Automatic Text Summarization System for Turkish Using Latent Semantic Analysis

Sefa Kılıç
Bilkent University, Department of Computer Engineering
Ankara, Turkey
sefak@cs.bilkent.edu.tr

Abstract—**In this study, a text summarization system for Turkish language, based on latent semantic analysis method is proposed. This system is also compared with previously implemented systems which use some features to select sentences more suitable than others for taking place in summary. For evaluation of automatically generated summaries, ROUGE evaluation technique is used. This study shows that the LSA method gives comparable results with previous systems.**

I. Introduction

With the fast grow up of the World Wide Web, access to the information is easier than ever. But with the increasing number of resources, access to relevant information is getting harder. To overcome the problem of identifying a document as relevant or not for a specific need, text summarization can be used.

Radev, Hovy and McKeown define the summary as a text that is produced from one or more texts, that conveys important information in the original text(s), and that is no longer than half of the original text(s) and usually significantly less than that [12]. So, summarization method should try to achieve two contrary aims:

- The summary should be short as much as possible.
- The summary should contain the information in the original text as much as possible.

Text summarization can be done by either extraction or abstraction. Extraction is the procedure of selecting the most important sentences from text(s) and using them directly. On the other hand, abstraction is about generating new sentences that summarize text. Abstraction requires natural language generation and since natural language generation is a challenging field itself, abstractive summarization is much harder than extractive summarization.

In terms of dimension, summaries can be formed from one document or a set of documents.

Text summaries can be either query-oriented or generic. In query-oriented summarization, summaries present documents according to the some user specified keywords. On the other hand, generic summaries do not have such queries and have sentences about the overall document(s).

The summarization system that is implemented on this study is **generic** summarizer that produces **extractive** summaries from a **single document**. The system takes a document as input and it produces concise summary.

The organization of this report is as follows. Section II gives information about previous works related to this study. Section III is about two different methods of extractive text summarization. The first method uses features of texts to select summary sentences. The second method which is used in the proposed system selects summary sentences using Latent Semantic Analysis method. Section IV reviews the ROUGE evaluation method used. Section V gives information about document dataset used for experiments and it gives results of experiments by comparing results with previously implemented Turkish summarization systems which are based on surface level features of sentences.

II. Related Work

Text summarization has been studied since 1950s and several methods for summarization of documents has been proposed. One of the most early and most important studies is [11]. It describes the research done at IBM in 1950s. It is proposed that frequency of a particular word is a good measure of the significance of that word. Edmundson describes a system that produces document extracts in [4]. Kupiec et al. describe a trainable document summarizer system in [6] derived from the system in [4], that is able to learn from data. Lin and Hovy studies the importance of sentence position for text summarization in [9].

In later work, Lin uses a lot of features for sentence extraction and used machine learning techniques to analyze the effect of each selection feature on sentence extraction from a text [7].

Barzilay and Elhadad use lexical chains for identifying semantically valuable sentences for extraction [1]. Lexical chain is a sequence of related words in a text.

Gong and Liu propose two generic text summarization methods that create text summaries by ranking and extracting sentences from original documents [5]. The first method uses standard IR methods to measure relevance between sentences and second method uses the Latent Semantic Analysis technique to identify semantically important sentences. The proposed system is based on the second technique proposed in that paper.

Cığır, Mutlu and Çiçekli present one of the first Turkish summarization systems in [2]. It uses several features for ranking sentences. Most relevant sentences are extracted for summary, according to their scores. The study also contributes other researchers who want to study on Turkish text summarization by collecting 115 human generating summaries.

Another Turkish text summarization software[1] was developed by Kemik NLP group of Yıldız Technical Univesity. It selects some sentences as summary sentences on two steps; the first step is assigning a score to each sentence by using some features of sentences and second step is selecting sentences having highest scores.

The fact that there is no ideal summary for a given document or a set of documents and the absence of standard human and automatic evaluation metrics, makes the evaluation of summaries very difficult task. In [10], Lin and Hovy discuss various manual and automatic evaluation metrics using data from the Document Understanding Conference[2] 2001. Lin introduces a set of metrics called Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [8]. The metrics compare produced text summary with a reference summary or a set of reference summaries. Reference summaries used for comparison are human-produced. Some forms of metrics are ROUGE-N which is n-grams based, ROUGE-L which applies the concept of longest common subsequences, skip bigrams (ROUGE-S). In this study, to evaluate summaries, to compare them with manually created summaries, ROUGE-L method is used.

III. Creating Generic Summaries

In this section, the process of extracting sentences from a document is explained. Two methods for selecting sentences are explained in this section. The first method uses several document features and the second method uses latent semantic analysis technique. In this study, the second method was implemented for Turkish language and the results of it were compared with the results of first method, on some dataset.

For both methods, first of all, document is needed to be decomposed into sentences. After that, sentences are ranked. The higher score a sentence gets means more information it gives about the overall document. Finally, top scored sentences are selected to form the summary. How many sentences should the summary contain or ratio between the length of original text and the length of summary can be determined by the user. Sentences in the summary should be in the same order as they appear in the original document. Details of two methods for the process of ranking and selection of sentences are given in following subsections.

A. Summarization Using Features

Several features are used for ranking sentences. Each feature gives a score to each sentence. The score function in (1) calculates the overall score of each sentence S. w_i indicates the weight of feature f_i to the overall score.

$$Score(S) = \sum_{i} w_i Score_{f_i}(S) \tag{1}$$

Features used in [2] are as follows:

- Term Frequency: Words that occur more than others may be important for summary. So, term frequency may be used as a feature. Term frequency for a term in a document is the number of occurrences of that word in that document. To prevent bias towards longer documents, it is normalized with the number of occurrences of all words.
- Title Similarity: Title of a document is important and it gives information what the document is about. It can be assumed that a sentence that has a word which also occurs in the title is important than other sentences.
- Key Phrases: The more content-covering keywords that a sentence has, the more important it is.
- Sentence Position: Sentences at the beginning of texts give the general information of the document, especially in news documents. Since a sentence giving the general information about the document is a good candidate to be in summary of that document, sentence position feature is used.
- Centrality: A sentence which has more common words with other sentences is more likely to be in summary.

Machine learning techniques are used to find the best combination of feature weights of the scoring function.

Features used in the system of Yıldız Technical University NLP Group are as follows:

- Title: This feature is about whether a sentence contains title words.
- High Frequency: The frequency of each word in the document is calculated and a list is constructed by using these words and their frequencies. 10% of words which have highest frequencies are considered as important and a sentence which has these words is likely to be selected as a summary sentence.
- Proper Names: Sentences containing proper names are considered as important.
- Positive Words: Sentences which contain some special words such as ”özetle” (in summary), ”sonuç olarak”, ”sonuçta”, ”neticede” (as a conclusion) in Turkish, summarize and conclude the document and their occurrence in the summary makes summary more successful.
- Negative Words: Sentences which contain some special words such as ”çünkü” (because), ”ancak” (but), ”öyleyse” (therefore) in Turkish usually contain detailed information and they should not be in summary.
- Collocation: This feature is about whether a sentence contains collocations or not.
- Sentence Position: Sentences at the beginning of a text are more important. Also sentences at the end of the text are also considered as important.
- Sentence Length: Sentences of average length are important.
- Quotations: Sentences containing quotations are important.
- Ending Mark: Sentences ending with question mark or

[1] http://www.kemik.yildiz.edu.tr/?id=29

[2] http://duc.nist.gov/

exclamation mark are more important than other sentences.

- Date: A sentence containing date information is considered as important.

Weight of each feature has some default value, determined after some experiments.

B. Summarization Using Latent Semantic Analysis

Gong and Liu propose a document summarization method by applying singular value decomposition [5]. First, terms by sentences matrix is needed to be computed. The $m \times n$ matrix $A = [A_1 A_2 \dots A_n]$ is computed. A_i is the term frequency vector of the sentence i. Since each sentence contains only small portion of all words in the document, the matrix A is usually sparse. m is the number of terms and n is the number of sentences in the document.

Given an $m \times n$ matrix A, where $m \geq n$ without loss of generality, singular value decomposition (SVD) of A is defined as

$$A = U \Sigma V^T \tag{2}$$

where $U = [u_{ij}]$ is an $m \times n$ column-orthonormal matrix whose columns are called left singular vectors. $\Sigma = diag(\sigma_1, \sigma_2 \dots \sigma_n)$ is an $n \times n$ diagonal matrix. Its diagonal elements are non-negative singular values sorted in descending order. $V = [v_{ij}]$ is an $n \times n$ orthonormal matrix whose columns are called right singular vectors. If $rank(A) = r$, then Σ satisfies

$$\sigma_1 \geq \sigma_2 \cdots \geq \sigma_r > \sigma_{r+1} = \cdots = \sigma_n = 0 \tag{3}$$

Each sentence and each term are now represented as a r-dimensional vector in the space provided by the SVD.

Two sentences may be semantically close to each other even if they have no common words. Traditional information retrieval methods are not capable of modeling the relationships among terms. But, SVD is capable of doing it and it can semantically cluster terms and sequences. Gong and Liu give a good example. Words *doctor* and *physician* are synonyms and *hospital*, *medicine*, *nurse* are semantically related words. Since, words *doctor* and *physician* generally appear in similar context that share many words such as *hospital*, *medicine* and *nurse*, they will be mapped near to each other in the r-dimensional vector space.

In the related paper, Gong and Liu propose the method in six steps:

1) Decompose the document D into individual sentences and use these sentences to form the candidate sentence set S, and set $k = 1$.
2) Construct the terms by sentences matrix A for the document D.
3) Perform the SVD on A to obtain singular value matrix Σ, and the right singular vector matrix V^T. In the singular vector space, each sentence i is represented by the column vector $\Psi_i = [v_{i1} v_{i2} \dots v_{ir}]^T$ of V^T.
4) Select the k-th right singular vector from the matrix V^T.
5) Select the sentence which has the largest index value with the k-th right singular vector, and include it in the summary.
6) If k reaches the predefined number, terminate the operation; increment k by one, and go to Step 4.

IV. Perfomance Evaluation

As in [2], the number of sentences in automatically generated summaries is equal to the number of sentences in manually generated summaries. Also in that paper, a part of the dataset is reserved for training. In training phase, optimal contribution of each feature to the overall scoring function is determined by using corpus data allocated for this process. Then by using test data, method is evaluated. One of the advantages of the LSA method is that it does not need a dataset for training. All dataset is used for evaluation of the method.

The first evaluation method, used by Cığır, Mutlu and Çiçekli in their study is searching for exact matches of sentences. It works for their dataset since manual summaries are also constructed from article sentences. However, manual summaries does not need to be extractive. They can be created by humans with newly generated sentences. In such a case, this evaluation method fails. So, another evaluation technique, namely ROUGE[3] is used, which is the second evaluation method used by Cığır, Mutlu and Çiçekli. ROUGE is based on various statistical metrics. It compares an automatically generated summary with manual one.

Das and Matrins give a good survey about text summarization methods and evaluation techniques in [3]. They also mention ROUGE method proposed in [8] by Lin. Let $R = r_1, r_2, \dots, r_m$ be the set of reference summaries (i.e. human-generated) and let s be the automatically generated summary. Let $\Phi_n(d)$ be a binary vector that represents the n-grams contained in the document d. The i-th component $\Phi_n^i(d)$ is 1 if the i-th n-gram is contained in document d. Otherwise $\Phi_n^i(d) = 0$. The ROUGE-N can be computed as follows:

$$\text{ROUGE-N}(S) = \frac{\sum_{r \in R} \langle \Phi_n(r), \Phi_n(s) \rangle}{\sum_{r \in R} \langle \Phi_n(r), \Phi_n(r) \rangle} \tag{4}$$

where $\langle X, Y \rangle$ denotes the inner product of vector X and vector Y. Since, each document in corpus data to be used has one summary, in this case,

$$\text{ROUGE-N}(S) = \frac{\langle \Phi_n(r), \Phi_n(s) \rangle}{\langle \Phi_n(r), \Phi_n(r) \rangle} \tag{5}$$

where r denotes the human generated summary for document d.

Another metric in [8] uses the concept *longest common subsequence* (LCS). Motivation for using LCS is that the longer LCS of two summary sentences, the more similar summaries. Let X be the reference summary sentence and Y be the summary sentence of generated by system. The

[3] http://berouge.com/

sentence-level LCS-based R, P and F values of are defined as follows:

$$R_{LCS} = \frac{LCS(X,Y)}{|X|} \tag{6}$$

$$P_{LCS} = \frac{LCS(X,Y)}{|Y|} \tag{7}$$

$$F_{LCS} = \frac{(1+\beta^2)R_{LCS}P_{LCS}}{R_{LCS}+\beta^2 P_{LCS}} \tag{8}$$

where $LCS(X,Y)$ denotes the length of the longest common subsequence between X and Y. $|X|$ and $|Y|$ denotes the length of X and the length of Y respectively. β is a parameter used to balance precision and recall.

When applying LCS-based metric to summary-level, the union of LCS matches between each reference sentence r_i of reference summary R containing a total of m words and every sentence s_j of automatically generated summary S containing a total of n words. ROUGE-L, the summary-level LCS-based F-measure can be computed as follows:

$$R_{LCS} = \frac{\sum_{i=1}^{u} LCS_{\cup}(r_i,S)}{m} \tag{9}$$

$$R_{LCS} = \frac{\sum_{i=1}^{u} LCS_{\cup}(r_i,S)}{n} \tag{10}$$

$$\text{ROUGE-L}(S) = F_{LCS} = \frac{(1+\beta^2)R_{LCS}P_{LCS}}{R_{LCS}+\beta^2 P_{LCS}} \tag{11}$$

$LCS_{\cup}(r_i,C)$ is the LCS score of the *union* longest common subsequence between reference sentence r_i and automatically generated summary S. Again β is a parameter to balance precision and recall.

ROUGE-L is used for evaluation of summaries constructed by the system. Precision, recall and F-measure values are given in Section V. In experiments, $\beta = 1$ so,

$$\text{ROUGE-L}(S) = F_{LCS} = \frac{2 \times R_{LCS} \times P_{LCS}}{R_{LCS}+P_{LCS}} \tag{12}$$

V. Experiment

A. Data Corpus

As data corpus, news articles constructed in [2] were used. It consists of 120 news articles and their manually constructed summaries. Sentences for a summary were selected from the article verbatim. Statistics of the dataset is given at Table I.

B. Results

Using implemented system, a summary was generated for each news article and it was compared with manually constructed summary by using ROUGE-L method. To compare the LSA method with the method that uses surface level features of a text, summaries of same articles were constructed by using summarization software of Kemik NLP group of Yıldız Technical Univesity. Although, in [2], Cığır, Mutlu and Çiçekli does not mention about the ROUGE metric they use, the results were also compared with theirs. Precision, recall and F-measure values of results for three different methods are given in Table II.

Statistics of news articles	
Number of articles	120
Average number of sentences per article	21.08
Average number of words per article	334.75
Maximum number of sentences in an article	79
Minimum number of sentences in an article	4
Maximum number of words in an article	1051
Minimum number of words in an article	75
Statistics of summaries	
Number of summaries of articles	120
Average number of sentences per summary	5.70
Average number of words per summary	132.72
Maximum number of sentences in a summary	17
Minimum number of sentences in a summary	1
Maximum number of words in a summary	402
Minimum number of words in a summary	27

TABLE I
STATISTICS OF THE CORPUS

	LSA	Features1	Features2
Recall	0.606	0.642	0.540
Precision	0.597	0.640	0.809
F-measure	0.582	0.616	0.648

TABLE II
RECALL, PRECISION AND F-MEASURE SCORES FOR THREE DIFFERENT METHODS. LSA STANDS FOR THE METHOD USED IN THE PROPOSED SYSTEM, FEATURES1 SHOWS THE RESULTS OF THE SUMMARIZATION SYSTEM OF KEMIK NLP GROUP OF YILDIZ TECHNICAL UNIVESITY. FINALLY, FEATURES2 SHOWS RESULTS OF THE METHOD OF CIĞIR, MUTLU AND ÇIÇEKLI, GIVEN IN [2].

The recall performance of the proposed system which uses LSA is between two other systems. The system which has the highest recall value is the system of Kemik NLP group of Yıldız Technical Univesity. However, the precision and f-measure results are lower than both other methods. The system proposed in [2] has best precision and f-measure values.

VI. Conclusion

In this study, an automatic text summarization system for Turkish language was proposed. This system is based on LSA method, which is capable of capturing and modeling relationships between terms unlike the method used in other systems mentioned.

Evaluation of generated summaries with reference summaries is not a trivial task. Method of comparison of sentences for exact matches is not a good idea, because it does not work if reference summaries are not extractive. So, another method, ROUGE-L was used which is based on longest common subsequences of terms in automatic and reference summary sentences.

Experiments on a news articles dataset were run and results are given in Section V. Results of the system proposed are comparable with other previous Turkish text summarization systems.

References

[1] Regina Barzilay and Michael Elhadad. Using lexical chains for text summarization. In *In Proceedings of the ACL Workshop on Intelligent Scalable Text Summarization*, pages 10–17, 1997.

[2] Cığır Celal, Kutlu Mücahid, and Çiçekli İlyas. Generic text summarization for turkish. In *Proceedings of the 24th Internation Symposium on Computer and Information Sciences*, 2009.

[3] Dipanjan Das and André F.T. Martins. A survey on automatic text summarization. Literature Survey for the Language and Statistics II course at CMU, 2007.

[4] H. P. Edmundson. New methods in automatic extracting. *Journal of the ACM*, 16(2):264–285, 1969.

[5] Yihong Gong and Xin Liu. Generic text summarization using relevance measure and latent semantic analysis. In *SIGIR' 01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 19–25, New York, NY, USA, 2001. ACM.

[6] Julian Kupiec, Jan Pedersen, and Francine Chen. A trainable document summarizer. In *SIGIR '95: Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 68–73, New York, NY, USA, 1995. ACM.

[7] Chin-Yew Lin. Training a selection function for extraction. In *CIKM '99: Proceedings of the eighth international conference on Information and knowledge management*, pages 55–62, New York, NY, USA, 1999. ACM.

[8] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Proc. ACL workshop on Text Summarization Branches Out*, page 10, 2004.

[9] Chin-Yew Lin and Eduard Hovy. Identifying topics by position. In *Proceedings of the fifth conference on Applied natural language processing*, pages 283–290, Morristown, NJ, USA, 1997. Association for Computational Linguistics.

[10] Chin-Yew Lin and Eduard Hovy. Manual and automatic evaluation of summaries. In *Proceedings of the ACL-02 Workshop on Automatic Summarization*, pages 45–51, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[11] H. P. Luhn. The automatic creation of literature abstracts. *IBM Journal of Research and Development*, 2(2), 1958.

[12] Dragomir R. Radev, Eduard H. Hovy, and Kathleen McKeown. Introduction to the special issue on summarization. *Computational Linguistics*, 28(4):399–408, 2002.