Headline Generation Based on Statistical Translation

Michele Banko
Computer Science Department
Johns Hopkins University
Baltimore, MD 21218
banko@cs.jhu.edu

Vibhu O. Mittal
Just Research
4616 Henry Street
Pittsburgh, PA 15213
mittal@justresearch.com

Michael J. Witbrock
Lycos Inc.
400-2 Totten Pond Road
Waltham, MA 023451
mwitbrock@lycos.com

Abstract

Extractive summarization techniques cannot generate document summaries shorter than a single sentence, something that is often required. An ideal summarization system would understand each document and generate an appropriate summary directly from the results of that understanding. A more practical approach to this problem results in the use of an approximation: viewing summarization as a problem analogous to statistical machine translation. The issue then becomes one of generating a target document in a more concise language from a source document in a more verbose language. This paper presents results on experiments using this approach, in which statistical models of the term selection and term ordering are jointly applied to produce summaries in a style learned from a training corpus.

1 Introduction

Generating effective summaries requires the ability to select, evaluate, order and aggregate items of information according to their relevance to a particular subject or for a particular purpose. Most previous work on summarization has focused on *extractive summarization*: selecting text spans - either complete sentences or paragraphs – from the original document. These extracts are then arranged in a linear order (usually the same order as in the original document) to form a summary document. There are several possible drawbacks to this approach, one of which is the focus of this paper: the inability to generate coherent summaries shorter than the smallest text-spans being considered – usually a sentence, and sometimes a paragraph. This can be a problem, because in many situations, a short headline style indicative summary is desired. Since, in many cases, the most important information in the document is scattered across multiple sentences, this is a problem for extractive summarization; worse, sentences ranked best for summary selection often tend to be even longer than the average sentence in the document.

This paper describes an alternative approach to summarization capable of generating summaries shorter than a sentence, some examples of which are given in Figure 1. It does so by building statistical models for content selection and surface realization. This paper reviews the framework, discusses some of the pros and cons of this approach using examples from our corpus of news wire stories, and presents an initial evaluation.

2 Related Work

Most previous work on summarization focused on extractive methods, investigating issues such as cue phrases (Luhn, 1958), positional indicators (Edmundson, 1964), lexical occurrence statistics (Mathis et al., 1973), probabilistic measures for token salience (Salton et al., 1997), and the use of implicit discourse structure (Marcu, 1997). Work on combining an information extraction phase followed by generation has also been reported: for instance, the FRUMP system (DeJong, 1982) used templates for both in-

Vibhu Mittal is now at Xerox PARC, 3333 Coyote Hill Road, Palo Alto, CA 94304, USA. e-mail: vmittal@parc.xerox.com; Michael Witbrock's initial work on this system was performed whilst at Just Research.

1:	time	-3.76	Beam 40
2:	new customers	-4.41	Beam 81
3:	dell computer products	-5.30	Beam 88
4:	new power macs strategy	-6.04	Beam 90
5:	apple to sell macintosh users	-8.20	Beam 86
6:	new power macs strategy on internet	-9.35	Beam 88
7:	apple to sell power macs distribution strategy	-10.32	Beam 89
8:	new power macs distribution strategy on internet products	-11.81	Beam 88
9:	apple to sell power macs distribution strategy on internet	-13.09	Beam 86

Figure 1: Sample output from the system for a variety of target summary lengths from a single input document.

formation extraction and presentation. More recently, summarizers using sophisticated post-extraction strategies, such as revision (McKeown et al., 1999; Jing and McKeown, 1999; Mani et al., 1999), and sophisticated grammar-based generation (Radev and McKeown, 1998) have also been presented.

The work reported in this paper is most closely related to work on statistical machine translation, particularly the 'IBM-style' work on CANDIDE (Brown et al., 1993). This approach was based on a statistical translation model that mapped between sets of words in a source language and sets of words in a target language, at the same time using an ordering model to constrain possible token sequences in a target language based on likelihood. In a similar vein, a summarizer can be considered to be 'translating' between two languages: one verbose and the other succinct (Berger and Lafferty, 1999; Witbrock and Mittal, 1999). However, by definition, the translation during summarization is lossy, and consequently, somewhat easier to design and experiment with. As we will discuss in this paper, we built several models of varying complexity;[1] even the simplest one did reasonably well at summarization, whereas it would have been severely deficient at (traditional) translation.

[1]We have very recently become aware of related work that builds upon more complex, structured models – syntax trees – to compress single sentences (Knight and Marcu, 2000); our work differs from that work in (i) the level of compression possible (much more) and, (ii) accuracy possible (less).

3 The System

As in any language generation task, summarization can be conceptually modeled as consisting of two major sub-tasks: (1) content selection, and (2) surface realization. Parameters for statistical models of both of these tasks were estimated from a training corpus of approximately 25,000 1997 Reuters news-wire articles on politics, technology, health, sports and business. The target documents – the summaries – that the system needed to learn the translation mapping to, were the headlines accompanying the news stories.

The documents were preprocessed before training: formatting and mark-up information, such as font changes and SGML/HTML tags, was removed; punctuation, except apostrophes, was also removed. Apart from these two steps, no other normalization was performed. It is likely that further processing, such as lemmatization, might be useful, producing smaller and better language models, but this was not evaluated for this paper.

3.1 Content Selection

Content selection requires that the system learn a model of the relationship between the appearance of some features in a document and the appearance of corresponding features in the summary. This can be modeled by estimating the likelihood of some token appearing in a summary given that some tokens (one or more, possibly different tokens) appeared in the document to be summarized. The very simplest, "zero-level" model for this relationship is the case when the two tokens

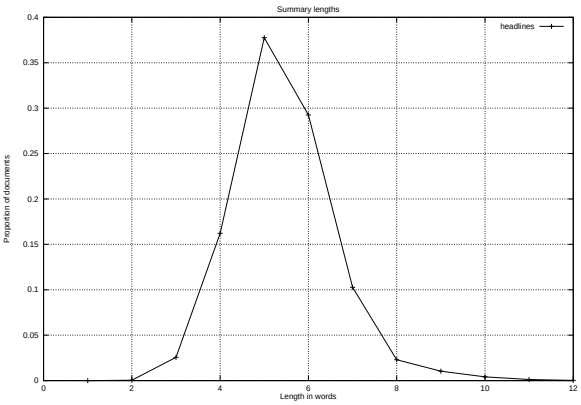

Figure 2: Distribution of Headline Lengths for early 1997 Reuters News Stories.

in the document and the summary are identical. This can be computed as the conditional probability of a word occurring in the summary given that the word appeared in the document:

$$P(w_i \in H \mid w_i \in D) =$$

$$\frac{P(w_i \in D \mid w_i \in H) \cdot P(w_i \in H)}{P(w_i \in D)}$$

where H and D represent the bags of words that the headline and the document contain. Once the parameters of a content selection model have been estimated from a suitable document/summary corpus, the model can be used to compute selection scores for candidate summary terms, given the terms occurring in a particular source document. Specific subsets of terms, representing the core summary content of an article, can then be compared for suitability in generating a summary. This can be done at two levels (1) likelihood of the length of resulting summaries, given the source document, and (2) likelihood of forming a coherently ordered summary from the content selected.

The length of the summary can also be learned as a function of the source document. The simplest model for document length is a fixed length based on document genre. For the discussions in this paper, this will be the model chosen. Figure 2 shows the distribution of headline length. As can be seen, a Gaussian distribution could also model the likely lengths quite accurately.

Finally, to simplify parameter estimation for the content selection model, we can assume that the likelihood of a word in the summary is independent of other words in the summary. In this case, the probability of any particular summary-content candidate can be calculated simply as the product of the probabilities of the terms in the candidate set. Therefore, the overall probability of a candidate summary, H, consisting of words $(w_1, w_2, \ldots, w_n)$, under the simplest, zero-level, summary model based on the previous assumptions, can be computed as the product of the likelihood of (i) the terms selected for the summary, (ii) the length of the resulting summary, and (iii) the most likely sequencing of the terms in the content set.

$$\begin{aligned} P(w_1, \ldots, w_n) &= \prod_{i=1}^{n} P(w_i \in H \mid w_i \in D) \\ &\quad \cdot P(len(H) = n) \\ &\quad \cdot \prod_{i=2}^{n} P(w_i \mid w_1, \ldots, w_{i-1}) \end{aligned}$$

In general, the probability of a word appearing in a summary cannot be considered to be independent of the structure of the summary, but the independence assumption is an initial modeling choice.

3.2 Surface Realization

The probability of any particular surface ordering as a headline candidate can be computed by modeling the probability of word sequences. The simplest model is a bigram language model, where the probability of a word sequence is approximated by the product of the probabilities of seeing each term given its immediate left context. Probabilities for sequences that have not been seen in the training data are estimated using back-off weights (Katz, 1987). As mentioned earlier, in principle, surface linearization calculations can be carried out with respect to any textual spans from characters on up, and could take into account additional information at the phrase level. They could also, of course, be extended to use higher order n-grams, providing that sufficient numbers of training headlines were available to estimate the probabilities.

3.3 Search

Even though content selection and summary structure generation have been presented separately, there is no reason for them to occur independently, and in fact, in our current implementation, they are used simultaneously to contribute to an overall weighting scheme that ranks possible summary candidates against each other. Thus, the overall score used in ranking can be obtained as a weighted combination of the content and structure model log probabilities. Cross-validation is used to learn weights α, β and γ for a particular document genre.

$$\begin{aligned}\arg\max_{H}(\quad & \alpha \cdot \sum_{i=1}^{n} log(P(w_i \in H \mid w_i \in D)) + \\ & \beta \cdot log(P(len(H) = n)) + \\ & \gamma \cdot \sum_{i=2}^{n} log(P(w_i \mid w_{i-1})))\end{aligned}$$

To generate a summary, it is necessary to find a sequence of words that maximizes the probability, under the content selection and summary structure models, that it was generated from the document to be summarized. In the simplest, zero-level model that we have discussed, since each summary term is selected independently, and the summary structure model is first order Markov, it is possible to use Viterbi beam search (Forney, 1973) to efficiently find a near-optimal summary. [2] Other statistical models might require the use of a different heuristic search algorithm. An example of the results of a search for candidates of various lengths is shown in Figure 1. It shows the set of headlines generated by the system when run against a real news story discussing Apple Computer's decision to start direct internet sales and comparing it to the strategy of other computer makers.

[2] In the experiments discussed in the following section, a beam width of three, and a minimum beam size of twenty states was used. In other experiments, we also tried to strongly discourage paths that repeated terms, by reweighting after backtracking at every state, since, otherwise, bigrams that start repeating often seem to pathologically overwhelm the search; this reweighting violates the first order Markovian assumptions, but seems to to more good than harm.

4 Experiments

Zero level–Model: The system was trained on approximately 25,000 news articles from Reuters dated between 1/Jan/1997 and 1/Jun/1997. After punctuation had been stripped, these contained about 44,000 unique tokens in the articles and slightly more than 15,000 tokens in the headlines. Representing all the pairwise conditional probabilities for all combinations of article and headline words[3] added significant complexity, so we simplified our model further and investigated the effectiveness of training on a more limited vocabulary: the set of all the words that appeared in any of the headlines.[4] Conditional probabilities for words in the headlines that also appeared in the articles were computed. As discussed earlier, in our zero-level model, the system was also trained on bigram transition probabilities as an approximation to the headline syntax. Sample output from the system using this simplified model is shown in Figures 1 and 3.

Zero Level–Performance Evaluation: The zero-level model, that we have discussed so far, works surprisingly well, given its strong independence assumptions and very limited vocabulary. There are problems, some of which are most likely due to lack of sufficient training data.[5] Ideally, we should want to evaluate the system's performance in terms both of content selection success and realization quality. However, it is hard to computationally evaluate coherence and phrasing effectiveness, so we have, to date, restricted ourselves to the content aspect, which is more amenable to a quantitative analysis. (We have experience doing much more laborious human eval-

[3] This requires a matrix with 660 million entries, or about 2.6GB of memory. This requirement can be significantly reduced by using a threshold to prune values and using a sparse matrix representation for the remaining pairs. However, inertia and the easy availability of the CMU-Cambridge Statistical Modeling Toolkit – which generates the full matrix – have so far conspired to prevent us from exercising that option.

[4] An alternative approach to limiting the size of the mappings that need to be estimated would be to use only the top n words, where n could have a small value in the hundreds, rather than the thousands, together with the words appearing in the headlines. This would limit the size of the model while still allowing more flexible content selection.

[5] We estimate that approximately 100MB of training data would give us reasonable estimates for the models that we would like to evaluate; we had access to much less.

<HEADLINE> **U.S. Pushes for Mideast Peace** </HEADLINE>

President Clinton met with his top Mideast advisers, including Secretary of State Madeleine Albright and U.S. peace envoy Dennis Ross, in preparation for a session with Israel Prime Minister Benjamin Netanyahu tomorrow. Palestinian leader Yasser Arafat is to meet with Clinton later this week. Published reports in Israel say Netanyahu will warn Clinton that Israel can't withdraw from more than nine percent of the West Bank in its next scheduled pullback, although Clinton wants a 12-15 percent pullback.

1:	clinton	-6	0
2:	clinton wants	-15	2
3:	clinton netanyahu arafat	-21	24
4:	clinton to mideast peace	-28	98
5:	clinton to meet netanyahu arafat	-33	298
6:	clinton to meet netanyahu arafat israel	-40	1291

Figure 3: Sample article (with original headline) and system generated output using the simplest, zero-level, lexical model. Numbers to the right are log probabilities of the string, and search beam size, respectively.

uation, and plan to do so with our statistical approach as well, once the model is producing summaries that might be competitive with alternative approaches.)

After training, the system was evaluated on a separate, previously unseen set of 1000 Reuters news stories, distributed evenly amongst the same topics found in the training set. For each of these stories, headlines were generated for a variety of lengths and compared against the (i) the actual headlines, as well as (ii) the sentence ranked as the most important summary sentence. The latter is interesting because it helps suggest the degree to which headlines used a different vocabulary from that used in the story itself.[6] Term overlap between the generated headlines and the test standards (both the actual headline and the summary sentence) was the metric of performance.

Gen. Headline Length (words)	Word Overlap	Percentage of complete matches
4	0.2140	19.71%
5	0.2027	14.10%
6	0.2080	12.14%
7	0.1754	08.70%
8	0.1244	11.90%

Table 1: Evaluating the use of the simplest lexical model for content selection on 1000 Reuters news articles. The headline length given is that a which the overlap between the terms in the target headline and the generated summary was maximized. The percentage of complete matches indicates how many of the summaries of a given length had all their terms included in the target headline.

For each news article, the maximum overlap between the actual headline and the generated headline was noted; the length at which this overlap was maximal was also taken into account. Also tallied were counts of headlines that matched completely – that is, all of the words in the generated headline were present in the actual headline – as well as their lengths. These statistics illustrate the system's performance in selecting content words for the headlines. Actual headlines are often, also, ungrammatical, incomplete phrases. It is likely that more sophisticated language models, such as structure models (Chelba, 1997; Chelba and Jelinek, 1998), or longer n-gram models would lead to the system generating headlines that were more similar in phrasing to real headlines because longer range dependencies

[6]The summarizer we used here to test was an off-the-shelf Carnegie Mellon University summarizer, which was the top ranked extraction based summarizer for news stories at the 1998 DARPA-TIPSTER evaluation workshop (Tip, 1998). This summarizer uses a weighted combination of sentence position, lexical features and simple syntactical measures such as sentence length to rank sentences. The use of this summarizer should not be taken as a indicator of its value as a testing standard; it has more to do with the ease of use and the fact that it was a reasonable candidate.

	Overlap with headline				Overlap with summary			
L	**Lex**	**+Position**	**+POS**	**+Position+POS**	**Lex**	**+Position**	**+POS**	**+Position+POS**
1	0.37414	0.39888	0.30522	0.40538	0.61589	0.70787	0.64919	0.67741
2	0.24818	0.26923	0.27246	0.27838	0.57447	0.63905	0.57831	0.63315
3	0.21831	0.24612	0.20388	0.25048	0.55251	0.63760	0.55610	0.62726
4	0.21404	0.24011	0.18721	0.25741	0.56167	0.65819	0.52982	0.61099
5	0.20272	0.21685	0.18447	0.21947	0.55099	0.63371	0.53578	0.58584
6	0.20804	0.19886	0.17593	0.21168	0.55817	0.60511	0.51466	0.58802

Table 2: Overlap between terms in the generated headlines and in the original headlines and extracted summary sentences, respectively, of the article. Using Part of Speech (POS) and information about a token's location in the source document, in addition to the lexical information, helps improve performance on the Reuters' test set.

could be taken into account. Table 1 shows the results of these term selection schemes. As can be seen, even with such an impoverished language model, the system does quite well: when the generated headlines are four words long almost one in every five has all of its words matched in the article s actual headline. This percentage drops, as is to be expected, as headlines get longer.

Multiple Selection Models: POS and Position As we mentioned earlier, the zero-level model that we have discussed so far can be extended to take into account additional information both for the content selection and for the surface realization strategy. We will briefly discuss the use of two additional sources of information: (i) part of speech (POS) information, and (ii) positional information.

POS information can be used both in content selection – to learn which word-senses are more likely to be part of a headline – and in surface realization. Training a POS model for both these tasks requires far less data than training a lexical model, since the number of POS tags is much smaller. We used a mixture model (McLachlan and Basford, 1988) – combining the lexical and the POS probabilities – for both the content selection and the linearization tasks.

Another indicator of salience is positional information, which has often been cited as one of the most important cues for summarization by ex-

1:	clinton	-23.27
2:	clinton wants	-52.44
3:	clinton in albright	-76.20
4:	clinton to meet albright	-105.5
5:	clinton in israel for albright	-129.9
6:	clinton in israel to meet albright	-158.57

(a) System generated output using a lexical + POS model.

1:	clinton	-3.71
2:	clinton mideast	-12.53
3:	clinton netanyahu arafat	-17.66
4:	clinton netanyahu arafat israel	-23.1
5:	clinton to meet netanyahu arafat	-28.8
6:	clinton to meet netanyahu arafat israel	-34.38

(b) System generated output using a lexical + positional model.

1:	clinton	-21.66
2:	clinton wants	-51.12
3:	clinton in israel	- 58.13
4:	clinton meet with israel	-78.47
5:	clinton to meet with israel	-87.08
6:	clinton to meet with netanyahu arafat	-107.44

(c) System generated output using a lexical + POS + positional model.

Figure 4: Output generated by the system using augmented lexical models. Numbers to the right are log probabilities of the generated strings under the generation model.

Original term	Generated term	Original headline	Generated headline
Nations Top Judge	Rehnquist	Wall Street Stocks Decline	Dow Jones index lower
Kaczynski	Unabomber Suspect	49ers Roll Over Vikings 38-22	49ers to nfc title game
ER	Top-Rated Hospital Drama	Corn, Wheat Prices Fall	soybean grain prices lower
Drugs	Cocaine	Many Hopeful on N. Ireland Accord	britain ireland hopeful of irish peace

Table 3: Some pairs of target headline and generated summary terms that were counted as errors by the evaluation, but which are semantically equivalent, together with some "equally good" generated headlines that were counted as wrong in the evaluation.

traction (Hovy and Lin, 1997; Mittal et al., 1999). We trained a content selection model based on the position of the tokens in the training set in their respective documents. There are several models of positional salience that have been proposed for sentence selection; we used the simplest possible one: estimating the probability of a token appearing in the headline given that it appeared in the 1st, 2nd, 3rd or 4th quartile of the body of the article. We then tested mixtures of the lexical and POS models, lexical and positional models, and all three models combined together. Sample output for the article in Figure 3, using both lexical and POS/positional information can be seen in Figure 4. As can be seen in Table 2,[7] Although adding the POS information alone does not seem to provide any benefit, positional information does. When used in combination, each of the additional information sources seems to improve the overall model of summary generation.

Problems with evaluation: Some of the statistics that we presented in the previous discussion suggest that this relatively simple statistical summarization system is not very good compared to some of the extraction based summarization systems that have been presented elsewhere (e.g., (Radev and Mani, 1997)). However, it is worth emphasizing that many of the headlines generated by the system were quite good, but were penalized because our evaluation metric was based on the word-error rate and the generated headline terms did not exactly match the original ones. A quick manual scan of some of the failures that might have been scored as successes in a subjective manual evaluation indicated that some of these errors could not have been avoided without adding knowledge to the system, for example, allowing the use of alternate terms for referring to collective nouns. Some of these errors are shown in Table 3.

5 Conclusions and Future Work

This paper has presented an alternative to extractive summarization: an approach that makes it possible to generate coherent summaries that are shorter than a single sentence and that attempt to conform to a particular style. Our approach applies statistical models of the term selection and term ordering processes to produce short summaries, shorter than those reported previously. Furthermore, with a slight generalization of the system described here, the summaries need not contain any of the words in the original document, unlike previous statistical summarization systems. Given good training corpora, this approach can also be used to generate headlines from a variety of formats: in one case, we experimented with corpora that contained Japanese documents and English headlines. This resulted in a working system that could simultaneously translate and summarize Japanese documents.[8]

The performance of the system could be improved by improving either content selection or linearization. This can be through the use of more sophisticated models, such as additional language models that take into account the signed distance between words in the original story to condition

[7] Unlike the data in Table 1, these headlines contain only six words or fewer.

[8] Since our initial corpus was constructed by running a simple lexical translation system over Japanese headlines, the results were poor, but we have high hopes that usable summaries may be produced by training over larger corpora.

the probability that they should appear separated by some distance in the headline.

Recently, we have extended the model to generate multi-sentential summaries as well: for instance, given an initial sentence such as "*Clinton to meet visit MidEast.*" and words that are related to nouns ("Clinton" and "mideast") in the first sentence, the system biases the content selection model to select other nouns that have high mutual information with these nouns. In the example sentence, this generated the subsequent sentence "*US urges Israel plan.*" This model currently has several problems that we are attempting to address: for instance, the fact that the words co-occur in adjacent sentences in the training set is not sufficient to build coherent adjacent sentences (problems with pronominal references, cue phrases, sequence, etc. abound). Furthermore, our initial experiments have suffered from a lack of good training and testing corpora; few of the news stories we have in our corpora contain multi-sentential headlines.

While the results so far can only be seen as indicative, this breed of non-extractive summarization holds a great deal of promise, both because of its potential to integrate many types of information about source documents and intended summaries, and because of its potential to produce very brief coherent summaries. We expect to improve both the quality and scope of the summaries produced in future work.

References

Adam Berger and John Lafferty. 1999. Information retrieval as statistical translation. In *Proc. of the 22nd ACM SIGIR Conference (SIGIR-99)*, Berkeley, CA.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, (2):263–312.

Ciprian Chelba and F. Jelinek. 1998. Exploiting syntactic structure for language modeling. In *Proc. of ACL-98*, Montreal, Canada. ACL.

Ciprian Chelba. 1997. A structured language model. In *Proc. of the ACL-97*, Madrid, Spain. ACL.

Gerald F. DeJong. 1982. An overview of the FRUMP system. In Wendy G. Lehnert and Martin H. Ringle, editors, *Strategies for Natural Language Processing*, pages 149–176. Lawrence Erlbaum Associates, Hillsdale, NJ.

H. P. Edmundson. 1964. Problems in automatic extracting. *Communications of the ACM*, 7:259–263.

G. D. Forney. 1973. The Viterbi Algorithm. *Proc. of the IEEE*, pages 268–278.

Eduard Hovy and Chin Yew Lin. 1997. Automated text summarization in SUMMARIST. In *Proc. of the Wkshp on Intelligent Scalable Text Summarization, ACL-97*.

Hongyan Jing and Kathleen McKeown. 1999. The decomposition of human-written summary sentences. In *Proc. of the 22nd ACM SIGIR Conference*, Berkeley, CA.

S. Katz. 1987. Estimation of probabilities from sparse data for the language model component of a speech recognizer. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 24.

Kevin Knight and Daniel Marcu. 2000. Statistics-based summarization — step one: Sentence compression. In *Proc. of AAAI-2000*, Austin, TX.

P. H. Luhn. 1958. Automatic creation of literature abstracts. *IBM Journal*, pages 159–165.

Inderjeet Mani, Barbara Gates, and Eric Bloedorn. 1999. Improving summaries by revising them. In *Proc. of ACL-99*, Baltimore, MD.

Daniel Marcu. 1997. From discourse structures to text summaries. In *Proc. of the ACL'97 Wkshp on Intelligent Text Summarization*, pages 82–88, Spain.

B. A. Mathis, J. E. Rush, and C. E. Young. 1973. Improvement of automatic abstracts by the use of structural analysis. *JASIS*, 24:101–109.

Kathleen R. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. 1999. Towards Multidocument Summarization by Reformulation: Progress and Prospects. In *Proc. of AAAI-99*. AAAI.

G.J. McLachlan and K. E. Basford. 1988. *Mixture Models*. Marcel Dekker, New York, NY.

Vibhu O. Mittal, Mark Kantrowitz, Jade Goldstein, and Jaime Carbonell. 1999. Selecting Text Spans for Document Summaries: Heuristics and Metrics. In *Proc. of AAAI-99*, pages 467–473, Orlando, FL, July. AAAI.

Dragomir Radev and Inderjeet Mani, editors. 1997. *Proc. of the Workshop on Intelligent Scalable Text Summarization, ACL/EACL-97 (Madrid)*. ACL, Madrid, Spain.

Dragomir Radev and Kathy McKeown. 1998. Generating natural language summaries from multiple online sources. *Compuutational Linguistics*.

Gerard Salton, A. Singhal, M. Mitra, and C. Buckley. 1997. Automatic text structuring and summary. *Info. Proc. and Management*, 33(2):193–207, March.

1998. Tipster text phase III 18-month workshop notes, May. Fairfax, VA.

Michael Witbrock and Vibhu O. Mittal. 1999. Headline generation: A framework for generating highly-condensed non-extractive summaries. In *Proc. of the 22nd ACM SIGIR Conference (SIGIR-99)*, pages 315–316, Berkeley, CA.