Self-Supervised Learning for Automatic Text Summarization by Text-span Extraction

Massih-Reza Amini, Patrick Gallinari

LIP6, University of Paris 6
Case 169, 4 Place Jussieu, F – 75252
Paris cedex 05, France.
{amini, gallinari}@poleia.lip6.fr

Abstract

We describe a system for automatic text summarization that operates by extracting the most relevant sentences from documents with regard to a query. The lack of labeled corpora makes it difficult to develop automatic techniques for summarization. We propose to use a self-supervised method which does not rely on the availability of labeled corpora for learning to rank sentences for the summary. The method operates in two steps: first a statistical similarity based system which does not require any training is developed, second a classifier is trained using self-supervised learning in order to improve this baseline method. This idea is evaluated on the Reuters news-wire corpus and compared to other strategies.

1 Introduction

With the increase of textual information, summarizing document is becoming an important issue. Text summaries allow users to rapidly consult retrieved documents and decide on their relevance.

Automated summarization dates back to the fifties [2]. The different attempts in this field have shown that human-quality text summarization was very complex since it encompasses discourse understanding, abstraction, and language generation [31]. Simpler approaches were explored which consist in extracting representative text-spans, using statistical techniques and/or techniques based on superficial domain-independent linguistic analyses. For these approaches, summarization can be defined as the selection of a subset of the document sentences which is representative of its content. This is typically done by ranking the document sentences and selecting those with higher score and with a minimum overlap. Most of the recent work in summarization uses this paradigm. Usually, sentences are used as text-span units but paragraphs have also been considered [25, 32]. The latter may sometimes appear more appealing since they contain more contextual information. Extraction based text summarization techniques can operate in two modes: generic summarization, which consists in abstracting the main ideas of a whole document and query-based summarization, which aims at abstracting the information relevant for a given query.

Our work takes the text-span extraction paradigm. It explores the use of self-supervised learning techniques for improving automatic summarization methods. The proposed model could be used both for generic and query-based summaries. However for evaluation purposes we present results on a generic summarization task. Previous work on the application of machine learning techniques for summarization [17, 18, 22, 24, 33] rely on the supervised learning paradigm. Such approaches usually need a training set of documents and associated summaries, which is used to label the document sentences as relevant or non-relevant for the summary. After training, these systems operate on unlabeled text by ranking the sentences of a new document according to their relevance for the summarization task. Labeling the large amount of documents, needed for supervised training, is a lengthy process, and labeling or ranking text spans is intrinsically difficult since there is not a simple characterization of what a good summary is. We explore two approaches for making the training of machine learning systems easier for this task, one is semi-supervised learning where the systems is trained using a small amount of labeled data together with a large set of unlabeled documents. The second approach is based on self-learning and goes further, since it does not require any training corpus of documents and associated summaries, nor labeled texts.

The paper is organized as follows, we first make a review of recent work in text summarization and briefly introduce the semi-supervised and self-learning paradigms (section 2). We then describe our approach to text summarization based on sentence segment extraction (section 3), and finally we present a series of experiments (section 4 and section 5).

2 Related Work

Several innovative methods for automated document summarization have been explored over the last years, they exploit either statistical approaches [16, 19, 25, 35] or linguistic approaches [12, 20, 23, 30], and combinations of the two [13, 18, 32]. We will focus here on a statistical approach to the problem and more precisely on the use of machine learning techniques.

2.1 Machine Learning-based Text Summarization

Some authors have proposed to use machine learning for improving summarization systems. [22] and [33] consider the problem of sentence extraction as a classification task. [22] propose a generic summarization model, which is based on a Naïve-Bayes classifier: each sentence is classified as relevant or non-relevant for the summary and those with highest score are selected. His system uses five features: an indication of whether or not the sentence length is below a specified threshold, occurrence of cue words, position of the sentence in the text and in the paragraph, occurrence of frequent words, and occurrence of words in capital letters, excluding common abbreviations.

[24] has used several machine learning techniques in order to discover features indicating the salience of a sentence. He addressed the production of generic and user-focused summaries. Features were divided into three groups: locational, thematic and cohesion features. The document database was CMP-LG also used in [10], which contains human summaries provided by the text author. The extractive summaries required for training were automatically generated as follows: the relevance of each document sentence with respect to the human summary is computed, highest score sentences are retained, for building the extractive summary. This model can be considered both as a generic and a query-based text summarizer.

We already described a query-relevant text summarization system based on interactive learning [11]. The system proceeds in two steps, it first extracts the most relevant sentences of a document with regard to a user query it then learns user feedback in order to improve its performances. Learning operates at two levels: query expansion and sentence scoring. This work was focused on user interaction whereas the present paper deals with automatic summarization.

[17] present an algorithm which generates a summary by extracting sentence segments in order to increase the summary concision. Each segment is represented by a set of predefined features such as its location, the average term frequencies of words occurring in the segment, the number of title words in the segment. Then they compare three supervised learning algorithms: C4.5, Naïve-Bayes and neural networks. Their conclusion is that all three methods successfully completed the task by generating reasonable summaries.

2.3 Learning with Labeled-Unlabeled Data

Labeling large text collections at the sentence level for summarization is required for training classifiers to discriminate between relevant and non-relevant sentences is very costly and non realistic. This is true for many other applications as well. On the other hand, gathering large quantities of unlabeled data is usually cheap and people in different fields such as signal processing, statistics and more recently machine learning have tried to use unlabeled data - sometimes together with small amounts of labeled data - in order to train classifiers. We have explored two such directions: self-supervised and semi-supervised learning.

2.3.1 Self-supervised learning

Self-supervised learning can be considered as a particular case of unsupervised learning. As for the latter, data are unlabeled, however the goal here is to classify data and not to cluster them or to estimate their density as it is usually the case for unsupervised learning. Available a priori information is used in order to design a baseline classifier. Then, training proceeds repeatedly by using the decisions of the classifier at step s, for labeling the examples, and then training the classifier at step s+1 to learn these labels, and so on. This is also called the *decision-directed* approach to unsupervised leaning. It can be applied sequentially by updating the classifier each time an unlabeled sample is classified [4, 5], this is the classical approach used in adaptive signal processing. Alternatively, as in our case, it can be applied in parallel by waiting until all examples are classified before updating the classifier. This process can be repeated until no change occurs in labels. It can be shown that this process converges and in many cases it has proceed to perform well.

2.3.2 Semi-supervised learning

Different attempts have been performed in order to exploit the availability of large corpus of unlabeled data for improving the accuracy of classification algorithms trained already with a small amount of labeled data. The machine learning community has recently re-discovered this paradigm and this research direction is becoming popular.

One approach is to use variations of the E-M algorithm [1]. A classifier is first trained using the available labeled data. EM steps are then iterated which alternate between the classification of unlabeled data by the current classifier and the estimation of a new classifier parameters for learning these labels, until convergence.

This is for example the approach pursued in [27, 28], they train a Naïve-Bayes classifier using EM, and discuss the role and importance of unlabeled data for semi-supervised training. Their experimental results obtained using text from three corpuses, show that the use of unlabeled data allows to reduce the classification error up to 33%.

A second approach is the idea of co-training [14, 29]. It is supposed here that data may be described from two different points of view. For example, web pages can be described by either plain text from the web page, or by hyperlinks text. Co-training consists in training a classifier from each data representation by alternating two steps until convergence: in the first step, the outputs of classifier *A* which operates on one representation, are used as labels for training classifier *B* which takes as input the other representation. In the second step, the roles of *A* and *B* are reversed. We will not use this idea here since the problem here may be more naturally handled using either self-learning or the E-M version of semi-supervised learning.

3 Design of a Text Summarizer based on Sentence Segment Extraction and Self-supervised learning

The system we propose proceeds in two steps, it first extracts the most relevant sentences of a document with regard to a user query using a classical *tf-idf* term weighting scheme. This allows computing an initial classification of the document sentences into relevant and non-relevant sentences for the summary. The classification scores also provide a ranking of the sentences. It then learns using self-supervised learning in order to improve this classification and ranking. For generic summarization, a query vector is calculated using high frequency document words. We describe below the first step.

3.1 Sentence extraction by using similarity measures

Many systems for sentence extraction have been proposed which use similarity measures between text spans (sentences or paragraphs) and queries, e.g. [18, 25]. Representative sentences are then selected by comparing the sentence score for a given document to a preset threshold. The main difference between these systems is the representation of textual information and the similarity measures they are using. Usually, statistical and/or linguistic characteristics are used in order to encode the text (sentences and queries) into a fixed size vector and simple similarities (e.g. cosine) are then computed.

We will build here on the work of [21] who used such a technique for the extraction of sentences relevant to a given query. They use a *tf-idf* representation and compute the similarity between sentence s_k and query q as:

$$Sim(q, s_k) = \sum_i tf(w_i, q).tf(w_i, s_k).\left(1 - \frac{\log(df(w_i)+1)}{\log(n+1)}\right)^2 \quad (1)$$

Where, *tf(w,x)* is the frequency of term w in x (q or s_k), *df(w)* is the document frequency of term w and n is the total number of documents in the collection. Sentence s_k and query q are pre-processed by removing stop-words and performing Porter-reduction on the remaining words. For each document a threshold is then estimated from data for selecting the most relevant sentences.

Our approach for the sentence extraction step is a variation of the above method where the query is enriched before computing the similarity. Since queries and sentences may be very short, this allows computing more meaningful similarities. Query expansion - via user feedback or via pseudo relevance feedback - has been successfully used for years in Information Retrieval (IR) e.g. [3, 34]. The query expansion proceeds in two steps: first the query is expanded via a similarity thesaurus - WordNet in our experiments -, second, relevant sentences are extracted from the document and the most frequent words in these sentences are included into the query. This process can be iterated. The similarity we consider is then:

$$Sim(q, s_k) = \sum_{w_i \in s_k, q} \bar{tf}(w_i, q).tf(w_i, s_k).\left(1 - \frac{\log(df(w_i)+1)}{\log(n+1)}\right) \quad (2)$$

Where, $\bar{tf}(w,q)$ is the number of terms within the "semantic" class of w_i in the query q.

This extraction system will be used as a baseline system for evaluating the impact of learning throughout the paper. Although it is basic, similar systems have been shown to perform well for sentence extraction based text summarization. For example [35] uses such an approach, which operates only on word frequencies for sentence extraction in the context of generic summaries, and shows that it compares well with human based sentence extraction.

3.2 Learning

Methods based on similarity measures do have intrinsic limitations: they rely on simple predefined features and measures, they are developed for generic documents, their adaptation to a specific corpus or to different document genres has to be manually settled. Machine learning allows to better exploit the corpus characteristics and to improve the qualities or the adaptability of summarization systems. This is particularly important for example in an Internet context since document types may vary considerably.

We propose below a technique, which takes into account the coherence of the whole set of relevant sentences for the summaries and allows to significantly increasing the quality of extracted sentences.

3.2.1 Features

We define new features in order to train our system for sentence classification. A sentence is considered as a sequence of terms, each of them being characterized by a set of features. The sentence representation will then be the corresponding sequence of these features.

We used four values for characterizing each term w of sentence s: $tf(w,s)$, $\bar{tf}(w,q)$, (1-(log(df(w)+1)/log(n+1)) and $Sim(q,s)$ -computed as in (2)- the similarity between q and s. The first three variables are frequency statistics which give the importance of a term for characterizing respectively the sentence, the query and the document. The last one gives the importance of the sentence containing w for the summary and is used in place of the term importance since it is difficult to provide a meaningful measure for isolated terms [21].

A first labeling of the sentences as relevant or irrelevant is provided by the baseline system. By tuning a threshold over the similarity measures of sentences for a given document, sentences having higher similarity measures than this threshold were set to be relevant. We then use self-supervised learning to train a classifier upon the sentence labels provided by the previous classifier and repeat the process until no change occurs in the labels.

3.3.2 Classifier

We used two linear classifiers, a one layer perceptron with a sigmoid activation function [6] and a Support Vector Machine (SVM) [15], to compute $P(R_Q/s)$, the posterior probability of relevance for the query given a sentence, using these training sets.

3.3.3 Semi-supervised learning

We use the same word representation as in the case of self-supervised learning. We have labeled 10% of the sentences in the training set using the news-wire summaries as the correct set of sentences. We then train our classifiers in a first step using these labels. Training proceeds after that in the same way as for the self supervised case: this first classifier is used to label all the sentences from the training set, these labels are used for the next step using unlabeled data and so on until convergence.

4 Database

A corpus of documents with the corresponding summaries is required for the evaluation. Note that as already said such a corpus is not necessary for implementing the self supervised system. We have used the Reuters data set consisting of news-wire summaries [7]: this corpus is composed of 1000 documents and their associated extracted sentence summaries. The data set was split into a training and a test set. Since the evaluation is performed for a

generic summarization task, a query was generated by collecting the most frequent words in the training set. Statistics about the data set collection and summaries are shown in table 1.

Reuters data set			
Collection	**Training**	**Test**	**All**
# of docs	300	700	1000
Average # of sentences/doc	26.18	22.29	23.46
Min sentence/doc	7	5	5
Max sentence/doc	87	88	88
News-wire summaries			
Average # of sentences /sum	4.94	4.01	4.3
% of summaries including 1st sentence of docs	63.3	73.5	70.6

Table 1. Characteristics of the Reuters data set and of the corresponding summaries.

Figure 1-up shows the histogram of summary lengths in sentences, it is narrowly distributed around 5 sentences and Figure 1-bottom shows the summary length in words which is approximately a normal distribution with a peak around 80 words.

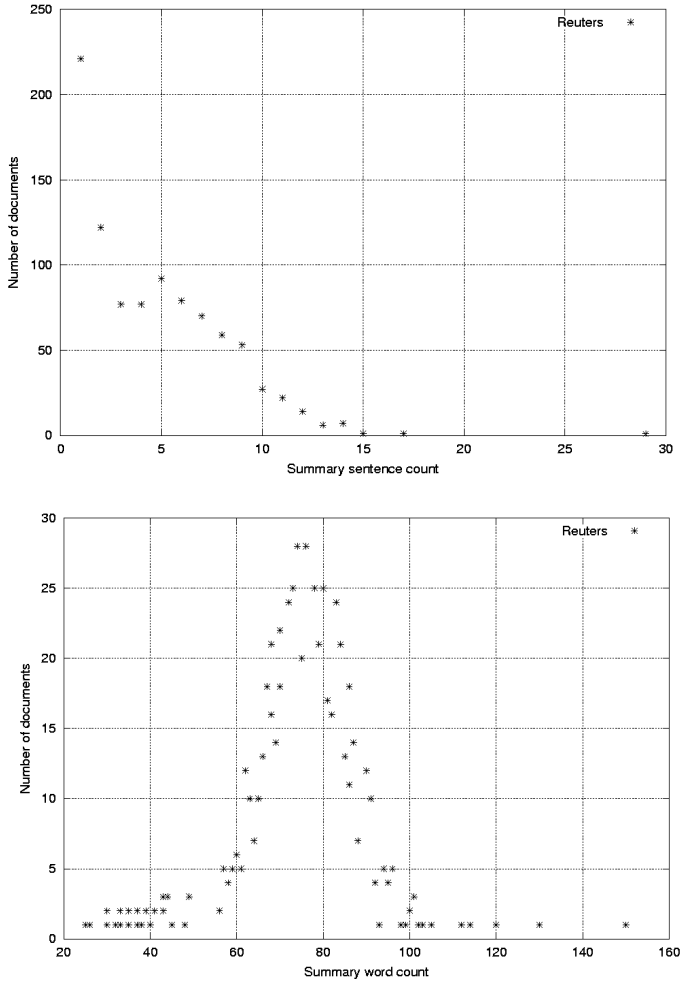

Figure 1: Up: Distribution of summary length in sentences, Bottom: Distribution of summary length in words

5 Evaluation

Evaluation issues of summarization systems have been the object of several attempts, many of them being carried within the tipster program [8] and the Summac competition [9]. This is a complex issue and many different aspects have to be considered simultaneously in order to evaluate and compare different summarizers [26].

Our methods provide a set of relevant document sentences. Taking all the selected sentences, we can build an *extract* for the document. For the evaluation, we compared this extract with the news-wire summary and used Precision and Recall measures, defined as follows:

$$\text{Precision} = \frac{\#\text{ of sentences extracted by the system which are in the news - wire summaries}}{\text{total \# of sentences extracted by the system}}$$

$$\text{Recall} = \frac{\#\text{ of sentences extracted by the system which are in the news - wire summaries}}{\text{total \# of sentences in the news - wire summaries}}$$

We give below the average precision (table 2) for the different systems and the precision/recall curves (figure 2). The baseline system gives bottom line performances, which allow evaluating the contribution of our training strategies. In order to provide an upper bound of the expected performances, we have also trained a classifier in a fully supervised way, by labeling all the training set sentences using the news-wire summaries.

	Precision (%)	**Total Average (%)**
Baseline system	54,94	56,33
Supervised learning	72,68	74,06
Semi-Supervised learning	63,94	65,32
Self-supervised learning	63,53	64,92

Table 2. Comparison between the baseline system and different learning schemes, using linear neural networks as classifier. Performances are on the test set.

Semi-supervised and self-supervised provide a clear increase of performances (up to 9 %). If we compare these results to fully supervised learning that is 9% better, these methods are able to extract from the unlabeled data half of the information needed for this "optimal" classification.

We have also compared the linear Neural Network model to a linear SVM model in the case of self-supervised learning as shown at Table 3. The two models performed similarly, both are linear classifiers although their training criterion is slightly different.

	Precision (%)	**Total Average (%)**
Self-Supervised learning with Neural-Networks	63,53	64,92
Self-Supervised learning with SVM	62,15	63,55

Table 3. Comparison between two different linear models: Neural Networks and SVM in the case of Self-supervised learning. Performances are on the test set.

11-point precision recall curves allow a more precise evaluation of the system behavior. Let M be the total number of sentences extracted by the system as relevant (correctly or incorrectly), N_s the total number of sentences extracted by the system which are in the newswire summaries, N_g the total number of sentences in newswire summaries and N the total number of sentences in the test set.

Precision and recall are computed respectively as N_s/M and N_s/N_g. For a given document, sentence s is ranked according to $P(R_Q/\ s)$. Precision and recall are computed for $M = 1,..,N$ and plotted here one against the other as an 11 point curve.

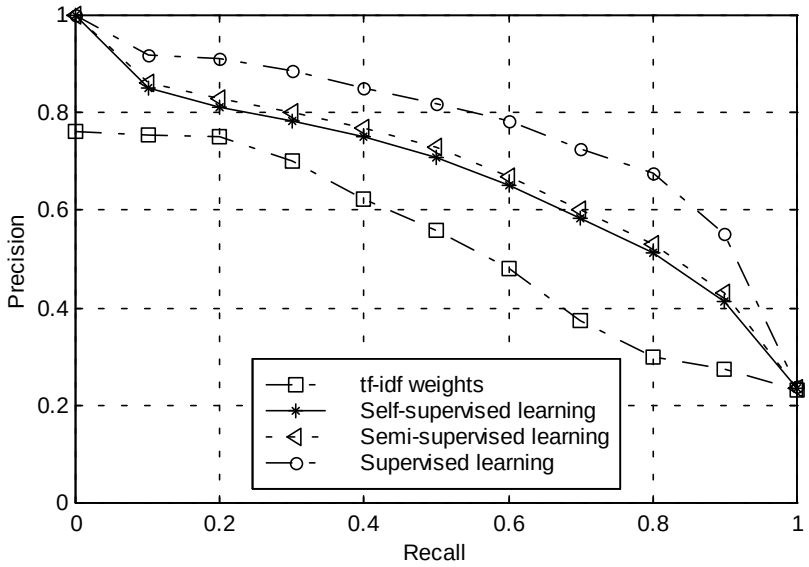

Figure 2. Precision-Recall curves for self-supervised learning (star), base line system (square), semi-supervised learning (triangle) and the supervised learning (circle). The classifier used is the sigmoid perceptron

The curves illustrate the same behavior as table 2, semi-supervised and self-supervised behave similarly and for all recall values their performance increase is half that of the fully supervised system. Self-supervised learning appear as a very promising technique since no labeling is required at all. Note that this method could be applied as well and exactly in the same way for query based summaries.

6 Conclusion

We have described a text summarization system in the context of sentence based extraction summaries. The main idea proposed here is the development of a fully automatic summarization system using a self-learning paradigm. This has been implemented using simple linear classifiers, experiments on Reuters news-wire have shown a clear performance increase. Self-learning allows to reach half of the performance increase allowed by a fully supervised system, and is much more realistic for applications. It can also be used in exactly the same way for query based summaries. Theoretical issues about the behavior of the model and algorithmic improvement are currently investigated.

7 References

Journal Article

1. Dempster A., Laird N., Rubin D. Maximum Likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society, 1977, 39(series B): 1-38.
2. Luhn P.H. Automatic creation of literature abstracts. IBM Journal, 1958, 159-165.
3. Gauch S., Wang J., Rachakonda S.-M. A Corpus Analysis Approach for Automatic Query Expansion and its Extension to Multiple DataBases. ACM Transactions on Information Systems, 1999, 17: 250-269.
4. Patrick E.A., Costello J.P., Monds F.C. Decision directed estimation of a two class descision boundary. IEEE Transactions on Information Theory, 1970, 197-205.
5. J. Sparings. Learning without a teacher. IEEE Transactions on Information Theory, , 1966, pp. 223-230.

Book

6. Bishop C. Neural Networks for Pattern Recognition., Oxford University Press, 1995.

7. *http://boardwatch.internet .com/mag/95/oct/bwm9.html*

8. NIST. TIPSTER Information-Retrieval Text Research Collection on CD-ROM. National Institute of Standards and Technology, Gaithersburg, Maryland, 1993.

9. SUMMAC. TIPSTER Text Summarization Evaluation Conference (SUMMAC). *http://www-nlpir.nist.gov/related_projects/tipster_summac/*

10. Teufel S., Moens M. Argumentative Classification of Extracted Sentences as a First Step Towards Flexible Abstracting. MIT Press, 1999.

Chapter in a Book (or Paper in a Proceedings)

11. Amini M.R. Interactive Learning for Text Summarization. Proceedings of PKDD'2000/MLTIA'2000 Workshop on Machine Learning and Textual Information Access, 2000, pp. 44-52, Lyon France.

12. Aone C., Okurowski M.E., Gorlinsky J., Larsen B. A scalable summarization system using robust NLP. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997, pp. 66-73, Madrid, Spain.

13. Barzilay R., Elhadad M. Using lexical chains for text summarization. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997, pp. 10-17, Madrid, Spain.

14. Blum A., Mitchell T. Combining labeled and unlabeled data with co-training. Proceeding of the Computational Theory, 1998.

15. Burges C. A tutorial on Support Vector Machines for Pattern Recognition. Available at http://svm.research.bell-labs.com/SVMdoc.html, 1998.

16. Carbonell J.G., Goldstein J. The use of MMR, diversity-based reranking for reordering documents and producing summaries. Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, 1998, pp. 335-336, Melbourne, Australia.

17. Chuang W.T., Yang J. Extracting sentence segments for text summarization: a machine learning approach. Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval, 2000, pp. 152--159, Athens, Greece.

18. Goldstein J., Kantrowitz M., Mittal V., Carbonell J. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, 1999, pp. 121-127, Berkeley, USA.

19. Hovy E., Lin C.Y. Automated text summarization in SUMMARIST. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997, pp. 18-24, Madrid, Spain.

20. Klavans J.L., Shaw J. Lexical semantics in summarization. Proceedings of the First Annual Workshop of the IFIP working Group for NLP and KR, 1995, Nantes, France.

21. Kanus D., Mittendorf E., Schauble P., Sheridan P. Highlighting Relevant Passages for Users of the Interactive SPIDER Retrieval System, 1994, TREC-4 proceedings.

22. Kupiec J., Pedersen J., Chen F. A Trainable Document Summarizer. Proceedings of the 18th ACM SIGIR conference on research and development in information retrieval, 1995, pp. 68-73, Seattle, USA.

23. Marcu D. From discourse structures to text summaries. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997, pp. 82-88, Madrid, Spain.

24. Mani I., Bloedorn E. Machine Learning of Generic and User-Focused Summarization. Proceedings of the Fifteenth National Conference on AI, 1998, pp. 821-826.

25. Mitra M., Singhal A., Buckley C. Automatic Text Summarization by Paragraph Extraction. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997, pp. 31-36, Madrid, Spain.

26. Mittal V., Kantrowitz M., Goldstein J., Carbonell J. Selecting Text Spans for Document Summaries: Heuristics and Metrics. Proceedings of the Sixteenth National Conference on AI, 1999, Orlando, USA.

27. Nigam K., McCallum A.K., Thrun S., Mitchell T. Learning to Classify Text from Labeled and Unlabeled Documents. Proceedings of the Fifteenth National Conference on AI, 1998, pp.792-799, Stanford.

28. Nigam K., McCallum A.K., Thrun S., Mitchell T. Text Classification from Labeled and Unlabeled Documents using EM. Machine Learning, 2000, pp. 103-134.

29. Nigam K., Ghani R. Understanding the Behavior of Co-Training. Proceedings of KDD'2000 Workshop on Text Mining, 2000.

30. Radev D., McKeown K. Generating natural language summaries from multiple online sources. Computational Linguistics, 1998.

31. Sparck Jones K. Discourse modeling for automatic summarizing. Technical Report 29D, Computer laboratory, university of Cambridge, 1993.

32. Strzalkowski T., Wang J., Wise B., A robust practical text summarization system. Proceedings of the Fifteenth National Conference on AI, 1998, pp.26-30, Stanford.

33. Teufel S., Moens M. Sentence Extraction as a Classification Task. Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization, 1997, pp. 58-65, Madrid, Spain.

34. Xu J., Croft W.B. Query Expansion Using Local and Global Document Analysis. Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 1996, pp. 4--11, Zurich, Switzerland.

35. Zechner K. Fast Generation of Abstracts from General Domain Text Corpora by Extracting Relevant Sentences. Proceedings of the 16th International Conference on Computational Linguistics, 1996, pp. 986-989, Denmark.