Text Summarization Techniques: A Brief Survey

Mehdi Allahyari*, Seyedamin Pouriyeh†, Mehdi Assefi†, Saeid Safaei†, Elizabeth D. Trippe‡,
Juan B. Gutierrez‡, Krys Kochut†
*Department of Computer Science, Georgia Sothern University, Statesboro, USA
†Department of Computer Science, University of Georgia, Athens, USA
‡Department of Mathematics Institute of Bioinformatics University of Georgia Athens, USA

***Abstract*—In recent years, there has been a explosion in the amount of text data from a variety of sources. This volume of text is an invaluable source of information and knowledge which needs to be effectively summarized to be useful. Text summarization is the task of shortening a text document into a condensed version keeping all the important information and content of the original document. In this review, the main approaches to automatic text summarization are described. We review the different processes for summarization and describe the effectiveness and shortcomings of the different methods.**

***Keywords*—Text summarization; extractive summary; abstractive summary knowledge bases; topic models**

I. Introduction

With the dramatic growth of the Internet, people are overwhelmed by the tremendous amount of online information and documents. This expanding availability of documents has demanded exhaustive research in the area of automatic text summarization. According to Radef et al. [1] a *summary* is defined 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".

Automatic text summarization is the task of producing a concise and fluent summary while preserving key information content and overall meaning. In recent years, numerous approaches have been developed for automatic text summarization and applied widely in various domains. For example, search engines generate snippets as the previews of the documents [2]. Other examples include news websites which produce condensed descriptions of news topics usually as headlines to facilitate browsing or knowledge extractive approaches in different domains [3]–[6].

Automatic text summarization is very challenging, because when we as humans summarize a piece of text, we usually read it entirely to develop our understanding, and then write a summary highlighting its main points. Since computers lack human knowledge and language capability, it makes automatic text summarization a very difficult and non-trivial task.

Automatic text summarization gained attraction as early as the 1950s. An important research of these days was [7] for summarizing scientific documents. Luhn et al. [7] introduced a method to extract salient sentences from the text using features such as *word* and *phrase frequency*. They proposed to weight the sentences of a document as a function of high frequency words, ignoring very high frequency common words. Edmundson et al. [8] described a paradigm based on *key phrases* which in addition to standard frequency depending weights, used the following three methods to determine the sentence weight:

1) *Cue Method*: The relevance of a sentence is calculated based on the presence or absence of certain cue words in the cue dictionary.
2) *Title Method*: The weight of a sentence is computed as the sum of all the content words appearing in the title and headings of a text.
3) *Location Method*: This method assumes that sentences appearing in the beginning of document as well as the beginning of individual paragraphs have a higher probability of being relevant.

Since then, many works have been published to address the problem of automatic text summarization (see [9], [10] for more information about more advanced techniques until 2000s).

In general, there are two different approaches for automatic summarization: *extraction* and *abstraction*. *Extractive summarization* methods work by identifying important sections of the text and generating them verbatim; thus, they depend only on extraction of sentences from the original text. In contrast, *abstractive summarization* methods aim at producing important material in a new way. In other words, they interpret and examine the text using advanced natural language techniques in order to generate a new shorter text that conveys the most critical information from the original text. Even though summaries created by humans are usually not extractive, most of the summarization research today has focused on extractive summarization. Purely extractive summaries often times give better results compared to automatic abstractive summaries [10]. This is because of the fact that abstractive summarization methods cope with problems such as semantic representation, inference and natural language generation which are relatively harder than data-driven approaches, such as sentence extraction. As a matter of fact, there is no completely abstractive summarization system today. Existing abstractive summarizers often rely on an extractive preprocessing component to produce the abstract of the text [11], [12].

Consequently, in this paper we focus on extractive summarization methods and provide an overview of some of the most dominant approaches in this category. There are a number of papers that provide extensive overviews of text summarization techniques and systems [13]–[16].

The rest of the paper is organized as follows: Section II describes the extractive summarization approaches. Topic representation methods are explained in Section III. Section IV

details knowledge bases and automatic summarization. Section V explains the impact of context in the summarization task. Indicator representation approaches are described in Section VI. Finally, Section VII outlines the evaluation methods for summarization.

II. Extractive Summarization

As mentioned before, extractive summarization techniques produce summaries by choosing a subset of the sentences in the original text. These summaries contain the most important sentences of the input. Input can be a single document or multiple documents.

In order to better understand how summarization systems work, we describe three fairly independent tasks which all summarizers perform [15]: *1*) Construct an intermediate representation of the input text which expresses the main aspects of the text. *2*) Score the sentences based on the representation. *3*) Select a summary comprising of a number of sentences.

A. Intermediate Representation

Every summarization system creates some intermediate representation of the text it intends to summarize and finds salient content based on this representation. There are two types of approaches based on the representation: *topic representation* and *indicator representation*. *Topic representation* approaches transform the text into an intermediate representation and interpret the topic(s) discussed in the text.

Topic representation-based summarization techniques differ in terms of their complexity and representation model, and are divided into frequency-driven approaches, topic word approaches, latent semantic analysis and Bayesian topic models [15]. We elaborate topic representation approaches in the following sections. *Indicator representation* approaches describe every sentence as a list of features (indicators) of importance such as sentence length, position in the document, having certain phrases, etc.

B. Sentence Score

When the intermediate representation is generated, we assign an *importance score* to each sentence. In topic representation approaches, the score of a sentence represents how well the sentence explains some of the most important topics of the text. In most of the indicator representation methods, the score is computed by aggregating the evidence from different indicators. Machine learning techniques are often used to find indicator weights.

C. Summary Sentences Selection

Eventually, the summarizer system selects the top k most important sentences to produce a summary. Some approaches use greedy algorithms to select the important sentences and some approaches may convert the selection of sentences into an optimization problem where a collection of sentences is chosen, considering the constraint that it should maximize overall importance and coherency and minimize the redundancy. There are other factors that should be taken into consideration while selecting the important sentences. For example, context in which the summary is created may be helpful in deciding the importance. Type of the document (e.g. news article, email, scientific paper) is another factor which may impact selecting the sentences.

III. Topic Representation Approaches

In this section we describe some of the most widely used topic representation approaches.

A. Topic Words

The topic words technique is one of the common topic representation approaches which aims to identify words that describe the topic of the input document. [7] was one the earliest works that leveraged this method by using frequency thresholds to locate the descriptive words in the document and represent the topic of the document. A more advanced version of Luhn's idea was presented in [17] in which they used log-likelihood ratio test to identify explanatory words which in summarization literature are called the "topic signature". Utilizing topic signature words as topic representation was very effective and increased the accuracy of multi-document summarization in the news domain [18]. For more information about log-likelihood ratio test, see [15].

There are two ways to compute the importance of a sentence: as a function of the number of topic signatures it contains, or as the proportion of the topic signatures in the sentence. Both sentence scoring functions relate to the same topic representation, however, they might assign different scores to sentences. The first method may assign higher scores to longer sentences, because they have more words. The second approach measures the density of the topic words.

B. Frequency-driven Approaches

When assigning weights of words in topic representations, we can think of binary (0 or 1) or real-value (continuous) weights and decide which words are more correlated to the topic. The two most common techniques in this category are: *word probability* and *TFIDF* (Term Frequency Inverse Document Frequency).

1) Word Probability: The simplest method to use frequency of words as indicators of importance is *word probability*. The probability of a word w is determined as the number of occurrences of the word, $f(w)$, divided by the number of all words in the input (which can be a single document or multiple documents):

$$P(w) = \frac{f(w)}{N} \tag{1}$$

Vanderwende et al. [19] proposed the SumBasic system which uses only the word probability approach to determine sentence importance. For each sentence, S_j, in the input, it assigns a weight equal to the average probability of the words in the sentence:

$$g(S_j) = \frac{\sum_{w_i \in S_j} P(w_i)}{|\{w_i | w_i \in S_j\}|} \tag{2}$$

where, $g(S_j)$ is the weight of sentence S_j.

In the next step, it picks the best scoring sentence that contains the highest probability word. This step ensures that the highest probability word, which represents the topic of the document at that point, is included in the summary. Then for each word in the chosen sentence, the weight is updated:

$$p_{new}(w_i) = p_{old}(w_i)p_{old}(w_i) \tag{3}$$

This word weight update indicates that the probability of a word appearing in the summary is lower than a word occurring once. The aforementioned selection steps will repeat until the desired length summary is reached. The sentence selection approach used by SumBasic is based on the greedy strategy. Yih et al. [20] used an optimization approach (as sentence selection strategy) to maximize the occurrence of the important words globally over the entire summary. [21] is another example of using an optimization approach.

2) TFIDF: Since word probability techniques depend on a stop word list in order to not consider them in the summary and because deciding which words to put in the stop list is not very straight forward, there is a need for more advanced techniques. One of the more advanced and very typical methods to give weight to words is TFIDF (Term Frequency Inverse Document Frequency). This weighting technique assesses the importance of words and identifies very common words (that should be omitted from consideration) in the document(s) by giving low weights to words appearing in most documents. The weight of each word w in document d is computed as follows:

$$q(w) = f_d(w) * log\frac{|D|}{f_D(w)} \tag{4}$$

where $f_d(w)$ is term frequency of word w in the document d, $f_D(w)$ is the number of documents that contain word w and $|D|$ is the number of documents in the collection D. For more information about TFIDF and other term weighting schemes, see [22]. TFIDF weights are easy and fast to compute and also are good measures for determining the importance of sentences, therefore many existing summarizers [10], [21], [23] have utilized this technique (or some form of it).

Centroid-based summarization, another set of techniques which has become a common baseline, is based on TFIDF topic representation. This kind of method ranks sentences by computing their salience using a set of features. A complete overview of the centroid-based approach is available in [24] but we outline briefly the basic idea.

The first step is topic detection and documents that describe the same topic clustered together. To achieve this goal, TFIDF vector representations of the documents are created and those words whose TFIDF scores are below a threshold are removed. Then, a clustering algorithm is run over the TFIDF vectors, consecutively adding documents to clusters and recomputing the centroids according to:

$$\boldsymbol{c}_j = \frac{\sum_{d \in C_j} d}{|C_j|} \tag{5}$$

where $\boldsymbol{c}_j$ is the centroid of the jth cluster and C_j is the set of documents that belong to that cluster. *Centroids* can be considered as pseudo-documents that consist of those words whose TFIDF scores are higher than the threshold and form the cluster.

The second step is using centroids to identify sentences in each cluster that are central to topic of the entire cluster. To accomplish this goal, two metrics are defined [25]: *cluster-based relative utility* (CBRU) and *cross-sentence informational subsumption* (CSIS). CBRU decides how relevant a particular sentence is to the general topic of the entire cluster and CSIS measure redundancy among sentences. In order to approximate two metrics, three features (i.e. central value, positional value and first-sentence overlap) are used. Next, the final score of each sentence is computed and the selection of sentences is determined. For another related work, see [26].

C. Latent Semantic Analysis

Latent semantic analysis (LSA) which is introduced by [27], is an unsupervised method for extracting a representation of text semantics based on observed words. Gong and Liu [28] initially proposed a method using LSA to select highly ranked sentences for single and multi-document summarization in the news domain. The LSA method first builds a term-sentence matrix (n by m matrix), where each row corresponds to a word from the input (n words) and each column corresponds to a sentence (m sentences). Each entry a_{ij} of the matrix is the weight of the word i in sentence j. The weights of the words are computed by TFIDF technique and if a sentence does not have a word the weight of that word in the sentence is zero. Then singular value decomposition (SVD) is used on the matrix and transforms the matrix A into three matrices: $A = U\Sigma V^T$.

Matrix U $(n \times m)$ represents a term-topic matrix having weights of words. Matrix Σ is a diagonal matrix $(m \times m)$ where each row i corresponds to the weight of a topic i. Matrix V^T is the topic-sentence matrix. The matrix $D = \Sigma V^T$ describes how much a sentence represent a topic, thus, d_{ij} shows the weight of the topic i in sentence j.

Gong and Liu's method was to choose one sentence per each topic, therefore, based on the length of summary in terms of sentences, they retained the number of topics. This strategy has a drawback due to the fact that a topic may need more than one sentence to convey its information. Consequently, alternative solutions were proposed to improve the performance of LSA-based techniques for summarization. One enhancement was to leverage the weight of each topic to decide the relative size of the summary that should cover the topic, which gives the flexibility of having a variable number of sentences. Another advancement is described in [29]. Steinberger et al. [29] introduced a LSA-based method which achieves a significantly better performance than the original work. They realized that the sentences that discuss some of important topics are good candidates for summaries, thus, in order to locate those sentences they defined the weight of the sentence as follows:

Let g be the "weight" function, then

$$g(s_i) = \sqrt{\sum_{j=1}^{m} d_{ij}^2} \tag{6}$$

For other variations of LSA technique, see [30], [31].

D. Bayesian Topic Models

Many of the existing multi-document summarization methods have two limitations [32]: *1*) They consider the sentences as independent of each other, so topics embedded in the documents are disregarded. *2*) Sentence scores computed by most existing approaches typically do not have very clear probabilistic interpretations, and many of the sentence scores are calculated using heuristics.

Bayesian topic models are probabilistic models that uncover and represent the topics of documents. They are quite powerful and appealing, because they represent the information (i.e. topics) that are lost in other approaches. Their advantage in describing and representing topics in detail enables the development of summarizer systems which can determine the similarities and differences between documents to be used in summarization [33].

Apart from enhancement of topic and document representation, topic models often utilize a distinct measure for scoring the sentence called Kullbak-Liebler (KL). The KL is a measure of difference (divergence) between two probability distributions P and Q [34]. In summarization where we have probability of words, the KL divergence of Q from P over the words w is defined as:

$$D_{KL}(P||Q) = \sum_{w} P(w) \log \frac{P(w)}{Q(w)} \tag{7}$$

where $P(w)$ and $Q(w)$ are probabilities of w in P and Q.

KL divergence is an interesting method for scoring sentences in the summarization, because it shows the fact that good summaries are intuitively similar to the input documents. It describes how the importance of words alters in the summary in comparison with the input, i.e. the KL divergence of a good summary and the input will be low.

Probabilistic topic models have gained dramatic attention in recent years in various domains [35]–[43]. *Latent Dirichlet allocation* (LDA) model is the state of the art unsupervised technique for extracting thematic information (topics) of a collection of documents. A complete review for LDA can be found in [44], [45], but the main idea is that documents are represented as a random mixture of latent topics, where each topic is a probability distribution over words.

LDA has been extensively used for multi-document summarization recently. For example, Daume et al. [46] proposed BAYESUM, a Bayesian summarization model for query-focused summarization. Wang et al. [32] introduced a Bayesian sentence-based topic model for summarization which used both term-document and term-sentence associations. Their system achieved significance performance and outperformed many other summarization methods. Celikyilmaz et al. [47] describe multi-document summarization as a prediction problem based on a two-phase hybrid model. First, they propose a hierarchical topic model to discover the topic structures of all sentences. Then, they compute the similarities of candidate sentences with human-provided summaries using a novel tree-based sentence scoring function. In the second step they make use of these scores and train a regression model according the lexical and structural characteristics of the sentences, and employ the model to score sentences of new documents (unseen documents) to form a summary.

IV. KNOWLEDGE BASES AND AUTOMATIC SUMMARIZATION

The goal of automatic text summarization is to create summaries that are similar to human-created summaries. However, in many cases, the soundness and readability of created summaries are not satisfactory, because the summaries do not cover all the semantically relevant aspects of data in an effective way. This is because many of the existing text summarization techniques do not consider the semantics of words. A step towards building more accurate summarization systems is to combine summarization techniques with knowledge bases (semantic-based or ontology-based summarizers).

The advent of human-generated knowledge bases and various ontologies in many different domains (e.g. Wikipedia, YAGO, DBpedia, etc.) has opened further possibilities in text summarization , and reached increasing attention recently. For example, Henning et al. [48] present an approach to sentence extraction that maps sentences to concepts of an ontology. By considering the ontology features, they can improve the semantic representation of sentences which is beneficial in selection of sentences for summaries. They experimentally showed that ontology-based extraction of sentences outperforms baseline summarizers. Chen et al. [49] introduce a user query-based text summarizer that uses the UMLS medical ontology to make a summary for medical text. Baralis et al. [50] propose a Yago-based summarizer that leverages YAGO ontology [51] to identify key concepts in the documents. The concepts are evaluated and then used to select the most representative document sentences. Sankarasubramaniam et al. [52] introduce an approach that employs Wikipedia in conjunction with a graph-based ranking technique. First, they create a bipartite sentence-concept graph, and then use an iterative ranking algorithm for selecting summary sentences.

V. IMPACT OF CONTEXT IN SUMMARIZATION

Summarization systems often have additional evidence they can utilize in order to specify the most important topics of document(s). For example when summarizing blogs, there are discussions or comments coming after the blog post that are good sources of information to determine which parts of the blog are critical and interesting. In scientific paper summarization, there is a considerable amount of information such as cited papers and conference information which can be leveraged to identify important sentences in the original paper. In the following, we describe some the contexts in more details.

A. Web Summarization

Web pages contains lots of elements which cannot be summarized such as pictures. The textual information they have is often scarce, which makes applying text summarization techniques limited. Nonetheless, we can consider the context of a web page, i.e. pieces of information extracted from content of all the pages linking to it, as additional material to improve

summarization. The earliest research in this regard is [53] where they query web search engines and fetch the pages having links to the specified web page. Then they analyze the candidate pages and select the best sentences containing links to the web page heuristically. Delort et al. [54] extended and improved this approach by using an algorithm trying to select a sentence about the same topic that covers as many aspects of the web page as possible.

For blog summarization, [55] proposed a method that first derives representative words from comments and then selects important sentences from the blog post containing representative words. For more related works, see [56]–[58].

B. Scientific Articles Summarization

A useful source of information when summarizing a scientific paper (i.e. citation-based summarization) is to find other papers that cite the target paper and extract the sentences in which the references take place in order to identify the important aspects of the target paper. Mei et al. [59] propose a language model that gives a probability to each word in the citation context sentences. They then score the importance of sentences in the original paper using the KL divergence method (i.e. finding the similarity between a sentence and the language model). For more information, see [60], [61].

C. Email Summarization

Email has some distinct characteristics that indicates the aspects of both spoken conversation and written text. For example, summarization techniques must consider the interactive nature of the dialog as in spoken conversations. Nenkova et al. [62] presented early research in this regard, by proposing a method to generate a summary for the first two levels of the thread discussion. A thread consists of one or more conversations between two or more participants over time. They select a message from the root message and from each response to the root, considering the overlap with root context. Rambow et al. [63] used a machine learning technique and included features related to the thread as well as features of the email structure such as position of the sentence in the tread, number of recipients, etc. Newman et al. [64] describe a system to summarize a full mailbox rather than a single thread by clustering messages into topical groups and then extracting summaries for each cluster.

VI. Indicator Representation Approaches

Indicator representation approaches aim to model the representation of the text based on a set of features and use them to directly rank the sentences rather than representing the topics of the input text. Graph-based methods and machine learning techniques are often employed to determine the important sentences to be included in the summary.

A. Graph Methods for Summarization

Graph methods, which are influenced by PageRank algorithm [65], represent the documents as a connected graph. Sentences form the vertices of the graph and edges between the sentences indicate how similar the two sentences are. A common technique employed to connect two vertices is to measure the similarity of two sentences and if it is greater then a threshold they are connected. The most often used method for similarity measure is cosine similarity with TFIDF weights for words.

This graph representation results in two outcomes. First, the partitions (sub-graphs) included in the graph, create discrete topics covered in the documents. The second outcome is the identification of the important sentences in the document. Sentences that are connected to many other sentences in the partition are possibly the center of the graph and more likely to be included in the summary.

Graph-based methods can be used for single as well as multi-document summarization [10]. Since they do not need language-specific linguistic processing other than sentence and word boundary detection, they can also be applied to various languages [66]. Nonetheless, using TFIDF weighting scheme for similarity measure has limitations, because it only preserves frequency of words and does not take the syntactic and semantic information into account. Thus, similarity measures based on syntactic and semantic information enhances the performance of the summarization system [67]. For more graph-based approaches, see [15].

B. Machine Learning for Summarization

Machine learning approaches model the summarization as a classification problem. [68] is an early research attempt at applying machine learning techniques for summarization. Kupiec et al. develop a classification function, *naive-Bayes classifier*, to classify the sentences as summary sentences and non-summary sentences based on the features they have, given a training set of documents and their extractive summaries. The classification probabilities are learned statistically from the training data using Bayes' rule:

$$P(s \in \mathcal{S}|F_1, F_2, \ldots, F_k) = \frac{P(F_1, F_2, \ldots, F_k)|s \in \mathcal{S})P(s \in \mathcal{S})}{P(F_1, F_2, \ldots, F_k)} \tag{8}$$

Where, s is a sentence from the document collection, $F_1, F_2, \ldots, F_k$ are features used in classification and $\mathcal{S}$ is the summary to be generated. Assuming the conditional independence between the features:

$$P(s \in \mathcal{S}|F_1, F_2, \ldots, F_k) = \frac{\prod_{i=1}^{k} P(F_i|s \in \mathcal{S})P(s \in \mathcal{S})}{\prod_{i=1}^{k} P(F_i).} \tag{9}$$

The probability a sentence to belongs to the summary is the score of the sentence. The selected classifier plays the role of a sentence scoring function. Some of the frequent features used in summarization include the position of sentences in the document, sentence length, presence of uppercase words, similarity of the sentence to the document title, etc. Machine learning approaches have been widely used in summarization by [69]–[71], to name a few.

Naive Bayes, decision trees, support vector machines, Hidden Markov models and Conditional Random Fields are among the most common machine learning techniques used

for summarization. One fundamental difference between classifiers is that sentences to be included in the summary have to be decided *independently*. It turns out that methods explicitly assuming the dependency between sentences such as Hidden Markov model [72] and Conditional Random Fields [73] often outperform other techniques.

One of the primary issues in utilizing supervised learning methods for summarization is that they need a set of training documents (labeled data) to train the classifier, which may not be always easily available. Researchers have proposed some alternatives to cope with this issue:

- **Annotated corpora creation:** Creating annotated corpus for summarization greatly benefits the researchers, because more public benchmarks will be available which makes it easier to compare different summarization approaches together. It also lowers the risk of overfitting with a limited data. Ulrich et al. [74] introduce a publicly available annotated email corpus and its creation process. However, creating annotated corpus is very time consuming and more critically, there is no standard agreement on choosing the sentences, and different people may select varied sentences to construct the summary.
- **Semi-supervised approaches:** Using a semi-supervised technique to train a classifier. In semi-supervised learning we utilize the unlabeled data in training. There is usually a small amount of labeled data along with a large amount of unlabeled data. For complete overview of semi-supervised learning, see [75]. Wong et al. [70] proposed a semi-supervised method for extractive summarization. They co-trained two classifiers iteratively to exploit unlabeled data. In each iteration, the unlabeled training examples (sentences) with top scores are included in the labeled training set, and the two classifiers are trained on the new training data.

Machine learning methods have been shown to be very effective and successful in single and multi-document summarization, specifically in class specific summarization where classifiers are trained to locate particular type of information such as scientific paper summarization [61], [76], [77] and biographical summaries [78]–[80].

VII. Evaluation

Evaluation of a summary is a difficult task because there is no ideal summary for a document or a collection of documents and the definition of a good summary is an open question to large extent [16]. It has been found that human summarizers have low agreement for evaluating and producing summaries. Additionally, prevalent use of various metrics and the lack of a standard evaluation metric has also caused summary evaluation to be difficult and challenging.

A. Evaluation of Automatically Produced Summaries

There have been several evaluation campaigns since the late 1990s in the US [16]. They include SUMMAC (1996-1998) [81], DUC (the Document Understanding Conference, 2000-2007) [82], and more recently TAC (the Text Analysis Conference, 2008-present) [1]. These conferences have primary role in design of evaluation standards and evaluate the summaries based on human as well as automatic scoring of the summaries.

In order to be able to do automatic summary evaluation, we need to conquer three major difficulties: *i*) It is fundamental to decide and specify the most important parts of the original text to preserve. *ii*) Evaluators have to automatically identify these pieces of important information in the candidate summary, since this information can be represented using disparate expressions. *iii*) The readability of the summary in terms of grammar and coherence has to be evaluated.

B. Human Evaluation

The simplest way to evaluate a summary is to have a human assess its quality. For example, in DUC, the judges would evaluate the coverage of the summary, i.e. how much the candidate summary covered the original given input. In more recent paradigms, in particular TAC, query-based summaries have been created. Then judges evaluate to what extent a summary answers the given query. The factors that human experts must consider when giving scores to each candidate summary are grammar, non redundancy, integration of most important pieces of information, structure and coherence. For more information, see [16].

C. Automatic Evaluation Methods

There has been a set of metrics to automatically evaluate summaries since the early 2000s. ROUGE is the most widely used metric for automatic evaluation.

1) ROUGE: Lin [83] introduced a set of metrics called Recall-Oriented Understudy for Gisting Evaluation (ROUGE) to automatically determine the quality of a summary by comparing it to human (reference) summaries. There are several variations of ROUGE (see [83]), and here we just mention the most broadly used ones:

- **ROUGE-n:** This metric is recall-based measure and based on comparison of n-grams. a series of n-grams (mostly two and three and rarely four) is elicited from the reference summaries and the candidate summary (automatically generated summary). Let p be "the number of common n-grams between candidate and reference summary", and q be "the number of n-grams extracted from the reference summary only". The score is computed as:
$$\text{ROUGE-}n = \frac{p}{q} \tag{10}$$
- **ROUGE-L:** This measure employs the concept of *longest common subsequence* (LCS) between the two sequences of text. The intuition is that the longer the LCS between two summary sentences, the more similar they are. Although this metric is more flexible than the previous one, it has a drawback that all n-grams must be consecutive. For more information about this metric and its refined metric, see [83].
- **ROUGE-SU:** This metric called *skip bi-gram and unigram* ROUGE and considers bi-grams as well as unigrams. This metric allows insertion of words between the first and the last words of the bi-grams, so they do not need to be consecutive sequences of words.

[1] http://www.nist.gov/tac/about/index.html

VIII. Conclusions

The increasing growth of the Internet has made a huge amount of information available. It is difficult for humans to summarize large amounts of text. Thus, there is an immense need for automatic summarization tools in this age of information overload. In this paper, we emphasized various extractive approaches for single and multi-document summarization. We described some of the most extensively used methods such as topic representation approaches, frequency-driven methods, graph-based and machine learning techniques. Although it is not feasible to explain all diverse algorithms and approaches comprehensively in this paper, we think it provides a good insight into recent trends and progresses in automatic summarization methods and describes the state-of-the-art in this research area.

References

[1] D. R. Radev, E. Hovy, and K. McKeown, "Introduction to the special issue on summarization," *Computational linguistics*, vol. 28, no. 4, pp. 399–408, 2002.

[2] A. Turpin, Y. Tsegay, D. Hawking, and H. E. Williams, "Fast generation of result snippets in web search," in *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2007, pp. 127–134.

[3] M. Allahyari, S. Pouriyeh, M. Assefi, S. Safaei, E. D. Trippe, J. B. Gutierrez, and K. Kochut, "A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques," *ArXiv e-prints*, 2017.

[4] G. K. Savova, J. J. Masanz, P. V. Ogren, J. Zheng, S. Sohn, K. C. Kipper-Schuler, and C. G. Chute, "Mayo clinical text analysis and knowledge extraction system (ctakes): architecture, component evaluation and applications," *Journal of the American Medical Informatics Association*, vol. 17, no. 5, pp. 507–513, 2010.

[5] E. D. Trippe, J. B. Aguilar, Y. H. Yan, M. V. Nural, J. A. Brady, M. Assefi, S. Safaei, M. Allahyari, S. Pouriyeh, M. R. Galinski, J. C. Kissinger, and J. B. Gutierrez, "A Vision for Health Informatics: Introducing the SKED Framework.An Extensible Architecture for Scientific Knowledge Extraction from Data," *ArXiv e-prints*, 2017.

[6] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia, and J. Gutierrez, "A comprehensive investigation and comparison of machine learning techniques in the domain of heart disease," in *Computers and Communications (ISCC), 2017 IEEE Symposium on*. IEEE, 2017, pp. 204–207.

[7] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal of research and development*, vol. 2, no. 2, pp. 159–165, 1958.

[8] H. P. Edmundson, "New methods in automatic extracting," *Journal of the ACM (JACM)*, vol. 16, no. 2, pp. 264–285, 1969.

[9] V. Gupta and G. S. Lehal, "A survey of text summarization extractive techniques," *Journal of Emerging Technologies in Web Intelligence*, vol. 2, no. 3, pp. 258–268, 2010.

[10] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Intell. Res.(JAIR)*, vol. 22, no. 1, pp. 457–479, 2004.

[11] K. Knight and D. Marcu, "Statistics-based summarization-step one: Sentence compression," in *AAAI/IAAI*, 2000, pp. 703–710.

[12] T. Berg-Kirkpatrick, D. Gillick, and D. Klein, "Jointly learning to extract and compress," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 481–490.

[13] K. Spärck Jones, "Automatic summarising: The state of the art," *Information Processing & Management*, vol. 43, no. 6, pp. 1449–1481, 2007.

[14] E. Lloret and M. Palomar, "Text summarisation in progress: a literature review," *Artificial Intelligence Review*, vol. 37, no. 1, pp. 1–41, 2012.

[15] A. Nenkova and K. McKeown, "A survey of text summarization techniques," in *Mining Text Data*. Springer, 2012, pp. 43–76.

[16] H. Saggion and T. Poibeau, "Automatic text summarization: Past, present and future," in *Multi-source, Multilingual Information Extraction and Summarization*. Springer, 2013, pp. 3–21.

[17] T. Dunning, "Accurate methods for the statistics of surprise and coincidence," *Computational linguistics*, vol. 19, no. 1, pp. 61–74, 1993.

[18] S. Harabagiu and F. Lacatusu, "Topic themes for multi-document summarization," in *Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2005, pp. 202–209.

[19] L. Vanderwende, H. Suzuki, C. Brockett, and A. Nenkova, "Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion," *Information Processing & Management*, vol. 43, no. 6, pp. 1606–1618, 2007.

[20] W.-t. Yih, J. Goodman, L. Vanderwende, and H. Suzuki, "Multi-document summarization by maximizing informative content-words." in *IJCAI*, vol. 2007, 2007, p. 20th.

[21] R. M. Alguliev, R. M. Aliguliyev, M. S. Hajirahimova, and C. A. Mehdiyev, "Mcmr: Maximum coverage and minimum redundant text summarization model," *Expert Systems with Applications*, vol. 38, no. 12, pp. 14 514–14 522, 2011.

[22] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information processing & management*, vol. 24, no. 5, pp. 513–523, 1988.

[23] R. M. Alguliev, R. M. Aliguliyev, and N. R. Isazade, "Multiple documents summarization based on evolutionary optimization algorithm," *Expert Systems with Applications*, vol. 40, no. 5, pp. 1675–1689, 2013.

[24] D. R. Radev, H. Jing, M. Styś, and D. Tam, "Centroid-based summarization of multiple documents," *Information Processing & Management*, vol. 40, no. 6, pp. 919–938, 2004.

[25] D. R. Radev, H. Jing, and M. Budzikowska, "Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies," in *Proceedings of the 2000 NAACL-ANLP Workshop on Automatic Summarization*. Association for Computational Linguistics, 2000, pp. 21–30.

[26] X. Wan and J. Yang, "Multi-document summarization using cluster-based link analysis," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2008, pp. 299–306.

[27] S. C. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman, "Indexing by latent semantic analysis," *JASIS*, vol. 41, no. 6, pp. 391–407, 1990.

[28] Y. Gong and X. Liu, "Generic text summarization using relevance measure and latent semantic analysis," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 19–25.

[29] J. Steinberger, M. Poesio, M. A. Kabadjov, and K. Ježek, "Two uses of anaphora resolution in summarization," *Information Processing & Management*, vol. 43, no. 6, pp. 1663–1680, 2007.

[30] B. Hachey, G. Murray, and D. Reitter, "Dimensionality reduction aids term co-occurrence based multi-document summarization," in *Proceedings of the workshop on task-focused summarization and question answering*. Association for Computational Linguistics, 2006, pp. 1–7.

[31] M. G. Ozsoy, I. Cicekli, and F. N. Alpaslan, "Text summarization of turkish texts using latent semantic analysis," in *Proceedings of the 23rd international conference on computational linguistics*. Association for Computational Linguistics, 2010, pp. 869–876.

[32] D. Wang, S. Zhu, T. Li, and Y. Gong, "Multi-document summarization using sentence-based topic models," in *Proceedings of the ACL-IJCNLP 2009 Conference Short Papers*. Association for Computational Linguistics, 2009, pp. 297–300.

[33] I. Mani and E. Bloedorn, "Summarizing similarities and differences among related documents," *Information Retrieval*, vol. 1, no. 1-2, pp. 35–67, 1999.

[34] S. Kullback and R. A. Leibler, "On information and sufficiency," *The Annals of Mathematical Statistics*, pp. 79–86, 1951.

[35] L. Na, L. Ming-xia, L. Ying, T. Xiao-jun, W. Hai-wen, and X. Peng, "Mixture of topic model for multi-document summarization," in *Control and Decision Conference (2014 CCDC), The 26th Chinese*. IEEE, 2014, pp. 5168–5172.

[36] F. C. T. Chua and S. Asur, "Automatic summarization of events from social media." in *ICWSM*, 2013.

[37] Z. Ren, S. Liang, E. Meij, and M. de Rijke, "Personalized time-aware tweets summarization," in *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2013, pp. 513–522.

[38] J. Hannon, K. McCarthy, J. Lynch, and B. Smyth, "Personalized and automatic social summarization of events in video," in *Proceedings of the 16th international conference on Intelligent user interfaces*. ACM, 2011, pp. 335–338.

[39] M. Allahyari and K. Kochut, "Automatic topic labeling using ontology-based topic models," in *Machine Learning and Applications (ICMLA), 2015 IEEE 14th International Conference on*. IEEE, 2015, pp. 259–264.

[40] M. Allahyari, S. Pouriyeh, K. Kochut, and H. R. Arabnia, "A knowledge-based topic modeling approach for automatic topic labeling," *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, vol. 8, no. 9, pp. 335–349, 2017.

[41] M. Allahyari and K. Kochut, "Semantic tagging using topic models exploiting wikipedia category network," in *Semantic Computing (ICSC), 2016 IEEE Tenth International Conference on*. IEEE, 2016, pp. 63–70.

[42] ——, "Semantic context-aware recommendation via topic models leveraging linked open data," in *International Conference on Web Information Systems Engineering*. Springer, 2016, pp. 263–277.

[43] ——, "Discovering coherent topics with entity topic models," in *Web Intelligence (WI), 2016 IEEE/WIC/ACM International Conference on*. IEEE, 2016, pp. 26–33.

[44] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *the Journal of machine Learning research*, vol. 3, pp. 993–1022, 2003.

[45] M. Steyvers and T. Griffiths, "Probabilistic topic models," *Handbook of latent semantic analysis*, vol. 427, no. 7, pp. 424–440, 2007.

[46] H. Daumé III and D. Marcu, "Bayesian query-focused summarization," in *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2006, pp. 305–312.

[47] A. Celikyilmaz and D. Hakkani-Tur, "A hybrid hierarchical model for multi-document summarization," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 815–824.

[48] L. Hennig, W. Umbrath, and R. Wetzker, "An ontology-based approach to text summarization," in *Web Intelligence and Intelligent Agent Technology, 2008. WI-IAT'08. IEEE/WIC/ACM International Conference on*, vol. 3. IEEE, 2008, pp. 291–294.

[49] P. Chen and R. Verma, "A query-based medical information summarization system using ontology knowledge," in *Computer-Based Medical Systems, 2006. CBMS 2006. 19th IEEE International Symposium on*. IEEE, 2006, pp. 37–42.

[50] E. Baralis, L. Cagliero, S. Jabeen, A. Fiori, and S. Shah, "Multi-document summarization based on the yago ontology," *Expert Systems with Applications*, vol. 40, no. 17, pp. 6976–6984, 2013.

[51] F. M. Suchanek, G. Kasneci, and G. Weikum, "Yago: a core of semantic knowledge," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 697–706.

[52] Y. Sankarasubramaniam, K. Ramanathan, and S. Ghosh, "Text summarization using wikipedia," *Information Processing & Management*, vol. 50, no. 3, pp. 443–461, 2014.

[53] E. Amitay and C. Paris, "Automatically summarising web sites: is there a way around it?" in *Proceedings of the ninth international conference on Information and knowledge management*. ACM, 2000, pp. 173–179.

[54] J.-Y. Delort, B. Bouchon-Meunier, and M. Rifqi, "Enhanced web document summarization using hyperlinks," in *Proceedings of the fourteenth ACM conference on Hypertext and hypermedia*. ACM, 2003, pp. 208–215.

[55] M. Hu, A. Sun, and E.-P. Lim, "Comments-oriented blog summarization by sentence extraction," in *Proceedings of the sixteenth ACM conference on Conference on information and knowledge management*. ACM, 2007, pp. 901–904.

[56] B. P. Sharifi, D. I. Inouye, and J. K. Kalita, "Summarization of twitter microblogs," *The Computer Journal*, p. bxt109, 2013.

[57] B. Sharifi, M.-A. Hutton, and J. Kalita, "Summarizing microblogs automatically," in *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*. Association for Computational Linguistics, 2010, pp. 685–688.

[58] M. Hu, A. Sun, and E.-P. Lim, "Comments-oriented document summarization: understanding documents with readers' feedback," in *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2008, pp. 291–298.

[59] Q. Mei and C. Zhai, "Generating impact-based summaries for scientific literature." in *ACL*, vol. 8. Citeseer, 2008, pp. 816–824.

[60] A. Abu-Jbara and D. Radev, "Coherent citation-based summarization of scientific papers," in *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 2011, pp. 500–509.

[61] V. Qazvinian and D. R. Radev, "Scientific paper summarization using citation summary networks," in *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2008, pp. 689–696.

[62] A. Nenkova and A. Bagga, "Facilitating email thread access by extractive summary generation," *Recent advances in natural language processing III: selected papers from RANLP*, vol. 2003, p. 287, 2004.

[63] O. Rambow, L. Shrestha, J. Chen, and C. Lauridsen, "Summarizing email threads," in *Proceedings of HLT-NAACL 2004: Short Papers*. Association for Computational Linguistics, 2004, pp. 105–108.

[64] P. S. Newman and J. C. Blitzer, "Summarizing archived discussions: a beginning," in *Proceedings of the 8th international conference on Intelligent user interfaces*. ACM, 2003, pp. 273–276.

[65] R. Mihalcea and P. Tarau, "Textrank: Bringing order into texts." Association for Computational Linguistics, 2004.

[66] ——, "A language independent algorithm for single and multiple document summarization," 2005.

[67] Y. Chali and S. R. Joty, "Improving the performance of the random walk model for answering complex questions," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*. Association for Computational Linguistics, 2008, pp. 9–12.

[68] J. Kupiec, J. Pedersen, and F. Chen, "A trainable document summarizer," in *Proceedings of the 18th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1995, pp. 68–73.

[69] L. Zhou and E. Hovy, "A web-trained extraction summarization system," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*. Association for Computational Linguistics, 2003, pp. 205–211.

[70] K.-F. Wong, M. Wu, and W. Li, "Extractive summarization using supervised and semi-supervised learning," in *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*. Association for Computational Linguistics, 2008, pp. 985–992.

[71] Y. Ouyang, W. Li, S. Li, and Q. Lu, "Applying regression models to query-focused multi-document summarization," *Information Processing & Management*, vol. 47, no. 2, pp. 227–237, 2011.

[72] J. M. Conroy and D. P. O'leary, "Text summarization via hidden markov models," in *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 2001, pp. 406–407.

[73] D. Shen, J.-T. Sun, H. Li, Q. Yang, and Z. Chen, "Document summarization using conditional random fields." in *IJCAI*, vol. 7, 2007, pp. 2862–2867.

[74] J. Ulrich, G. Murray, and G. Carenini, "A publicly available annotated corpus for supervised email summarization," in *Proc. of aaai email-2008 workshop, chicago, usa*, 2008.

[75] O. Chapelle, B. Schölkopf, A. Zien *et al.*, *Semi-supervised learning*. MIT press Cambridge, 2006, vol. 2.

[76] S. Teufel and M. Moens, "Summarizing scientific articles: experiments with relevance and rhetorical status," *Computational linguistics*, vol. 28, no. 4, pp. 409–445, 2002.

[77] V. Qazvinian, D. R. Radev, S. M. Mohammad, B. Dorr, D. Zajic, M. Whidby, and T. Moon, "Generating extractive summaries of scientific paradigms," *arXiv preprint arXiv:1402.0556*, 2014.

[78] S. Soares, B. Martins, and P. Calado, "Extracting biographical sentences from textual documents," in *Proceedings of the 15th Portuguese Conference on Artificial Intelligence (EPIA 2011), Lisbon, Portugal*, 2011, pp. 718–30.

[79] L. Zhou, M. Ticrea, and E. Hovy, "Multi-document biography summarization," *arXiv preprint cs/0501078*, 2005.

[80] B. Schiffman, I. Mani, and K. J. Concepcion, "Producing biographical summaries: Combining linguistic knowledge with corpus statistics," in *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*. Association for Computational Linguistics, 2001, pp. 458–465.

[81] I. Mani, G. Klein, D. House, L. Hirschman, T. Firmin, and B. Sundheim, "Summac: a text summarization evaluation," *Natural Language Engineering*, vol. 8, no. 01, pp. 43–68, 2002.

[82] P. Over, H. Dang, and D. Harman, "Duc in context," *Inf. Process. Manage.*, vol. 43, no. 6, pp. 1506–1520, Nov. 2007. [Online]. Available: http://dx.doi.org/10.1016/j.ipm.2007.01.019

[83] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out: Proceedings of the ACL-04 Workshop*, 2004, pp. 74–81.

[84] S. Pouriyeh, M. Allahyari, K. Kochut, G. Cheng, and H. R. Arabnia, "Es-lda: Entity summarization using knowledge-based topic modeling," in *International Joint Conference on Natural Language Processing (IJCNLP)*, 2017.