion probabilities, $P(i|j,l)$, the probability that a target word produced by a source word at position j in a target sentence of length l appears at target position i.

(Brown et al., 1993) describes methods for estimating these parameters.

(Wu and Wong, 1998) proposes replacing the translation model with a stochastic inversion transduction grammar (SITG). In an ITG all terminal symbols come in couples, x/y, where for example x is a Chinese word and y is an English word. Each production has either straight orientation or inverted orientation.

- Straight orientation: VP → [VP PP]
- Inverted orientation: VP → ⟨VP PP⟩

Rules with straight orientation visit the right-hand-side symbols from left to right for both languages. Rules with inverted orientation visit the right-hand-side symbols left to right for Chinese, but right to left for English. This permits an ITG tree to produce surface strings with correct word order for both Chinese and English. In a SITG a probability is associated with each production, so a probability $P(c,e,q)$ can be assigned to all generable trees q, Chinese sentences c and English sentences e. This probability can replace the translation model in the expression

$$\hat{e} = \operatorname*{argmax}_e P(c,e,q)P(e)$$

The approach has the advantage that it is possible to eliminate the distortion and fertility probabilities from the translation model.

1.2 Text Summarization

Text summarization, as a task performed by humans, involves reading and understanding a document for content, then generating a new document expressing a consise version of the content. The first efforts at automatic text summarization consisted of selecting important sentences from a document and concatenating them together. (Luhn, 1958) uses term frequencies to measure sentence relevance. Sentences are included in the summary if the words in the sentence have high enough term frequencies. Luhn used a stoplist to exclude common words with little topic-specific value, such as prepositions and determiners, and also aggregated terms by orthographic similarity. (Salton, 1988) observed that documents in a certain domain will share certain common words beyond the obvious stop words. Terms that are common for the domain will have high term frequency in all documents about that topic, and thus are not good topic-sentence indicators. The relevance of term in a document is inversely proportional to the number of documents in the collection containing the term. For example, documents about classical music will often contain the terms *classical*, *music* and *composer*, and so these terms are not good indicators of topic within the classical music domain. A measure which takes both term frequency within a document and term rarity in the general collection is $tf_i * idf_i$. tf_i is the frequency of term i in a document and idf_i is the inverted document frequency where,

$$idf_i = log(\frac{N}{dtf_i})$$

N is the number of documents in the collection and dtf_i is the number of documents containing term i. Techniques involving term frequency have been extended to concept frequency using WordNet (Hovy and Lin, 1999). Thus an occurrence of the concept *fruit* is counted for every term that is a hypernym of fruit in WordNet.

Summaries consisting of extracted sentences suffer when the selected sentences contain anaphoric expressions, such as pronouns and definite noun phrases, that can only be understood with respect to their antecedents in earlier sentences. Also, these systems are unable to generate summaries shorter than the text-spans being extracted. (Witbrock and Mittal, 1999) proposed a method capable of producing summaries of arbitrary length. They use a corpus of documents and summaries to calculate the conditional probability that a word occurs in a summary given that it occurs in that summary's document. They also use a bigram language model of the summaries to calculate the probability of a possible summary. These probabilities are weighted together to calculate the score of a possible summary. A Viterbi beam search is used to find the near-optimal summary with summary length as a parameter.

Sentence compression is another way to produce summaries that are not limited to selecting entire sentences from a document. (Knight and Marcu, 2000) introduces a Noisy-Channel approach to translating long sentences into short sentences. Their source model considers not only the bigram probability of the surface string, but also a probabilistic context-free grammar (PCFG) score, computed over the grammar rules that were applied to yield the parse tree computed by the Collins parser (Collins, 1997a). The channel model describes how a larger tree is grown from a smaller tree by use of an *expansion template* based on the labels of the nodes and their children. For instance, a prepositional phrase is added to an S node with probability $\mathrm{P(S \rightarrow NP\ VP\ PP \mid S \rightarrow NP\ VP)}$.

(Daumé et al., 2002) extends the approach of (Knight and Marcu, 2000) by using Rhetorical Structure Theory (RST) (Mann and Thompson, 1988) to improve the coherency of the compressed sentences. The source model is extended to bias against incoherently juxtaposed sentences and the channel model is capable of adding discourse units to the summary as well as syntactic constituents to the sentences.

Text summarization has been extended to transcribed broadcast news by (Jin and Hauptmann, 2001) and (Hori et al., 2002). In (Jin and Hauptmann, 2001), several different approaches were applied to broadcast speech to try to discover good title words. They report three approaches that showed promise. In the Naive Bayesian approach with limited vocabulary, similar to (Witbrock and Mittal, 1999), for all words that appeared in titles in the training data, they calculated the conditional probability that the word appeared in the title if it appeared in the document. In TF*IDF the words with the highest TF*IDF scores were chosen as title words. In the k nearest neighbor approach, the training corpus is searched for the most closely related document, and the title words for that document are used. The titles produced were scored by calculating precision and recall on individual words compared to human-generated titles, so it does not appear that grammaticality was considered in their evaluation. (Hori et al., 2002) points out that summarization for transcribed speech must deal with problems not encountered for written documents, such as disfluencies, filled pauses, repairs, word fragments and and irrelevant information caused by recognition errors.

1.3 Document Selection in Information Retrieval

The Information Retrieval process begins with a user having an information need, and proceeds into the following iterative sequence of steps (Hearst, 1999):

1. The user formulates a query

2. The user sends the query to the search system.

3. The search system returns results to the user.

4. The user scans, evaluates, and interprets the results.

5. The user stops, or,

6. The user reformulates the query and goes to step 2.

In this process there are two points at which the user provides important input: the query formulation and the evaluation of results. Users evaluate the documents in the result sets in various ways, including gathering information to refine the query, discovering a new and different information need, deciding that a document might contain links to other relevant or interesting documents, and most basically deciding whether the document is relevant to the query. Deciding whether the document is relevant is called *document selection.*

The most common way that result sets are shown to users is in lists ranked by computed relevance to the query. These lists typically include the document's title along with some important metadata, such as date, source, length and a score indicating the computed degree of relevance. These collections of data, which serve to represent a document in a small amount of computer screen space, are called *document surrogates.* If the user suspects, based on the document surrogate, that a document might be relevant the user can look at the entire document by clicking on the title or an iconic representation of the document.

One way of focusing the users attention on the relevant section of a document is to highlight the query terms within the document in a contrasting color. (Landauer et al., 1993). The concept is translated into a different kind of surrogate called *keyword-in-context* (KWIC). In KWIC surrogates sentences containing a high concentration of query words are extracted from the document to illustrate the context of the query words in the document.

There are also graphical forms of document surrogates. In TileBars (Hearst, 1995) the user specifies a small number of query terms. A document is represented by stacked horizontal bars, one for each query term. From left to right the bars represent the document from beginning to end. The bars are divided into vertical slices. Each slice represents the presence of the term in a particular window of a document. The color of the slice represents the density of the query term in the window. Users can tell from TileBars whether a particular term is present in a document and whether it is a central theme of the document or is mentioned sparsely.

The most common method for evaluating IR systems that produce ranked lists of documents is the mean uninterpolated average precision (MAP) measure. Map is defined as:

$$MAP = E_i[E_j[\frac{j}{r(i,j)}]]$$

where $E_i[]$ is the average over a set of queries, $E_j[]$ is the average over the documents relevant to query i, and $r(i,j)$ is the rank of the j^th relevant document for query i. (Oard, 2001).

However, some researchers have argued that improvements in MAP do not necessarily correlate with improvements on user performance on various tasks. (Hersh et al., 2000) describes an experiment in which subjects used two different IR systems, one which ranked by TF*IDF and the other which ranked by an Okapi weighting scheme which had been shown to significantly outperform TF*IDF on MAP. The users were asked to do an instance recall task. An instance recall query asks the subject to find documents which provide an answer to a specific question, such as "What countries import Cuban sugar?". The results of the experiment showed that the subjects did not perform significantly better at the instance recall task when they saw results sorted by the Okapi weighting scheme over the TF*IDF weighting scheme. (Turpin and Hersh, 2001) reproduced these results for a question answering task, and suggested that the subjects were able to recognize relevant documents by looking at the documents or the surrogates so well that the order in which the documents were presented only slowed them down or required them to make more queries.

What if the documents and surrogates are in a language the user cannot read? In the case of cross-language information retrieval (CLIR) the user has active (writing, speaking) language skills in the query language and passive (reading) skills or no skills in the document language. In this case, MT is used to provide the user with a document surrogate the user can read well enough to determine if the document is worth spending the resources required for creating a fluent translation. (Oard, 2001) suggests an extension to MAP that takes into consideration the cost of a false positive, i.e. selecting an irrelevant document:

$$C = k \cdot p_r \cdot MAP + (1-k)(1 - p_f(1 - MAP))$$

where p_r is the probability of correctly recognizing a relevant document, p_f is the probability of selecting for translation an irrelevant document, and $k \in [0,1]$ is a parameter reflecting the relative importance of limiting the number of documents examined or limiting the number of irrelevant documents translated.

(Oard et al., 2003) suggests an experiment design for evaluating CLIR systems. Subjects are shown a topic, in English, and a ranked list of document surrogates produced by a CLIR system, and asked to determine whether the documents are "relevant," "somewhat relevant," or "not relevant" to the topic. A Latin square is used to ensure that different subjects different combinations of topics and CLIR systems and to avoid the confounding effects of subject fatigue, subject learning, and topic or system order. The principal measure of effectiveness is a weighted harmonic mean of precision and recall:

$$F_\alpha = \frac{1}{\alpha/P + (1-\alpha)/R}$$

where P is precision, R is recall and α is reflects a bias of value towards P or R. (van Rijsbergen, 1979)

(Wang and Oard, 2001) describes an experiment of this type in which word-for-word glosses of documents were compared to machine translation. This experiment showed that subjects shown either glosses or machine translations were able to perform better document selection than the baseline of selecting all the documents. Subjects did perform better at document selection using machine translation over glosses, but the difference was not significant. This may have been because too few subjects (4) were used.

Chapter 2

Headline Generation

2.1 Motivation

This section defines the notion of a headline as an extremely short informative summary, discusses my approach to automatic headline generation at an implementation independent level and describes some initial studies that justify the approach.

2.1.1 Headlines

Newspaper articles are usually associated with a headline. Headlines are written by copy editors after an article is complete. The copy editors try to construct headlines that satisfy three goals: to summarize the story, to draw in the reader and to fit in the specified space. (Rooney and Witte, 2000). In order to achieve these goals, headline writers adopt a form of compressed English, sometimes referred to as Headlinese (Mårdh, 1980). Some differences between Headlinese and standard English are the omission of determiners and forms of the verb "to be", and use of present tense for events in the past. A headline that summarizes a story is an *informative* abstract, whereas a headline that identifies the topic or topics of a story is an *indicative* abstract. Automatic generation of informative headlines is an important goal because there are documents for which an informative headline in English is not provided, such as non-English articles and transcriptions of broadcast news. Moreover, the headlines provided with newspaper articles are often not informative.

Consider the following headlines.

1. Under God Under Fire
2. Pledge of Allegiance
3. U.S. Court Decides Pledge of Allegiance Unconstitutional

Headline 1 is designed to make the reader curious enough to read the article in order to find out what is going on. A reader could guess what happens in the article if the reader already knew that a court was considering the constitutionality of the pledge of allegiance. Headline 2 tells the reader that the topic is the pledge of allegiance, but does not tell what happened. Our goal is to produce headlines like Headline 3 that tell what happens.

In some articles, especially those describing the content of a recently published report or study, the main theme of the article is not an event, but instead is a claim about something. Thus for an article about the publication of a study showing that schizophrenia is related to brain chemistry, Headline 5 is preferable to Headline 4.

4. Yale Researcher Publishes Article in Science Magazine

5. Yale Study Says Schizophrenia Related to Brain Chemistry

2.1.2 Constructing Headlines by Selecting Story Words

Informative headlines can often be constructed by selecting words in order from story words found in the article. Allowing morphological variation of the story words in the headline makes it possible to mimic the features of Headlinese. To determine the feasibility of this approach, I attempted to apply the technique by hand. I examined 56 stories randomly chosen from the Tipster corpus. Taking hand-selected story words in the order in which they appeared I was able to construct fluent and accurate headlines for 53 of the stories. The remaining 3 stories were a list of commodity prices, a chronology of events and list of entertainment events. From this I concluded that this approach has promise for stories that are written as paragraphs of prose.

Only 7 of the 53 headlines used words beyond the 60th story word, and of those only one went beyond the 200th word. This suggests that headlines can often be constructed from words taken from the front of the article. Stories whose headlines required the later words tended to be human-interest stories with tangential, attention-whetting introductions, or appeared to be excerpts from the middle of larger stories.

In another experiment, two subjects were asked to write headlines for 73 AP stories from the Tipster corpus for January 1, 1989 by selected words in order from the story. Of the 146 headlines, 2 did not meet the story-words-in-order criterion because of accidental word reordering. At least one fluent and accurate headline meeting the criterion was created for each of the stories. The average length of the headlines was 10.76 words. Another way to examine the results is to consider the distribution of the headline words among the sentences of the stories, i.e. how many came from the first sentence of a story, how many from the second sentence, etc. The results of this study are shown in Figure 2.1. 86.8% of the headline words were chosen from the first sentence of the story.

In a subsequent study two subjects created 100 headlines for 100 AP stories from August 6, 1990. 51.4% of the headline words in the second study were chosen from the first sentence. The distribution of headline words for the second set is shown in Figure 2.2.

Although humans do not always select headline words from the first sentence, we observe that a large percentage of headline words are often found in the first sentence, and that the incidence of headline words chosen from sentences trails off quickly as the sentences are farther into the story. This coincides with the informal observation that news stories are often written with a lead sentence and lead paragraph that summarize the story.

Consider the following excerpt from a news story and corresponding headline.

6. After months of debate following the Sept. 11 terrorist highjackings, the Transportation Department has decided that airline **pilots** will **not** be **allowed to have guns in** the **cockpits**.

7. Pilots not allowed to have guns in cockpits.

The bold words in 6 form a fluent and accurate headline (7).

This basic approach has been realized in two ways. The first, HMM Hedge, uses a method analogous to Statistical MT to find the most likely headline for a given story. The second, Hedge Trimmer, uses empirically-motivated heuristics to remove grammatical constituents from the lead sentence until it meets a shortness requirement.

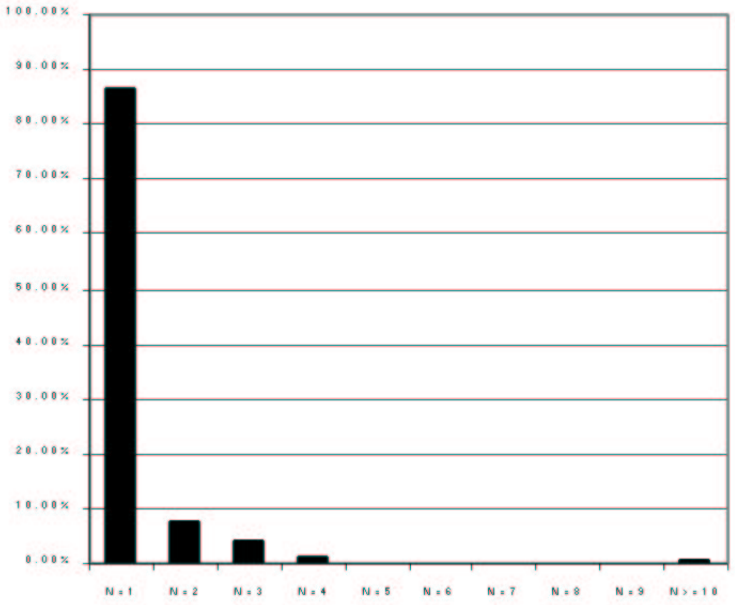

Figure 2.1: Percentage of words from human-generated headlines drawn from Nth sentence of story (Set 1)

2.2 Statistical Headline Generation

HMM Hedge (Hidden Markov Model HEaDline GEnerator) is a statistical approach to headline generation. This approach to headline generation is similar to statistical machine translation in that the observed story is treated as the garbled version of an unseen headline transmitted through a noisy channel. The noisy-channel approach has been used for a wide range of Natural Language Processing (NLP) applications including speech recognition (Bahl, Jelinek, and Mercer, 1983), machine translation (Brown et al., 1990), sentence boundary detection (Gotoh and Reynolds, 2000), spelling correction (Mays, Damerau, and Mercer, 1990), language identification (Dunning, 1994), part-of-speech tagging (Cutting, Pedersen, and Sibun, 1992), syntactic parsing (Collins, 1997c) (Charniak, 1997), semantic clustering (Lin, 1998) (Pereira, Tishby, and Lee, 1993), sentence generation (Langkilde and Knight, 1998) (Bangalore and Rambow, 2000), and text summarization (Knight and Marcu, 2000). I apply a similar technique to a new domain: automatic generation of headlines from stories.

2.2.1 Language Models and Hidden Markov Models

The intuition behind the algorithm is to treat the observed data (articles) as the result of unobserved data (headlines) that have been distorted by transmission through a noisy channel. The effect of the noisy channel is to add story words between the headline words and to change the morphology of some headline words. The task is to find the headline most likely to have generated a given story. That is, each story word is taken to be generated either from a headline word or from a general story language model. A headline word can generate a story word which is identical to it, or one which is a morphological variant of it. Thus stories consist of headline words (or morphological variants of headline words) with many other words interspersed amongst them.

Formally, if H is an ordered subset of the first N words of story S, we want to find the H which

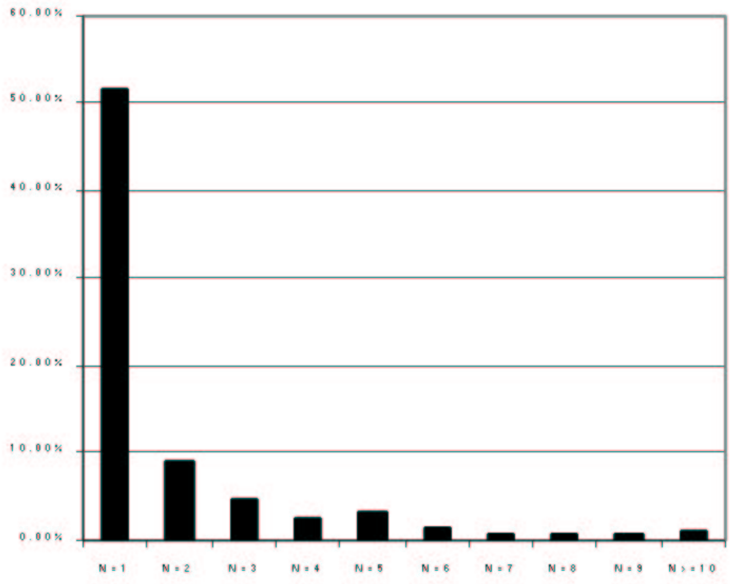

Figure 2.2: Percentage of words from human-generated headlines drawn from Nth sentence of story (Set 2)

maximizes the likelihood that H generated S, or:

$$argmax_H P(H|S)$$

It is difficult to estimate $P(H|S)$, but this probability can be expressed in terms of other probabilities that are easier to compute, using Bayes' rule:

$$P(H|S) = \frac{P(S|H)P(H)}{P(S)}$$

Since the goal is to maximize this expression over H, and $P(S)$ is constant with respect to H, $P(S)$ can be omitted. Thus we wish to find:

$$argmax_H P(S|H)P(H)$$

Let H be a headline consisting of words $h_1, h_2, ..., h_n$. The special symbols *start* and *end* represent the beginning and end of a headline. $P(H)$ is estimated using the bigram probabilities of the words in the headlines:

$$P(H) = P(h_1|start)P(h_2|h_1)...P(h_n|end)$$

The bigram probabilities of the headline words were calculated from a corpus of 714,184 English headlines from the Tipster corpus. The headlines contain 8,692,181 words from a vocabulary of 146,702 distinct words.

To estimate $P(S|H)$ we must consider the process by which a headline generates a story. This process can be represented by a Hidden Markov Model (HMM). A HMM is a weighted finite-state automaton in which each state probabilistically emits a string. The simplest HMM for generating headlines is shown in Figure 2.3. Consider the story in Sentence 6. The H state will emit the words

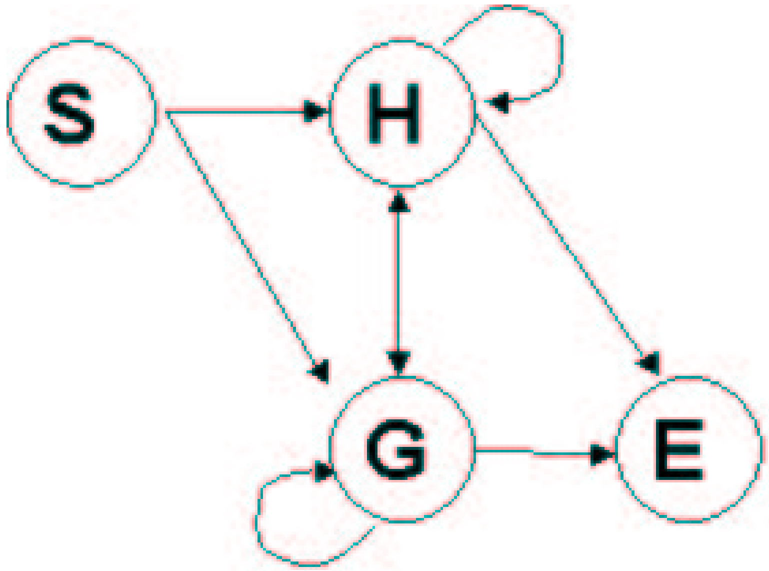

Figure 2.3: Simplest HMM to generate stories from headlines

in bold (pilots, not, allowed, to, have, guns, in, cockpits), and the G state will emit all the other words. The HMM will transition between the H and G state as needed to generate the words of the story.

Because a bigram model of headlines is used the HMM in Figure 1 will not be sufficient. For unconstrained headlines there would have to be an H state for each word in the headline vocabulary. However, because the headline words are chosen from the story, it will be sufficient to have an H state for each word in the story. Each H state will have a corresponding G state which emits story words until the next headline word and remembers the previously emitted headline word. The HMM for a three-word story is shown in Figure 2.4.

The G states emit the non-headline words in the story. A G state can emit any word in the story language model. The story language is represented by a unigram model. The unigram probabilities were calculated from a corpus of 790,942 English stories from the Tipster corpus. The stories contain 285,507,715 words from a vocabulary of 1,434,988 distinct words.

For any story, the HMM is consists of a start state S, end state E, an H for each word in the story, and a corresponding G state S and each H state. Each H state can emit only its particular word. The corresponding G state remembers which word was emitted by its H state and can emit any word in the story language. A headline corresponds to a path through the HMM from S to E that emits all the words in the story in the correct order. In practice the HMM is constructed with states for only the first N words of the story, where N is a constant (60), or N is the number of words in the first sentence. Limiting consideration to early part of the story is justified by the intuition that newspaper articles tend to begin with lead sentences and lead paragraphs that summarize the story. Other methods of selecting the window of story words are possible and will be explored in future research.

$P(S|H)$ is the probability of the story words that are inserted amongst the headline words. For a given story and headline, if we let $W = w_1, w_2, ..., w_m$ be the words from the story which are not

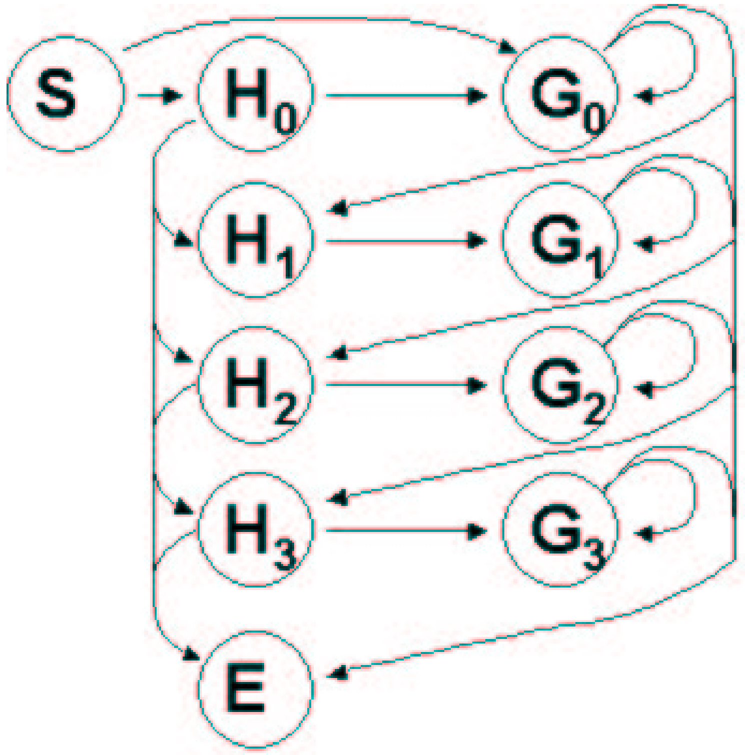

Figure 2.4: HMM for a three-word story

in the headline, and $P(w_i)$ be the unigram probability in the story language of w_i then

$$P(S|H) = P(w_1)P(w_2)...P(w_m)$$

Consider the story in 6. The HMM for this story will have states $Start, G_{start}, End$ and 28 H states with 28 corresponding G states. The headline given in 7 corresponds to the following sequence of states: $Start, G_{start}$ 18 times, $S_{pilots}, G_{pilots}, S_{not}, G_{not}, S_{allowed}, S_{to}, S_{have}, S_{guns}, S_{in}, G_{in}, S_{cockpits}, End.$ This path is not the only one that could generate the story in 6. Other possibilities are:

8. Transportation Department decided airline pilots not to have guns.

9. Months of the terrorist has to have cockpits.

Although 8 and 9 are possible headlines for 6, the conditional probability of 9 given 6 will be lower than the conditional probability of 8 given 6.

Transitions from H states to other H states correspond to a *clump* of sequential headline words in the story. A transition from an H state to a G state corresponds to the end of a clump and the start of a *gap*, i.e. a headline word followed by a non-headline word.

Conversely, a transition from a G state to an H state corresponds to the end of a gap and the start of a clump.

2.2.2 Morphological Variation

Morphological variation is allowed by creating multiple H states with corresponding G states for story words. Thus the word *said* in a story can be generated by the headline words *says, saying, said,* or *say.* This is particularly important for verbs. News stories are typically written in the past tense, while headlines are written in the present tense. Without morphological variation, the language model of headlines will bias against the forms of verbs most often seen in stories.

2.2.3 Viterbi Decoding

A Viterbi algorithm is used to select the most likely headline for a story. A two dimensional array of cells is constructed with a row for each state in the HMM and a column for each word in the observed story. Each cell contains a log probability and a backtrace to a cell in the previous column. The cells in the first column are initialized so that the log probability of the start state is 0 and all others are negative infinity. The subsequent columns are filled in with reference to the contents of the previous column.

The H state cells are assigned as follows. For each H state in the previous column, add to the log probability in that cell the log probability that the current story word follows the headline word emitted by that H state in the headline language model. For each G state in the previous column, add to the log probability in that cell the log probability that the current story word follows the headline word emitted by the H state corresponding to that G state. Then select the highest log probability to store in the current cell and include a backtrace to the cell in the previous column from which that log probability was calculated.

The G state cells are assigned as follows. There are only two states in the previous column which can transition to a given G state: the corresponding H state and the G state itself. Select the one with the highest log probability and add to it the log probability that the current story word is generated by the story model, and set the backtrace.

Once the final column is filled in, follow the backtraces from the cell for the end state and end symbol. Include in the headline only the words which were emitted by H states.

2.2.4 Length Controls

In order to ensure that a headline of the desired length is produced, it is not enough to bias the system in favor of shorter or longer headlines. The Viterbi algorithm above is modified so that a headline is found for every length in the desired range. Each cell contains a subcell for each possible headline length. When looking into the previous column, for H cells consider only subcells of length one less that the current column, because H cells contribute a word to the headline. For G cells, which do not contribute a word to the headline, consider only subcells with the same headline length.

I observed that stories had inherent length biases. Some stories favored longer headlines while others favored shorter headlines. While I do not have an explanation for it at this time, I have adjusted for this phenomenon by adjusting the scores of the best headlines at each length by the slope of the least-squares fit line of the plot of length vs. score.

2.2.5 Decoding Parameters

Three decoding parameters are used to help the system choose the headlines that best mimic actual headlines: a position penalty, a string penalty and a gap penalty. Note that the incorporation of these parameters changes the values in the cells from log probabilities to relative desirability scores.

The three parameters were motivated by intuitive observations of the output and their values were set by trial and error. A logical extension to this work would be to attempt to learn the best setting of these parameters, e.g., through Expectation Maximization (Collins, 1997b).

In the human-constructed headlines, the headline words tended to appear near the front of the story because many newspaper stories begin with a topic sentence that states the main point of the story. The position penalty is used to favor headlines which include story words near the front of the story. The initial position penalty p is a positive number less than one. The story word in the nth position is assigned a position penalty of $log(p^n)$.

When an H state emits a story word, the position penalty is added to the desirability score. Thus words near the front of the story carry less of a position penalty than words farther along. This generalization doesn't hold in the case of human interest and sports stories that start with a hook to get the reader's attention, rather than a topic sentence.

We observed that in the human-constructed headlines, there were often contiguous strings of story words in the headlines. 7 illustrates this with the string "allowed to have guns." The string penalty is used as a bias for "clumpiness", i.e., the tendency to generate headlines composed of strings of contiguous story words. The log of the string penalty is added to the desirability score with each transition from an H state to its G state. A string penalty lower than one will cause the algorithm to prefer clumpy headlines.

In human-constructed headlines, Very large gaps between headline words tends to be a sign of great effort from the human to piece together a headline from unrelated words. I believe that the algorithm would not be nearly as successful as the humans in constructing large gap headlines, and that allowing it to try would cause it to miss easy, non-gappy headlines.

The gap penalty is used to bias against headline "gappiness", i.e., the tendency to generate headlines in which contiguous headline words correspond to widely separated story words. At each transition from a G state to a H state, a gap penalty is applied which depends on the size of the gap since the last headline word was emitted. This can also be seen as a penalty for spending too much time in one G state. Low gap penalties will cause the algorithm to favor headlines with few large gaps.

2.2.6 Multiple Results

Keeping only one backpointer per cell meant that only one headline was produced for each length. In order to get multiple possible headlines at each length, the algorithm is expanded to include the n-best back-pointers for each cell. This allows the system to produce many possible headlines which can then be scored by global measures, such as how well the headline can be parsed.

2.2.7 Linguistic Improvement: Requiring Verbs

Based on the observation that most headlines contain at least one verb, the statistical system rejects all headlines that do not contain any verbs. Verbs are recognized by using the BBN SIFT parser (Miller et al., 1998) as a part-of-speech tagger. Moreover, only verbs are allowed to have morphological variants. Unconstrained morphological variation results in undesirable substitutions such as *Kim il Singing* for *Kim il Sung.*

2.2.8 Cross-Language Headline Generation

HMM Hedge has been extened to generating English headlines for non-English stories. A scored bilingual lexicon takes the place of the database of morphological variation. For each word in the story, an H state is created for each possible translation of that word. The headline language

Level	Phenomenon	Count	Percentage
Headline	preposed adjuncts	0/212	0%
Headline	conjoined S	1/218	0.5%
Headline	conjoined VP	7/218	3%
Noun Phrase	relative clause	3/957	0.3%
Noun Phrase	determiner	31/957	3%
Clause	time expression	5/315	1.5%
Clause	trailing PP	165/315	52%
Clause	trailing SBAR	24/315	8%

Table 2.1: Percentages found in human-generated headlines

model for English is preserved, while a new story language model and a scored bilingual lexicon is required for each new language. This approach has been implemented for Hindi. At present, each translation above a minimum score is treated as equiprobable. In future work, the probability of an English word given a source language word will be taken into consideration.

2.3 Parse-and-Trim Headline Generation

The second approach to constructing headlines by selecting words in order from a story, called Hedge Trimmer, removes grammatical constituents from a parse of the lead sentence until a length threshold has been met. The parses are created by the BBN SIFT parser. As described in (Miller et al., 1998) the BBN SIFT parser builds augmented parse trees according to a process similar to that described in (Collins, 1997a). The BBN SIFT parser has been used successfully for the task of information extraction in the SIFT system (Miller et al., 2000).

The approach taken by Hedge Trimmer is most similar to that of (Knight and Marcu, 2000), where a single sentence is shortened using statistical compression. However, Hedge Trimmer uses linguistically motivated heuristics for shortening the sentence. There is no statistical model, so prior training on a large corpus of stories and headlines is not required.

2.3.1 Hedge Trimmer Algorithm

The input to Hedge Trimmer is a story. The first sentence of the story is passed through the BBN SIFT parser. The parse-tree result serves as input to a linguistically motivated module that selects story words to form headlines based on key insigts gained from observations of human-constructed headlines. That is, the headlines constructed by humans in the studies described in 2.1.2 were analyzed for the purpose of developing the Hedge Trimmer algorithm.

218 human-constructed headlines were parsed by the BBN SIFT parser. This parse run produced 957 noun phrases (NP) and 315 clauses (S). At each level (headline, noun phrase and clause), linguistic phenomena were counted. These are the phenomena that were considered for trimming. At headline level, the number of headlines containing preposed adjuncts, conjoined clauses and conjoined verb phrases were counted. At the NP level, the number of NPs containing determiners and relative clauses were counted. At the S level, the number of clauses containing time expressions, trailing SBARs and trailing PPs were counted. The results are shown in table 2.1

For comparison, the same phenomena were counted for parses of the first sentences of 73 AP stories from the Tipster corpus for January 1, 1989. The parser results included 817 noun phrases and 316 clauses. The counts and percentages of the phenomena in the first sentences of stories are

Level	Phenomenon	Count	Percentage
Headline	preposed adjuncts	2/73	2.7%
Headline	conjoined S	3/73	4%
Headline	conjoined VP	20/73	27%
Noun Phrase	relative clause	29/817	3.5%
Noun Phrase	determiner	205/817	25%
Clause	time expression	77/316	24%
Clause	trailing PP	184/316	58%
Clause	trailing SBAR	49/316	16%

Table 2.2: Percentages found in story first sentences

shown in Table 2.2.

The comparison of the prevalence of these phenomena in human-generated headlines to story first sentences suggests that they are reasonable choices for trimming, with the exception of trailing PPs. Thus special care is taken to remove PPs late in the process and to reduce the likelihood of removing a PP which important content.

Hedge Trimmer uses the following algorithm for parse-tree trimming:

1. Choose the lowest leftmost S with NP, VP
2. Remove low content units
   (a) Some determiners
   (b) Time expressions
3. Iterative shortening
   (a) XP-over-XP reduction
   (b) Remove preposed adjuncts of root S
   (c) Remove trailing PPs
   (d) Remove trailing SBARs

The steps of the Hedge Trimmer algorithm will be described in more detail in the following sections.

2.3.2 Choose the Correct S Node

The first step relies on what is referred to as the *Projection Principle* in Linguistic theory (Chomsky, 1981). Predicates project a subject (both dominated by S) in the surface structure. The human-generated headlines studied in section 2.3.1 always conformed to this rule. Thus it has been adopted as a constraint in the Hedge Trimmer algorithm that the lowest leftmost S node which has as children both a NP node and a VP node in that order is taken to be the root node of the headline.

An example of the application of this step is shown in (10). The boldfaced material in the parse is retained and the italicized material is eliminated.

10. Input: Rebels agreed to talks with government officials, international observers said Tuesday.
Parse: *[S* **[S [NP Rebels][VP agreed to talks with government officials]]**, *international observers said Tuesday.]*
Output: Rebels agreed to talks with government officials.

When the parser produces a correct tree, this step provides a grammatical headline. However, the parser often produces incorrect output. When the parser was run on the 624-sentence DUC-2003 evaluation set, human evaluation of the output revealed that there were two such scenarios.

11. Parse: **[S[SBAR What started as a local controversy][VP has evolved into an international scandal.]]**

12. Parse: **[NP[NP Bangladesh][CC and][NP[NP India][VP signed a water sharing accord.]]]**

In 11, an S exists, but it does not conform to the requirements of the Projection Principle because it does not have as children a NP followed by a VP. This occurred in 2.6% of the sentences in the DUC-2003 evaluation data. The problem is resolved by selecting the lowest leftmost S, whether or not it is the parent of an NP followed by a VP.

In 12, no S is present in the parse. This occurred in 3.4% of the sentences in the DUC-2003 evaluation data. This problem is resolved by selecting the root of the entire parse tree as the root of the headline.

2.3.3 Removal of Low Content Nodes

Step 2 of the algorithm removes low-content units from the parse tree. The simplest low-content unites are the determiners *a* and *the.* Other determiners are not considered for deletion because the analysis of the human-constructed headlines revealed that most of the other determiners provide important information, e.g., negation (not), quantifiers (each, many, several), and deictics (this, that).

Few of the human-generated headlines contained time expressions, which, although certainly not content-free, are not vital for summarizing the theme of an article. Since the goal is to provide an informative headline, the identification and elimination of time expressions allows other more important details to remain in the length-constrained headline.

Time expressions are identified with BBN's IdentiFinder$^{\text{TM}}$(Bikel, Schwartz, and Weischedel, 1999). The elimination of time expressions is a two step process.

1. Use IdentiFinder$^{\text{TM}}$to mark time expressions.

2. Remove [PP ... [NP [X] ...] ...] and [NP [X]] where X is tagged as part of a time expression

The following examples illustrate the application of time expression removal.

13. Input: The State Department on Friday lifted the ban it had imposed on foreign fliers.
Parse: **[S [NP***[Det The]* **State Department** *[PP [IN on] [NP [NNP Friday]]]* **[VP lifted** *[Det the]* **ban it had imposed on foreign fliers.]]**
Output: State Departement lifted ban it had imposed on foreign fliers.

14. Input: An international relief agency announced Wednesday that it is withdrawing from North Korea.
    Parse: [S [NP *[Det An]***international relief agency**][**VP announced** *[NP [NNP Wednesday]]* **that it is withdrawing from North Korea.**]]
    Output: International relief agency announced that it is withdrawing from North Korea.

53.2% of the first sentences in the DUC-2003 evaluation data contained at least one time expression that could be removed. Human inspection of 50 deleted time expressions showed that 38 were desirable deletions, 10 were locally undesirable because they introduced and ungrammatical fragment, and 2 were undesirable because the removed a potentially relevant constituent. However, even an undesirable deletion often pans out for two reasons: (1) the ungrammatical fragment is frequently deleted later by some other rule; and (2) every time a constituent is removed it makes room under the threshold for some other, possibly more relevant constituent. Consider the following examples

15. At least two people were killed Sunday.

16. At least two people were killed when single-engine airplane crashed.

Headline (15) was produced by a system which did not remove time expressions. Headline (16) shows that if the time expression Sunday were removed, it would make room below the 10-word threshold for another important piece of information

2.3.4 Iterative Shortening

The final step, iterative shortening, removes linguistically peripheral material through successive deletions of constitutents until the sentence is shorter than a given threshold. The headline length threshold is a configurable parameter. There are four types of iterative shortening. For each type of shortening, the positions in the parse tree where it is possible to apply the shortening rule are found, and then the shortening rule is applied to those positions from the deepest, rightmost back until the headline is under the length threshold. When a shortening rule has been applied at all possible places in the parse tree and the headline is still above the length threshold, the algorithm moves to the next type of shortening rule. The three shortening rules are

1. XP-over-XP Reduction

2. Remove preamble of root S

3. Remove trailing PPs

4. Remove trailing SBARs

XP-over-XP reduction is implemented as follows: In constructions of the form [XP [XP ...] ...] remove the other children of the higher XP, where XP is NP, VP or S. This is a linguistic generalization that allows the application of a single rule to capture three different phenomena: relative clauses, ver-phrase conjunction and sentential conjunction. The rule is applied iteratively, from the deepest rightmost applicable node backwards, until the length threshold is reached.

The impact of XP-over-XP reduction can be seen in these examples of NP-over-NP (relative clauses), VP-over-VP (verb-phrase conjunction) and S-over-S (sentential conjunction), respectively.

17. Input: A fire killed a firefighter who was fatally injured as he searched the house.
    Parse: [**S**[*Det A*]**fire killed**[*Det a*][**NP [NP firefighter**[*SBAR who was fatally injured as he searched the house.*]]]]
    Output: Fire killed firefighter.

18. Input: Illegal fireworks injured hundreds of people and started six fires.
    Parse: [**S Illegal fireworks [VP [VP injured hundreds of people**][*CC and*] [*VP started six fires.*]]]
    Output: Illegal fireworks injured hundreds of people.

19. Input: A company offering blood cholesterol tests in grocery stores says medical technology has outpaced state laws, but the state says the company doesn't have the proper licenses.
    Parse: [**S**[*Det A*]**company offering blood cholesterol tests in grocery stores says [S [S medical technology has outpaced state laws,**][*CC but*] [*S the state says the company doesn't have the proper licenses.*]]]
    Output: Company offering blood cholesterol tests in grocery stores says medical technology has outpaced state laws.

The motivation for removal of preposed adjuncts of shortening is that all of the human-generated headlines ignored what we refer to as the *preamble* of the story. Assuming the Projection Principle has been satisfied, the preamble is viewed as the phrasal material occuring before the subject of the sentence. Thus, adjuncts are identified linguistically as any XP unit preceding the first NP (the subject) under the S chosen by step 1. Note that this step is not iterative, but it included here because it is only applied if the first step of iterative shortening has not reduced the headline below the threshold length. The impact of preposed adjunct removal can be seen in example (20).

20. Input: According to a now finalized blueprint described by U.S. officials and other sources, the Bush administration plans to take complete, unilateral control of a post-Saddam Hussein Iraq.
    Parse: [**S**[*PP According to a now finalized blueprint described by U.S. officials and other sources*], [*Det the*]**Bush administration plans to take complete, unilateral control of**[*Det a*]**post-Saddam Hussein Iraq.**]
    Output: Bush administration plans to take complete unilateral control of post-Saddam Hussein Iraq.

The third and fourth types of iterative shortening are the removal of trailing PPs and SBARs, respectively. These are the riskiest of the iterative shortening rules, as indicated in the analysis of the human-generated headlines. Thus, these rules are applied last, only when there are no other categories of rules to apply. Moreover, these rules are applied with a backoff option to avoid over-trimming the parse tree. First the PP shortening rule is applied. If the threshold has been reached, no more shortening is done. However, if the threshold has not been reached, the system reverts to the parse tree as it was before any PPs were removed, and applies the SBAR shortening rule. If the threshold still has not been reached, the PP rule is applied to the output of the SBAR rule. The intuition is that when removing constituents from a parse tree, it's best to remove smaller portions during each iteration to avoid producing trees with very few words. PPs tend to represent small parts of the tree while SBARs represent large parts of the tree. Thus we try to reach the threshold by removing small constituents, but if we can't reach the threshold that way, we restore the small constituents, remove a large constituent and resume the deletion of small constituents.

In an effort to reduce the risk of removing PPs containing important information, BBN's IdentiFinder^{TM}is used to distinguish PPs containing a named entity. PPs containing named entities are not removed during the first round of PP removal. However, PPs containing named entities that are descendents of SBARs are removed before the parent SBAR is removed. The reason is that we should try to reach the threshold by removing a small constituent before removing a larger constituent that subsumes it.

The impact of these two types of shortening can be seen in examples (21) and (22).

21. Input: More oil-covered sea birds were found over the weekend. Parse: **[S More oil-covered sea birds were found]***[PP over the weekend]*]
    Output: More oil-covered sea birds were found.

22. Input: Visiting China Interpol chief expressed confidence in Hong Kong's smooth transition while assuring closer cooperation after Hong Kong returns.
    Parse: **[S Visiting China Interpol chief expressed confidence in Hong Kong's smooth transition***[SBAR while assuring closer cooperation after Hong Kong returns.]*
    Output: Visting China Interpol chief expressed confidence in Hong Kong's smoot transition.

Other sequences of shortening rules are possible. The one above was observed to produce the best results on a 73-sentence development set of stories from the TIPSTER corpus.

2.3.5 Cross-Language Headline Generation

At present, Hedge Trimmer is applied to the problem of cross-language headline generation by translating the first sentence of a story into English and running the Hedge Trimmer process on the resulting translation. The obvious drawback is that it requires a translation process.

2.4 Evaluation

I conducted two evaluations on the two headline generation systems, HMM Hedge and Hedge Trimmer. One was an informal human assessment and one was a formal automatic evaluation. In addition an earlier version of HMM Hedge was submitted to the DUC 2002 Text Summarization Workshop evaluation and Hedge Trimmer was submitted to the DUC 2003 Text Summarization evaluation. Hedge Trimmer was used to produce surrogates for an experiment submitted to the Interactive track for the 2003 Cross-Language Evaluation Forum. These evaluations and experiments will be described in this section.

2.4.1 BLEU: Automatic Evaluation

BLEU (Papineni et al., 2002) is a system for automatic evaluation of machine translation. BLEU use a modified n-gram precision measure to compare machine translations to reference human translations. In my evaluation of headline generation systems, I treat summarization as a type

System	AP900806		DUC2003	
	Score	Avg Len	Score	Avg Len
HMM60	0.0997 ± 0.0322	8.62	0.1050 ± 0.0154	8.54
HMM 1st Sentence	0.0998 ± 0.0354	8.78	0.1115 ± 0.0173	8.95
Hedge Trimmer	0.1067 ± 0.0301	8.27	0.1341 ± 0.0181	8.50

Table 2.3: BLEU results

of translation from a verbose language to a concise one, and compare automatically generated headlines to human generated headlines.

For this evaluation I used 100 headlines created for 100 AP stories from the TIPSTER collection for August 6, 1990 as reference summarizations for those stories. These 100 stories had never been run through either system prior to this evaluation. I also used the 2496 manual abstracts for the DUC2003 10-word summarization task as reference translations for the 624 test documents of that task. I used two variants of HMM Hedge, one which selects headline words from the first 60 words of the story, and one which selects words from the first sentence of the story. Table 2.3 shows the BLEU score using trigrams and the 95% confidence interval for the score.

These results show that although Hedge Trimmer scores slightly higher than HMM Hedge on both data sets, the difference is not statistically significant. However, I believe that the difference in the quality of the systems is not adequately reflected by this automatic evaluation.

2.4.2 Human Evaluation

Human evaluation indicates significantly higher scores than might be guessed frm the automatic evaluation. For the 100 AP stories from the TIPSTER corpus for August 6, 1990, the output of Hedge Trimmer and HMM Hedge was evaluated by the author. Each headline was given a subjective score from 1 to 6, with 1 being the worst and 5 being the best. The average score of HMM Hedge was 3.01 with standard deviation of 1.11. The average score of Hedge Trimmer was 3.72 with standard deviation of 1.26. Using a t-score the difference, though not great, is significant with greater than 99.9% confidence.

The types of problems exhibited by the two systems are qualitatively different. HMM Hedge is more likely to produce an ungrammatical result or omit a necessary argument, as in the examples below.

23. HMM60: Nearly drowns in satisfactory condition satisfactory condition

24. HMM60: A county jail inmate who noticed.

In contrast, Hedge Trimmer is more likely to fail by producing a grammatical but semantically useless headline.

25. HedgeTr: It may not be everyone's idea especially coming on heels.

Finally, even when both systems produce acceptable output, Hedge Trimmer usually produces headlines which are more fluent.

26. a. HMM60: New Year's eve capsizing
    b. HedgeTr: Sightseeing cruise boat capsized

2.4.3 DUC Evaluations

HMM Hedge was submitted to the Document Understanding Conference 2002 Workshop on Text Summarization evaluation. In this evaluation, single-document summaries were judged on grammaticality and coverage of important topics. The system performed poorly with respect to the grammaticality questions, in part because it was designed to immitate Headlinese, and performed poorly on coverage because the headlines were on average less than 10 words long. HMM Hedge received a score for mean coverage of topics of 0.06, the lowest of the 13 systems submitted. However when the score was adjusted for length, HMM Hedge received a score of 0.30, the highest score. Primarily this reflected a defect in the length adjustment formula and the fact that the evaluation was not geared towards extremely short summaries, or headlines.

Hedge Trimmer was submitted to the Document Understanding Conference 2003 Workshop on Text Summarization evaluation. In the 2003 evaluation, a separate task was set up for very short summaries and 13 systems were submitted. Not all of the systems attempted to produce headlines, and in this evaluation no attention was paid to grammaticality, only to topic coverage. Hedge Trimmer received a coverage score of 0.275 and a length-adjusted score of 0.267 placing the system 9th and 6th respectively. Using the BLEU scoring described in section 2.4.1, Hedge Trimmer placed 3rd among the 13 systems.

2.4.4 iCLEF Experiment

Hedge Trimmer was used in an experiment for the Interactive track for the 2003 Cross-Language Evaluation Forum. In this experiment two methods were used to produce English surrogates for Spanish documents. Surrogate A, FIRST40, consisted of the first 40 words of a machine translation of the document. Surrogate B, HEDGETRIMMER, was a headline constructed by Hedge Trimmer from the machine translation of the first sentence. Eight subjects were shown surrogates for the results of IR searches on eight topics. The translations and search results were provided by iCLEF to all participants. Each search result consisted of 50 documents. For each topic, the subjects were shown a description of the topic and surrogates for the 50 documents. The subjects were asked to judge whether the document was highly relevant, somewhat relevant or not relevant to the topic and whether they were highly confident, somewhat confident or not confident in their relevance judgment. The order of topics, and whether the subject saw FIRST40 or HEDGETRIMMER for a particular topic was varied according to the Latin Square provided by iCLEF as part of the standard experiment design.

My goal was to show that the two surrogates had close recall and precision, but that HEDGETRIMMER took the subjects less time to perform the task. Subjects were able to complete 1189 judgments in a total of 290:34 minutes with FIRST40, while they completed 1388 judgments in 272:37 minutes with HEDGETRIMMER. That is, using FIRST40 subjects made 4.092 judgments per minute, while with HEDGETRIMMER they made 5.091 judgments per minute. However the results of the experiment showed that over 32 searches FIRST40 had an average F_α of 0.473 and HEDGETRIMMER had an average F_α 0.379.

Inter-annotator agreement did not differ much between the two systems. I used Cohen's κ (Cohen, 1960) to measure the pairwise inter-annotator agreement. κ is 0 when the agreement between annotators is what would be expected by chance, and is 1 when there is perfect agreement. The average κ over all pairs of distinct subjects was 0.245537. Because of the experiment design, some pairs of subjects never used the same surrogate for the same topic; some pairs always used the same surrogate for the same topic; and some pairs saw some topics with the same surrogate and others

	Surrogate A	Surrogate B
Question 1	2.09	1.97
Question 2	3.65	2.91
Question 3	3.75	3.28
Question 4	3.78	3.13

Table 2.4: Average Question Responses by System

with different surrogates. The system-specific κ scores were calculated using only the judgments for which both subjects had seen the same surrogate to make the judgment. The average pairwise κ score for FIRST40 was 0.260087 and the average pairwise κ score for HEDGETRIMMER was 0.270415.

After the subjects completed judging the documents for a topic, they were asked the following questions:

1. Were you familiar with this topic before the search?
2. Was it easy to guess what the document was about based on the surrogate?
3. Was it easy to make relevance judgments for this topic?
4. Do you have confidence in your judgments for this topic?

The subjects answered each question by selecting a number from 1 to 5, where 1 meant "not at all", 3 meant "somewhat" and 5 meant "extremely." The responses are shown in Table 2.4.

I do not take this result necessarily to mean that informative headlines are worse surrogates than the first forty words. It is likely that the headlines used in Surrogate B were not good enough headlines to make a conclusion about informative summaries in general. Also, the average length of the headlines used in the Surrogate B was much shorter than forty words, giving Surrogate A the advantage of including more topic information.

Chapter 3

Proposed Research

Although the existing systems show promise, they have not yet reached a point at which they are consistently producing headlines that are useful for any task. An important element of the proposed research will be to continue development of headline generation systems. The next step is to apply those systems to different types of input documents and in different extrinsic tasks and refine the systems to these environments. Experiments will compare how well headlines support human performance on an extrinsic task with respect to topic lists, sentence extraction, first-N-words and other summarization approaches. Finally, task-sensitive evaluation methods will be developed that coincide with human performance on extrinsic tasks.

3.1 HMM Hedge

Several improvements are planned for the HMM Hedge system. They include:

- Use of a higher-order n-gram model of both the story and headline language models. At present the story model is a unigram model and the headline model is a bigram model. I would like to use a bigram model of stories and a trigram model of headlines to improve the grammaticality of the output.

- Better selections of the window of story words from which the headline words are chosen. Often, but not always, the lead sentence of news articles is the first sentence of the document. However in human interest stories, sports stories and opinion pieces, this is not usually the case. It is not likely that this will be the case for transcriptions of broadcast news, and certainly not for transcribed conversations. I have experimented with using the window of words having the maximum tf.idf score (Sparck-Jones, 1972) with respect to the document and the corpus. This produced lower BLEU scores, possibly because the window with the highest tf.idf score tended to occur late in the article as specific names and details were mentioned. Best results were achieved by limiting the story word window to the first sentence. This could be improved by using BBN's Unsupervised Topic Detection (UTD) system (Schwartz, Sista, and Leek, 2001) or sentence extraction summary systems, e.g. (Conroy and O'Leary, 2001), to guide the selection of story word windows.

- Use of the probabilities of morphological variations given story words in the HMM. At present, all morphological variants of a verb are treated as equiprobable. However, given the linguistic features of headlines, it is more likely that present tense of a verb in a headline will correspond to a past tense verb in a story. The HMM should reflect this.

- Identification of certain key words as no-skip words. UTD could be used to identify words that should either be forced into the headline, or should be given document-specific probability beyond the probability of the language models.

- Re-tokenizization of certain strings of words, so that if any of them appear in a headline they all must appear in a headline. This could be used to force the preservation of certain names, such as *Department of Computer Science* instead of *Department*, or replace long names with shorter variants, such as *President Bush* instead of *Presdident George W. Bush*.

- Use of Expectation Maximization (Collins, 1997b) to deterimine the best settings for parameters such as the gap, position and string penalties.

- Testing of headlines for properties at the global level, such as how many fragments a parser recognizes, whether there is subject verb agreement, and whether verbs and prepositions have their required arguments.

3.2 Hedge Trimmer

The following improvements are planned for the Hedge Trimmer system:

- As with HMM Hedge, selection of the right sentence or sentences to compress is vital to producing a useful headline. UTD and sentence extraction technologies will also be applicable to Hedge Trimmer.

- Production of multiple results. There are many possible orderings of constituent removal. It would be possible to over-generate a large number of possible trimmings of the parse tree, and use the language models from HMM Hedge to rank them.

- Studies to determine how humans would trim parse trees. I performed several user studies to see how humans constructed headlines by selecting words from stories in the manner of HMM Hedge. The same could be done for Hedge Trimmer by providing a graphical user interface (GUI) for tree trimming, and allowing subjects to remove constituents from lead sentences until the length threshold has been met, and studying the types of constituents removed, and the order in which they were removed.

- Use of UTD to preserve constituents containing important topics.

3.3 Cross-Language

Both HMM Hedge and Hedge trimmer have been applied to cross-language headline generation, but there is much work to be done here. For HMM Hedge, I treat translations of words from the source language to English as if they were morphological variations. This approach has been applied to Hindi. For cross-language headline generation, having indendent probabilities for different possible translations is vital. Another important enhancement would be to allow word reordering from the story to the headline.

For Hedge Trimmer, I translate the lead sentence into English and then run the existing Hedge Trimmer process. This approach is obviously fraught with peril, for it depends on the quality of the translation system. This approach has been applied to Cebuano and Hindi. The current trimming heuristics are highly specific to English, however I plan to work with native speakers of other languages to determine if a language-independent approach can be abstracted, and parameterized for use with a variety of source languages. If so, it would be possible to simplify and shorten a lead sentence in the source language, and translate the result into English.

I plan to extend both systems for cross-language headline generation to Spanish and Arabic. At present it is not clear which approach, statistical or heuristic, produces better results in the cross-language application.

3.4 Speech Transcripts

I plan to apply both systems to speech transcriptions of broadcast news and conversations. Determining which window of a speech transcript will be a significant issue, because I do not expect broadcast news to follow the same discourse patterns as written news articles. Also, speech transcripts will be much noisier than written text, because of errors in the transcription and speech disfluencies in the source. Determination of sentence boundaries is also an important issue. As with cross-language headline generation, it is an open question which techniques will perform best for transcribed speech.

3.5 Extrinsic Tasks

In order to test the hypothesis that the important features of a summary will vary with the intended use of the summary, we will evaluate summaries in the context of extrinsic tasks that make use of summaries. To date I have performed experiments with summaries used as surrogates for documents in Information Retrieval systems. Other possibilities include Question Answering systems and Foreign Language Tutoring systems.

3.6 Experiments

I will perform experiments similar to those described in 2.4.4 to determine if acutal headlines are more useful in the IR task than topic lists. Experiments will be designed to determine how well headlines or topic lists can convey the theme of an article, and how important knowledge of the theme of an article is to correctly judging relevance to a topic. Subjects will be asked to state the theme of an article based on reading a headline, a topic list, or the article. BLEU scoring could be used to determine whether themes written by the readers of headlines or topic lists were closer to those written the readers of articles. Also it could be determined whether those subjects whose themes were most correct were also able to make the best relevance judgments.

3.7 Evaluation

Developing automatic headline evaluation methods that correspond to human performance on a specific extrinsic task will be an important component of this research. Different features, such as brevity, clarity, topic coverage and fluency, could be weighted differently for different extrinsic tasks.

Appendix A: Reading List

Text Summarization

Banko, M., V. Mittal, and M. Witbrock. 2000. Headline Generation Based on Statistical Translation. In *Proceedings of the 38th Meeting of Association for Computational Linguistics*, pages 218–235, Hong Kong.

Conroy, John and Dianne P. O'Leary. 2001. Text summarization via hidden markov models and pivoted qr matrix decomposition. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, Louisiana.

Daumé, H., A. Echihabi, D. Marcu, D. Munteanu, and R. Soricut. 2002. GLEANS: A Generator of Logical Extracts and Abstracts for Nice Summaires. In *Workshop on Automatic Summarization*, pages 9–14, Philadelphia, PA.

Hori, Chiori, Sadaoki Furui, Rob Malkin, Hua Yu, and Alex Waibel. 2002. Automatic speech summarization applied to english broadcast news speech. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing 2002*, Orlando, Florida.

Jin, Rong and Alexander G. Hauptmann. 2001a. Automatic title generation for spoken broadcast news. In *Proceedings of the Human Language Technology Conference (HLT 2001)*, San Diego, California.

Johnson, F.C., C.D. Paice, W.J. Black, and A.P. Neal. 1993. The applicaiton of linguistic processing to automatic abstract generation. *Journal of Document and Text Management*, 1(3):215–42.

Knight, Kevin and Daniel Marcu. 2000. Statistics-based summarization – step one: Sentence compression. In *The 17th National Conference of the American Association for Artificial Intelligence AAAI2000*, Austin, Texas.

Kupiec, J., J. Pedersen, and F. Chen. 1995. A trainable document summarizer. In *Proceedings of the 18th ACM-SIGIR Conference*.

Radev, Dragomir, Simone Teufel, Horacio Saggion, Wai Lam, John Blitzer, Arda Çelebi, Hong Qi, Elliott Drabek, and Dany Liu. 2002. Evaluation of text summarization in a cross-lingual information retrieval framework. Technical report, Center for Language and Speech Processing, Johns Hopkins University, Baltimore, Maryland.

Witbrock, Michael J. and Vibhu O. Mittal. 1999. Ultra-summarization: A statistical approach to generating highly condensed non-extractive summaries. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, California.

Statistical Machine Translation

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguiscits*, 19:263–311.

Brown, P.F., J. Cocke, S.A.D. Pietra, V.J.D. Pietra, F. Jelinek, J.D. Lafferty, R.L. Mercer, and P.S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

Knight, Kevin. 1997. Automating knowledge acquisition for machine translation. *AI Magazine*, 18:81–96.

Kupiec, Julian. 1993. An algorithm for finding noun phrase correspondences in bilingual corpora. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics.*

Tillmann, Christoph and Hermann Ney. 2003. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational Linguistics*, 29:97–133.

Wu, Dekai and Hongsing Wong. 1998. Machine translation with a stochastic grammatical channel. In *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics.*

Document Selection in Information Retrieval

Hearst, Marti A. 1999. User interfaces and visualization. In Ricardo Baeza-Yates and Berthier Ribeiro-Neto, editors, *Modern Information Retrieval.* Addison Wesley, New York, chapter 10.

Hersh, William, Andrew Turpin, Susan Price, Benjamin Chan, Dale Kramer, Lynetta Sacherek, and Daniel Olson. 2000. Do batch and user evaluations give the same results?

Oard, Douglas. 2001. Evaluating interactive cross-language information retrieval: Document selection. In *Proceedings of the First Cross-Language Evaluation Forum.*

Ogden, William, James Cowie, Mark Davis, Eugene Ludovik, Hugo Molina-Salgado, and Hypil Shin. 1999. Getting information from documents you cannot read: An interactive cross-language text retrieval and summarization system. In *Joint ACM DL/SIGIR Workshop on Multilingual Information Discovery and Access.*

Turpin, Andrew H. and William Hersh. 2001. Why batch and user evaluations do not give the same results. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 225–231, New Orleans, Louisiana.

Wang, Jianqiang and Douglas W. Oard. 2001. iCLEF 2001 at Maryland: Comparing Word-for-Word Gloss and MT. In *Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum, CLEF 2001*, pages 336–354, Darmstadt, Germany.

References

Bahl, L.R., F. Jelinek, and R.L. Mercer. 1983. A maximum likelihood approach to speech recognition. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, volume PAMI-5(2), pages 179–190.

Bangalore, S. and O. Rambow. 2000. Exploiting a probabilistic hierarchical model for generation. In *Proceedings of COLING 2000.*

Bikel, D., R. Schwartz, and R. Weischedel. 1999. An algorithm that learns what's in a name. *Machine Learning*, 34(1/3).

Booth, Taylor L. and Richard A. Thomson. 1973. Applying Probability Measures to Abstract Languages. In *IEEE Transactions on Computers*, volume C-22, pages 442–450.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguiscits*, 19:263–311.

Brown, P.F., J. Cocke, S.A.D. Pietra, V.J.D. Pietra, F. Jelinek, J.D. Lafferty, R.L. Mercer, and P.S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

Charniak, M. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of AAAI-97.*

Chomsky, Noam A. 1981. *Lectures on Government and Binding.* Foris Publications, Dordrecht, Holland.

Cohen, J. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measures*, 20:37–46.

Collins, M. 1997a. Three generative lexicalised models for statistical parsing. In *Proceedings of the 35th ACL.*

Collins, Michael. 1997b. The EM Algorithm (In fulfillment of the Written Preliminary Exam II requirement).

Collins, Michael. 1997c. Three Generative, Lexicalised Models for Statistical Parsing.

Conroy, John and Dianne P. O'Leary. 2001. Text summarization via hidden markov models and pivoted qr matrix decomposition. In *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, New Orleans, Louisiana.

Cutting, D., J. Pedersen, and P. Sibun. 1992. A practical part-of-speech tagger. In *Proceedings of the Third Conference on Applied Natural Language Processing*, Trento, Italy.

Daumé, H., A. Echihabi, D. Marcu, D. Munteanu, and R. Soricut. 2002. GLEANS: A Generator of Logical Extracts and Abstracts for Nice Summaires. In *Workshop on Automatic Summarization*, pages 9–14, Philadelphia, PA.

Dunning, T. 1994. Statistical identification of language. Technical Report Techincal Report MCCS 94-273, New Mexico State University.

Gale, William A. and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19:75–102.

Gale, William A., Kenneth W. Church, and David Yarowsky. 1992. A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, 26:415–439.

Gotoh, Y. and S. Reynolds. 2000. Sentence boundary detection in broadcast speech transcripts. In *Proceedings of the International Speech Communication Association Workshop: Automatic Speech Recognition: Challenges for the New Millenium*, Paris.

Hearst, Marti A. 1995. Tilebars: Visualization of term distribution information in full text information access. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pages 56–66, Denver, Colorado.

Hearst, Marti A. 1999. User interfaces and visualization. In Ricardo Baeza-Yates and Berthier Ribeiro-Neto, editors, *Modern Information Retrieval.* Addison Wesley, New York, chapter 10.

Hersh, William, Andrew Turpin, Susan Price, Benjamin Chan, Dale Kramer, Lynetta Sacherek, and Daniel Olson. 2000. Do batch and user evaluations give the same results? In *Proceedings of the 23rd annual internation ACM SIGIR conference on Research and Development in Information Retieval*, pages 17–24, Athens, Greece.

Hori, Chiori, Sadaoki Furui, Rob Malkin, Hua Yu, and Alex Waibel. 2002. Automatic speech summarization applied to english broadcast news speech. In *Proceedings of the International Conference on Acoustics Speech and Signal Processing 2002*, Orlando, Florida.

Hovy, Eduard and Chin-Yew Lin. 1999. Automated Text Summarization in SUMMARIST.

Jin, Rong and Alexander G. Hauptmann. 2001. Headline generation using a training corpus. In *Proceedings of Second International Conference on Intelligent Text Processing and Computational Linguistics*, Mexico City, Mexico.

Knight, Kevin and Daniel Marcu. 2000. Statistics-based summarization – step one: Sentence compression. In *The 17th National Conference of the American Association for Artificial Intelligence AAAI2000*, Austin, Texas.

Landauer, Thomas K., Dennis E. Egan, Joel R. Remde, Michael Lesk, Carol C. Lochbaum, and Daniel Ketchum. 1993. Enhancing the usability of text through computer delivery and formative evaluation: the superbook project. pages 71–136.

Langkilde, I. and K. Knight. 1998. Generation that exploits corpus-based statistical knowledge. In *Proceedings of COLING-ACL.*

Lin, D. 1998. Automatic retrieval and clustering of similar words. In *Proceedings of Coling/ACL.*

Luhn, H.P. 1958. The automatic creation of literature abstracts. *IBM Journal of Research Development*, 2(2):159–165.

Mann, William C. and Sandra A. Thompson. 1988. Rhetorical Structure Theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Mårdh, Ingrid. 1980. *Headlinese: On the Grammar of English Front Page Headlines.* Malmo.

Mays, E., F.J. Damerau, and R.L. Mercer. 1990. Context-based spelling correction. In *Proceedings of IBM Natural Language ITL*, pages 517–522, France.

Miller, S., M. Crystal, H. Fox, L. Ramshaw, R. Schwartz, R. Stone, and R. Weischedel. 1998. Algorithms that Learn to Extract Information; BBN: Description of the SIFT System as Used for MUC-7. In *Proceedings of the MUC-7.*

Miller, S., L. Ramshaw, H. Fox, and R. Weischedel. 2000. A novel use of statistical parsing to extract information from text. In *Proceedings of the 1st Meeting of the North Amererican Chapter of the ACL*, pages 226–233, Seattle, WA.

Oard, Douglas. 2001. Evaluating interactive cross-language information retrieval: Document selection. In *Proceedings of the First Cross-Language Evaluation Forum.*

Oard, Douglas W., Julio Gonzalo, Mark Sanderson, Fernando López-Ostenero, and Jianqiang Wang. 2003. Interactive cross-language document selection. *to appear in Information Retrieval Journal.*

Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a Method for Automatic Evaluation of Machine Translation. In *Proceedings of Association of Computational Linguistics*, Philadelphia, PA.

Pereira, F., N. Tishby, and L. Lee. 1993. Distributional clustering of english words. In *Proceedings of 31st ACL.*

Rooney, Edmund and Oliver Witte. 2000. *Copy Editing for Professionals.* Stipes Publishing Co.

Salton, Gerald. 1988. *Automatic Text Processing.* Addison-Wesley Publishing Company.

Schwartz, Richard, Sreenivasa Sista, and Timothy R. Leek. 2001. Unsupervised topic discovery. In *Proceedings of Workshop on Language Modeling and Information Retrieval*, pages 72–77, Pittsburgh, PA.

Sparck-Jones, Karen. 1972. A statisitical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21.

Turpin, Andrew H. and William Hersh. 2001. Why batch and user evaluations do not give the same results. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 225–231, New Orleans, Louisiana.

van Rijsbergen, C.J. 1979. *Information REtrieval.* Butterworths, London.

Wang, Jianqiang and Douglas W. Oard. 2001. iCLEF 2001 at Maryland: Comparing Word-for-Word Gloss and MT. In *Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum, CLEF 2001*, pages 336–354, Darmstadt, Germany.

Witbrock, Michael J. and Vibhu O. Mittal. 1999. Ultra-summarization: A statistical approach to generating highly condensed non-extractive summaries. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, Berkeley, California.

Wu, Dekai and Hongsing Wong. 1998. Machine translation with a stochastic grammatical channel. In *Proceedings of 36th Annual Meeting of the Association for Computational Linguistics*.