Chapter 3

A SURVEY OF TEXT SUMMARIZATION TECHNIQUES

Ani Nenkova
University of Pennsylvania
nenkova@seas.upenn.edu

Kathleen McKeown
Columbia University
kathy@cs.columbia.edu

Abstract Numerous approaches for identifying important content for automatic text summarization have been developed to date. Topic representation approaches first derive an intermediate representation of the text that captures the topics discussed in the input. Based on these representations of topics, sentences in the input document are scored for importance. In contrast, in indicator representation approaches, the text is represented by a diverse set of possible indicators of importance which do not aim at discovering topicality. These indicators are combined, very often using machine learning techniques, to score the importance of each sentence. Finally, a summary is produced by selecting sentences in a greedy approach, choosing the sentences that will go in the summary one by one, or globally optimizing the selection, choosing the best set of sentences to form a summary. In this chapter we give a broad overview of existing approaches based on these distinctions, with particular attention on how representation, sentence scoring or summary selection strategies alter the overall performance of the summarizer. We also point out some of the peculiarities of the task of summarization which have posed challenges to machine learning approaches for the problem, and some of the suggested solutions[1].

[1]Portions of this chapter have already appeared in our more detailed overview of summarization research [67]. The larger manuscript includes sections on generation techniques for

C.C. Aggarwal and C.X. Zhai (eds.), *Mining Text Data*, DOI 10.1007/978-1-4614-3223-4_3,

Keywords: Extractive text summarization, topic representation, machine learning for summarization

1. How do Extractive Summarizers Work?

Summarization systems need to produce a concise and fluent summary conveying the key information in the input. In this chapter we constrain our discussion to extractive summarization systems for short, paragraph-length summaries and explain how these systems perform summarization. These summarizers identify the most important sentences in the input, which can be either a single document or a cluster of related documents, and string them together to form a summary. The decision about what content is important is driven primarily by the input to the summarizer.

The choice to focus on extractive techniques leaves out the large body of text-to-text generation approaches developed for abstractive summarization, but allows us to focus on some of the most dominant approaches which are easily adapted to take users' information need into account and work for both single- and multi-document inputs. Moreover, by examining the stages in the operation of extractive summarizers we are able to point out commonalities and differences in summarization approaches which relate to critical components of a system and could explain the advantages of certain techniques over others.

In order to better understand the operation of summarization systems and to emphasize the design choices system developers need to make, we distinguish three relatively independent tasks performed by virtually all summarizers: creating an intermediate representation of the input which captures only the key aspects of the text, scoring sentences based on that representation and selecting a summary consisting of several sentences.

Intermediate representation Even the simplest systems derive some intermediate representation of the text they have to summarize and identify important content based on this representation. *Topic representation* approaches convert the text to an intermediate representation interpreted as the topic(s) discussed in the text. Some of the most popular summarization methods rely on topic representations and this class of approaches exhibits an impressive variation in sophistication and representation power. They include frequency, TF.IDF and topic word approaches in which the topic representation consists of a simple table

summarization, evaluation issues and genre specific summarization which we do not address in this chapter. http://dx.doi.org/10.1561/1500000015

of words and their corresponding weights, with more highly weighted words being more indicative of the topic; lexical chain approaches in which a thesaurus such as WordNet is used to find topics or concepts of semantically related words and then give weight to the concepts; latent semantic analysis in which patterns of word co-occurrence are identified and roughly construed as topics, as well as weights for each pattern; full blown Bayesian topic models in which the input is represented as a mixture of topics and each topic is given as a table of word probabilities (weights) for that topic. *Indicator representation* approaches represent each sentence in the input as a list of indicators of importance such as sentence length, location in the document, presence of certain phrases, etc. In graph models, such as LexRank, the entire document is represented as a network of inter-related sentences.

Score sentences Once an intermediate representation has been derived, each sentence is assigned a score which indicates its importance. For topic representation approaches, the score is commonly related to how well a sentence expresses some of the most important topics in the document or to what extent it combines information about different topics. For the majority of indicator representation methods, the weight of each sentence is determined by combining the evidence from the different indicators, most commonly by using machine learning techniques to discover indicator weights. In LexRank, the weight of each sentence is derived by applying stochastic techniques to the graph representation of the text.

Select summary sentences Finally, the summarizer has to select the best combination of important sentences to form a paragraph length summary. In the *best n* approaches, the top n most important sentences which combined have the desired summary length are selected to form the summary. In *maximal marginal relevance* approaches, sentences are selected in an iterative greedy procedure. At each step of the procedure the sentence importance score is recomputed as a linear combination between the original importance weight of the sentence and its similarity with already chosen sentences. Sentences that are similar to already chosen sentences are dispreferred. In *global selection* approaches, the optimal collection of sentences is selected subject to constraints that try to maximize overall importance, minimize redundancy, and, for some approaches, maximize coherence.

There are very few inherent dependencies between the three processing steps described above and a summarizer can incorporate any combination of specific choices on how to perform the steps. Changes in the way a specific step is performed can markedly change the performance

of the summarizer, and we will discuss some of the known differences as we introduce the traditional methods.

In ranking the importance of sentences for summaries, other factors also come into play. If we have information about the context in which the summary is generated, this can help in determining importance. Context can take the form of information about user needs, often presented through a query. Context can include the environment in which an input document is situated, such as the links which point to a web page. Another factor which affects sentence ranking is the genre of a document. Whether the input document is a news article, an email thread, a web page or a journal article influences the strategies used to select sentences.

We begin with a discussion of topic representation approaches in Section 2. In these approaches the independence between the methods for deriving the intermediate representation and those for scoring sentences is most clear and we emphasize the range of choices for each as we discuss individual approaches. In Section 3 we discuss approaches that focus attention on the contextual information necessary for determining sentence importance rather than the topic representation itself. We follow with a presentation of indicator representation approaches in Section 4. We then discuss approaches to selecting the sentences of a summary in Section 5 before concluding.

2. Topic Representation Approaches

Topic representation approaches vary tremendously in sophistication and encompass a family of methods for summarization. Here we present some of the most widely applied topic representation approaches, as well as those that have been gaining popularity because of their recent successes.

2.1 Topic Words

In remarkably early work on text summarization [53], Luhn proposed the use of frequency thresholds to identify descriptive words in a document to be summarized, a simple representation of the document's topic. The descriptive words in his approach exclude the most frequent words in the document, which are likely to be determiners, prepositions, or domain specific words, as well as those occurring only a few times A modern statistical version of Luhn's idea applies the log-likelihood ratio test [22] for identification of words that are highly descriptive of the input. Such words have been traditionally called "topic signatures" in the summarization literature [46]. The use of topic signatures words

as representation of the input has led to high performance in selecting important content for multi-document summarization of news [15, 38].

Topic signatures are words that occur often in the input but are rare in other texts, so their computation requires counts from a large collection of documents in addition to the input for summarization. One of the key strengths of the log-likelihood ratio test approach is that it provides a way of setting a threshold to divide all words in the input into either descriptive or not. The decision is made based on a test for statistical significance, to large extent removing the need for the arbitrary thresholds in the original approach.

Information about the frequency of occurrence of words in a large background corpus is necessary to compute the statistic on the basis of which topic signature words are determined. The likelihood of the input I and the background corpus is computed under two assumptions: (H1) that the probability of a word in the input is the same as in the background B or (H2) that the word has a different, higher probability, in the input than in the background.

H1: $P(w|I) = P(w|B) = p$ (w is not descriptive)

H2: $P(w|I) = p_I$ and $P(w|B) = p_B$ and $p_I > p_B$ (w is descriptive)

The likelihood of a text with respect to a given word of interest, w, is computed via the binomial distribution formula. The input and the background corpus are treated as a sequence of words w_i: $w_1 w_2 \ldots w_N$. The occurrence of each word is a Bernoulli trial with probability p of success, which occurs when $w_i = w$. The overall probability of observing the word w appearing k times in the N trials is given by the binomial distribution

$$b(k, N, p) = \binom{N}{k} p^k (1-p)^{N-k} \tag{3.1}$$

For H1, the probability p is computed from the input and the background collection taken together. For H2, p_1 is computed from the input, p_2 from the background, and the likelihood of the entire data is equal to the product of the binomial for the input and that for the background. More specifically, the likelihood ratio is defined as

$$\lambda = \frac{b(k, N, p)}{b(k_I, N_I, p_I).b(k_B, N_B, p_B)} \tag{3.2}$$

where the counts with subscript I are computed only from the input to the summarizer and those with index B are computed over the background corpus.

The statistic equal to $-2 \log \lambda$ has a known statistical distribution (χ^2), which can be used to determine which words are topic signatures.

Topic signature words are those that have a likelihood statistic greater than what one would expect by chance. The probability of obtaining a given value of the statistic purely by chance can be looked up in a χ^2 distribution table; for instance a value of 10.83 can be obtained by chance with probability of 0.001.

The importance of a sentence is computed as the number of topic signatures it contains or as the proportion of topic signatures in the sentence. Both of these sentence scoring functions are based on the same topic representation, the scores they assign to sentences may be rather different. The first approach is likely to score longer sentences higher, simply because they contain more words. The second approach favors density of topic words.

2.2 Frequency-driven Approaches

There are two potential modifications that naturally come to mind when considering the topic words approach. The weights of words in topic representations need not be binary (either 1 or 0) as in the topic word approaches. In principle it would even be beneficial to be able to compare the continuous weights of words and determine which ones are more related to the topic. The approaches we present in this section—word probability and TF.IDF—indeed assign non-binary weights related on the number of occurrences of a word or concept. Research has already shown that the binary weights give more stable indicators of sentence importance than word probability and TF.IDF [34]. Nonetheless we overview these approaches because of their conceptual simplicity and reasonable performance. We also describe the lexical chains approach to determining sentence importance. In contrast to most other approaches, it makes use of WordNet, a lexical database which records semantic relations between words. Based on the information derived from WordNet, lexical chain approaches are able to track the prominence, indicated by frequency, of different topics discussed in the input.

Word probability is the simplest form of using frequency in the input as an indicator of importance[2]. The probability of a word w, $p(w)$ is computed from the input, which can be a cluster of related documents or a single document. It is calculated as the number of occurrences of a word, $c(w)$ divided by the number of all words in the input, N:

[2]Raw frequency would be even simpler, but this measure is too strongly influenced by document length. A word appearing twice in a 10 word document may be important, but not necessarily so in a 1000 word document. Computing word probability makes an adjustment for document length.

$$p(w) = \frac{c(w)}{N} \tag{3.3}$$

SumBasic is one system developed to operationalize the idea of using frequency for sentence selection. It relies only on word probability to calculate importance [94]. For each sentence S_j in the input it assigns a weight equal to the average probability $p(w_i)$ of the content words in the sentence[3], estimated from the input for summarization:

$$Weight(S_j) = \frac{\sum_{w_i \in S_j} p(w_i)}{|\{w_i | w_i \in S_j\}|} \tag{3.4}$$

Then, in a greedy fashion, SumBasic picks the best scoring sentence that contains the word that currently has the highest probability. This selection strategy assumes that at each point when a sentence is selected, a single word—that with highest probability—represents the most important topic in the document and the goal is to select the best sentence that covers this word. After the best sentence is selected, the probability of each word that appears in the chosen sentence is adjusted. It is set to a smaller value, equal to the square of the probability of the word at the beginning of the current selection step, to reflect the fact that the probability of a word occurring twice in a summary is lower than the probability of the word occurring only once. This selection loop is repeated until the desired summary length is achieved.

With continuous weights, there are even greater number of possibilities for defining the sentence scoring function compared to the topic words method: the weights can be summed, multiplied, averaged, etc. In each case the scoring is derived by the same representation but the resulting summarizer performance can vary considerably depending on the choice [68]. The sentence selection strategy of SumBasic is a variation of the maximal marginal relevance strategy, but an approach that optimizes the occurrence of important words globally over the entire summary instead of greedy selection perform better [89]. Word probabilities can serve as the basis for increasingly complex views of summarization [50].

TF*IDF weighting (Term Frequency*Inverse Document Frequency)

The word probability approach relies on a stop word list to eliminate too common words from consideration. Deciding which words to include in a stop list, however, is not a trivial task and assigning TF*IDF weights to words [79, 87] provides a better alternative. This weighting

[3]Sentences that have fewer than 15 content words are assigned weight zero and a stop word list is used to eliminate very common words from consideration.

exploits counts from a background corpus, which is a large collection of documents, normally from the same genre as the document that is to be summarized; the background corpus serves as indication of how often a word may be expected to appear in an arbitrary text.

The only additional information besides the term frequency $c(w)$ that we need in order to compute the weight of a word w which appears $c(w)$ times in the input for summarization is the number of documents, $d(w)$, in a background corpus of D documents that contain the word. This allows us to compute the inverse document frequency:

$$TF * IDF_w = c(w). \log \frac{D}{d(w)} \tag{3.5}$$

In many cases $c(w)$ is divided by the maximum number of occurrences of any word in the document, which normalizes for document length. Descriptive topic words are those that appear often in a document, but are not very common in other documents. Words that appear in most documents will have an IDF close to zero. The TF*IDF weights of words are good indicators of importance, and they are easy and fast to compute. These properties explain why TF*IDF is incorporated in one form or another in most current systems [25, 26, 28–30, 40].

Centroid summarization [73], which has become a popular baseline system, is also built on TF.IDF topic representation. In this approach, an empirically determined threshold is set, and all words with TF.IDF below that threshold are considered to have a weight of zero. In this way the centroid approach is similar to the topic word approach because words with low weight are treated as noise and completely ignored when computing sentence importance. It also resembles the word probability approach because it keeps differential weights (TF.IDF) for all word above the threshold. The sentence scoring function in the centroid method is the sum of weights of the words in it.

Lexical chains [3, 86, 31] and some related approaches represent topics that are discussed throughout a text by exploiting relations between words. They capture semantic similarity between nouns to determine the importance of sentences. The lexical chains approach captures the intuition that topics are expressed using not a single word but instead different related words. For example, the occurrence of the words "car", "wheel", "seat", "passenger" indicates a clear topic, even if each of the words is not by itself very frequent. The approach heavily relies on WordNet [63], a manually compiled thesaurus which lists the different senses of each word, as well as word relationships such as synonymy, antonymy, part-whole and general-specific.

A large part of Barzilay and Elhadad's original work on applying lexical chains for summarization [3] is on new methods for constructing good lexical chains, with emphasis on word sense disambiguation of words with multiple meanings (i.e. the word "bank" can mean a financial institution or the land near a river or lake). They develop an algorithm that improves on previous work by waiting to disambiguate polysemous words until all possible chains for a text have been constructed; word senses are disambiguated by selecting the interpretations with the most connections in the text. Later research further improved both the runtime of the algorithms for building of lexical chains, and the accuracy of word sense disambiguation [86, 31].

Barzilay and Elhadad claim that the most prevalent discourse topic will play an important role in the summary and argue that lexical chains provide a better indication of discourse topic than does word frequency simply because different words may refer to the same topic. They define the strength of a lexical chain by its length, which is equal to the number of words found to be members of the same chain, and its homogeneity, where homogeneity captures the number of distinct lexical items in the chain divided by its length. They build the summary by extracting one sentence for each highly scored chain, choosing the first sentence in the document containing a representative word for the chain.

This strategy for summary selection—one sentence per important topic—is easy to implement but possibly too restrictive. The question that stands out, and which Barzilay and Elhadad raise but do not address, is that maybe for some topics more than one sentence should be included in the summary. Other sentence scoring techniques for lexical chain summarization have not been explored, i.e. sentences that include several of the highly scoring chains may be even more informative about the connection between the discussed topics.

In later work, researchers chose to avoid the problem of word sense disambiguation altogether but still used WordNet to track the frequency of all members of a concept set [82, 102]. Even without sense disambiguation, these approaches were able to derive concepts like {war, campaign, warfare, effort, cause, operation, conflict}, {concern, carrier, worry, fear, scare} or {home, base, source, support, backing}. Each of the individual words in the concept could appear only once or twice in the input, but the concept itself appeared in the document frequently.

The heavy reliance on WordNet is clearly a bottleneck for the approaches above, because success is constrained by the coverage of WordNet. Because of this, robust methods such as latent semantic analysis that do not use a specific static hand-crafted resource have much appeal.

2.3 Latent Semantic Analysis

Latent semantic analysis (LSA) [19] is a robust unsupervised technique for deriving an implicit representation of text semantics based on observed co-occurrence of words. Gong and Liu [33] proposed the use of LSA for single and multi-document generic summarization of news, as a way of identifying important topics in documents without the use of lexical resources such as WordNet.

Building the topic representation starts by filling in a n by m matrix A: each row corresponds to a word from the input (n words) and each column corresponds to a sentence in the input (m sentences). Entry a_{ij} of the matrix corresponds to the weight of word i in sentence j. If the sentence does not contain the word, the weight is zero, otherwise the weight is equal to the TF*IDF weight of the word. Standard techniques for singular value decomposition (SVD) from linear algebra are applied to the matrix A, to represent it as the product of three matrices: $A = U\Sigma V^T$. Every matrix has a representation of this kind and many standard libraries provide a built-in implementation of the decomposition.

Matrix U is a n by m matrix of real numbers. Each column can be interpreted as a topic, i.e. a specific combination of words from the input with the weight of each word in the topic given by the real number. Matrix Σ is diagonal m by m matrix. The single entry in row i of the matrix corresponds to the weight of the "topic", which is the ith column of U. Topics with low weight can be ignored, by deleting the last k rows of U, the last k rows and columns of Σ and the last k rows of V^T. This procedure is called dimensionality reduction. It corresponds to the thresholds employed in the centroid and topic words approaches, and topics with low weight are treated as noise. Matrix V^T is a new representation of the sentences, one sentence per row, each of which is expressed not in terms of words that occur in the sentence but rather in terms of the topics given in U. The matrix $D = \Sigma V^T$ combines the topic weights and the sentence representation to indicate to what extent the sentence conveys the topic, with d_{ij} indicating the weight for topic i in sentence j.

The original proposal of Gong and Liu was to select one sentence for each of the most important topics. They perform dimensionality reduction, retaining only as many topics as the number of sentences they want to include in the summary. The sentence with the highest weight for each of the retained topics is selected to form the summary. This strategy suffers from the same drawback as the lexical chains approach because more than one sentence may be required to convey all information pertinent

to that topic. Later researchers have proposed alternative procedures which have led to improved performance of the summarizer in content selection. One improvement is to use the weight of each topic in order to determine the relative proportion of the summary that should cover the topic, thus allowing for a variable number of sentences per topic. Another improvement was to notice that often sentences that discuss several of the important topics are good candidates for summaries [88]. To identify such sentences, the weight of sentence s_i is set to equal

$$Weight(s_i) = \sqrt{\sum_{j=1}^{m} d_{i,j}^2} \tag{3.6}$$

Further variations of the LSA approach have also been explored [72, 35]. The systems that rely on LSA best exemplify the significance of the procedure for sentence scoring. In the many variants of the algorithm, the topic representation remains the same while the way sentences are scored and chosen varies, directly influencing the performance of the summarizer when selecting important content.

2.4 Bayesian Topic Models

Bayesian models are the most sophisticated approach for topic representation proposed for summarization which has been steadily gaining popularity [18, 36, 97, 11].

The original Bayesian model for multi-document summarization [18, 36], derives several distinct probabilistic distributions of words that appear in the input. One distribution is for general English (G), one for the entire cluster to be summarized (C) and one for each individual document i in that cluster (D_i). Each of G, C and D consist of tables of words and their probabilities, or weights, much like the word probability approach, but the weights are very different in G, C and D: a word with high probability in general English is likely to have (almost) zero weight in the cluster table C. The tables (probability distributions) are derived as a part of a hierarchical topic model [8]. It is an unsupervised model and the only data it requires are several multi-document clusters; the general English weights reflect occurrence of words across most of the input clusters.

The topic model representations are quite appealing because they capture information that is lost in most of the other approaches. They, for example, have an explicit representation of the individual documents that make up the cluster that is to be summarized, while it is customary in other approaches to treat the input to a multi-document summarizer

as one long text, without distinguishing document boundaries. The detailed representation would likely enable the development of better summarizers which conveys the similarities and differences among the different documents that make up the input for multi-document summarization [55, 24, 54]. It is also flexible in the manner in which it derives the general English weights of words, without the need for a pre-determined stop word list, or IDF values from a background corpus.

In addition to the improved representation, the topic models highlight the use of a different sentence scoring procedure: Kullback-Lieber (KL) divergence. The KL divergence between two probability distributions captures the mismatch in probabilities assigned to the same events by the two distributions. In summarization, the events are the occurrence of words. The probability of words in the summary can be computed directly, as the number of times the word occurs divided by the total number of words.

In general the KL divergence of probability distribution Q with respect to distribution P over words w is defined as

$$KL(P||Q) = \sum_{w} P(w) \log \frac{P(w)}{Q(w)} \tag{3.7}$$

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

Sentences are scored and selected in a greedy iterative procedure [36]. In each iteration the best sentence i to be selected in the summary is determined as the one for which the KL divergence between C, the probabilities of words in the cluster to be summarized, and the summary so far, including i, is smallest.

KL divergence is appealing as a way of scoring and selecting sentence in summarization because it truly captures an intuitive notion that good summaries are similar to the input. Thinking about a good summary in this way is not new in summarization [21, 74] but KL provides a way of measuring how the importance of words, given by their probabilities, changes in the summary compared to the input. A good summary would reflect the importance of words according to the input, so the divergence between the two will be low. This intuition has been studied extensively in work on automatic evaluation of content selection in summarization, where another indicator of divergence—Jensen Shannon divergence—has proven superior to KL [45, 52].

Given all this, information theoretic measures for scoring sentences are likely to gain popularity even outside the domain on Bayesian topic model representations. All that is necessary in order to apply a divergence to score the summary is a table with word probabilities. The word probability approaches in the spirit of SumBasic [68] can directly ap-

ply divergence measures to score sentences rather than sum, multiply or average the probabilities of words; other methods that assign weights to words can normalize the weights to get a probability distribution of words. In the next section we will also discuss an approach for summarizing academic articles which uses KL divergence to score sentences.

2.5 Sentence Clustering and Domain-dependent Topics

In multi-document summarization of news, the input by definition consists of several articles, possibly from different sources, on the same topic. Across the different articles there will be sentences that contain similar information. Information that occurs in many of the input documents is likely important and worth selecting in a summary. Of course, verbatim repetition on the sentence level is not that common across sources. Rather, similar sentences can be clustered together [59, 39, 85]. In summarization, cosine similarity is standardly used to measure the similarity between the vector representations of sentences [78].

In this approach, clusters of similar sentences are treated as proxies for topics; clusters with many sentences represent important topic themes in the input. Selecting one representative sentence from each main cluster is one way to produce an extractive summary, while minimizing possible redundancy in the summary.

The sentence clustering approach to multi-document summarization exploits repetition at the sentence level. The more sentences there are in a cluster, the more important the information in the cluster is considered. Below is an example of a sentence cluster from different documents in the input to a multi-document summarizer. All four sentences share common content that should be conveyed in the summary.

S1 PAL was devastated by a pilots' strike in June and by the region's currency crisis.

S2 In June, PAL was embroiled in a crippling three-week pilots' strike.

S3 Tan wants to retain the 200 pilots because they stood by him when the majority of PAL's pilots staged a devastating strike in June.

S4 In June, PAL was embroiled in a crippling three-week pilots' strike.

Constraining each sentence to belong to only one cluster is a distinct disadvantage of the sentence clustering approach, and graph methods for summarization which we discuss in the next section, have proven to exploit the same ideas in a more flexible way.

For domain-specific summarization, however, clustering of sentences from many samples from the domain can give a good indication about the topics that are usually discussed in the domain, and the type of

information that a summary would need to convey. In this case, Hidden Markov Models (HMM) that capture "story flow"—what topics are discussed in what order in the domain— can be trained [5, 28]. These models capitalize on the fact that within a specific domain, information in different texts is presented following a common presentation flow. For example, news articles about earthquakes often first talk about where the earthquake happened, what its magnitude was, then mention human casualties or damage, and finally discuss rescue efforts. Such "story flow" can be learned from multiple articles from the same domain. States in the HMM correspond to topics in the domain, which are discovered via iterative clustering of similar sentences from many articles from the domain of interest. Each state (topic) is characterized by a probability distribution which indicates how likely a given word is to appear in a sentence that discusses the topic. Transitions between states in the model correspond to topic transitions in typical texts. These HMM models do not require any labelled data for training and allow for both content selection and ordering in summarization. The sentences that have highest probability of conveying important topics are selected in the summary.

Even simpler approach to discovering the topics in a specific domain can be applied when there are available samples from the domain that are more structured and contain human-written headings. For example, there are plenty of Wikipedia articles about actors and diseases. Clustering similar section headings, where similarity is defined by cosine similarity for example, will identify the topics discussed in each type of article [80]. The clusters with most headings represent the most common topics, and the most common string in the cluster is used to label it. This procedure discovers for example that when talking about actors, writers most often include information about their biography, early life, career and personal life. Then to summarize web pages returned by a search for a specific actor, the system can create a Wikipedia-like web page on the fly, selecting sentences from the returned results that convey these topics.

3. Influence of Context

In many cases, the summarizer has available additional materials that can help determine the most important topics in the document to be summarized. For example in web page summarization, the augmented input consists of other web pages that have links to the pages that we want to summarize. In blog summarization, the discussion following the blog post is easily available and highly indicative of what parts of the blog post are interesting and important. In summarization of scholarly

papers, later papers that cite the paper to be summarized and the citation sentences in particular, provide a rich context that indicate what sentences in the original paper are important. User interests are often taken into account in query-focused summarization, where the query provides additional context. All of these approaches relying on augmented input have been exploited for summarization.

3.1 Web Summarization

One type of web page context to consider is the text in pages that link to the one that has to be summarized, in particular the text surrounded by the hyperlink tag pointing to the page. This text often provides a descriptive summary of a web page (e.g., "Access to papers published within the last year by members of the NLP group"). Proponents of using context to provide summary sentences argue that a web site includes multimedia, may cover diverse topics, and it may be hard for a summarizer to distinguish good summary content from bad [20]. The earliest work on this approach was carried out to provide snippets for each result from a search engine [2]. To determine a summary, their system issued a search for a URL, selected all sentences containing a link to that URL and the best sentence was identified using heuristics. Later work has extended this approach through an algorithm that allows selection of a sentence that covers as many aspects of the web page as possible and that is on the same topic [20] . For coverage, Delort et al. used word overlap, normalized by sentence length, to determine which sentences are entirely covered by others and thus can be removed from consideration for the summary. To ensure topicality, Delort's system selects a sentence that is a reference to the page (e.g., "CNN is a news site") as opposed to content (e.g., "The top story for today..."). He computes topicality by measuring overlap between each context sentence and the text within the web page, normalizing by the number of words in the web page. When the web page does not have many words, instead he clusters all sentences in the context and chooses the sentence that is most similar to all others using cosine distance.

In summarization of blog posts, important sentences are identified based on word frequency [41]. The critical difference from other approaches is that here frequency is computed over the comments on the post rather then the original blog entry. The extracted sentences are those that elicited discussion.

3.2 Summarization of Scientific Articles

Impact summarization [60] is defined as the task of extracting sentences from a paper that represent the most influential content of that paper. Language models provide a natural way for solving the task. For each paper to be summarized, impact summarization methods find other papers in a large collection that cite that paper and extract the areas in which the references occur. A language model is built using the collection of all reference areas to a paper, giving the probability of each word to occur in a reference area. This language model gives a way of scoring the importance of sentences in the original article: important sentences are those that convey information similar to that which later papers discussed when referring to the original paper. The measure of similarity between a sentence and the language model is measured by KL divergence. In order to account for the importance of each sentence within the summarized article alone, the approach uses word probabilities estimated from the article. The final score of a sentence is a linear combination of impact importance coming from KL divergence and intrinsic importance coming from the word probabilities in the input article.

3.3 Query-focused Summarization

In query-focused summarization, the importance of each sentence will be determined by a combination of two factors: how relevant is that sentence to the user question and how important is the sentence in the context of the input in which it appears. There are two classes of approaches to this problem. The first adapts techniques for generic summarization of news. For example, an approach using topic signature words [15] is extended for query-focused summarization by assuming that the words that should appear in a summary have the following probability: a word has probability zero of appearing in a summary for a user defined topic if it neither appears in the user query nor is a topic signature word for the input; the probability of the word to appear in the summary is 0.5 if it either appears in the user query or is a topic signature, but not both; and the probability of a word to appear in a summary is 1 if it is both in the user query and in the list of topic signature words for the input. These probabilities are arbitrarily chosen, but in fact work well when used to assign weights to sentences equal to the average probability of words in the sentence. Graph-based approaches [71] have also been adapted for query-focused summarization with minor modifications.

Other approaches have been developed that use new methods for identifying relevant and salient sentences. These approaches have usually

been developed for specific types of queries. For example, many people have worked on generation of biographical summaries, where the query is the name of the person for whom a biography should be generated. Most people use some balance of top-down driven approaches that search for patterns of information that might be found in a biography, often using machine learning to identify the patterns, combined with bottom-up approaches that sift through all available material to find sentences that are biographical in nature [7, 98, 81, 105, 6]. The most recent of these approaches uses language modeling of biographical texts found on Wikipedia and non-biographical texts in a news corpus to identify biographical sentences in input documents.

Producing snippets for search engines is a particularly useful query focused application [92, 95].

3.4 Email Summarization

Summarization must be sensitive to the unique characteristics of email, a distinct linguistic genre that exhibits characteristics of both written text and spoken conversation. A thread or a mailbox contains one or more conversations between two or more participants over time. As in summarization of spoken dialog, therefore, summarization needs to take the interactive nature of dialog into account; a response is often only meaningful in relation to the utterance it addresses. Unlike spoken dialog, however, the summarizer need not concern itself with speech recognition errors, the impact of pronunciation, or the availability of speech features such as prosody. Furthermore, responses and reactions are not immediate and due to the asynchronous nature of email, they may explicitly mark the previous email passages to which they are relevant.

In early research on summarization of email threads, [66] used an extractive summarizer to generate a summary for the first two levels of the discussion thread tree, producing relatively short "overview summaries." They extracted a sentence for each of the two levels, using overlap with preceding context. Later work on summarization of email threads [75] zeroed in on the dialogic nature of email. Their summarizer used machine learning and relied on email specific features in addition to traditional features, including features related to the thread and features related to email structure such as the number of responders to a message, similarity of a sentence with the subject, etc. Email conversations are a natural means of getting answers to one's questions and the asynchronous nature of email makes it possible for one to pursue several questions in parallel. As a consequence, question-answer exchanges

figure as one of the dominant uses of email conversations. These observations led to research on identification of question and answer pairs in email [84, 64] and the integration of such pairs in extractive summaries of email [58].

Email summarizers have also been developed for a full mailbox or archive instead of just a thread. [69] present a system that can be used for browsing an email mailbox and that builds upon multi-document summarization techniques. They first cluster all email in topically related threads. Both an overview and a full-length summary are then generated for each cluster.A more recent approach to summarization of email within a folder uses a novel graph-based analysis of quotations within email [10]. Using this analysis, Carenini et al.'s system computes a graph representing how each individual email directly mentions other emails, at the granularity of fragments and sentences.

4. Indicator Representations and Machine Learning for Summarization

Indicator representation approaches do not attempt to interpret or represent the topics discussed in the input. Instead they come up with a representation of the text that can be used to directly rank sentences by importance. Graph methods are unique because in their most popular formulations they base summarization on a single indicator of importance, derived from the centrality of sentences in a graph representation of the input. In contrast other approaches employ a variety of indicators and combine them either heuristically or using machine learning to decide which sentences are worthy to be included in the summary.

4.1 Graph Methods for Sentence Importance

In the graph models inspired by the PageRank algorithm [25, 61], the input is represented as a highly connected graph. Vertices represent sentences and edges between sentences are assigned weights equal to the similarity between the two sentences. The method most often used to compute similarity is cosine similarity with TF*IDF weights for words. Sometimes, instead of assigning weights to edges, the connections between vertices can be determined in a binary fashion: the vertices are connected only if the similarity between the two sentences exceeds a predefined threshold. Sentences that are related to many other sentences are likely to be central and would have high weight for selection in the summary.

When the weights of the edges are normalized to form a probability distribution so that the weight of all outgoing edges from a given vertex

sum up to one, the graph becomes a Markov chain and the edge weights correspond to the probability of transitioning from one state to another. Standard algorithms for stochastic processes can be used to compute the probability of being in each vertex of the graph at time t while making consecutive transitions from one vertex to next. As more and more transitions are made, the probability of each vertex converges, giving the stationary distribution of the chain. The stationary distribution gives the probability of (being at) a given vertex and can be computed using iterative approximation. Vertices with higher probabilities correspond to more important sentences that should be included in the summary.

Graph-based approaches have been shown to work well for both single-document and multi-document summarization [25, 61]. Since the approach does not require language-specific linguistic processing beyond identifying sentence and word boundaries, it can also be applied to other languages, for example, Brazilian Portuguese [62]. At the same time, incorporating syntactic and semantic role information in the building of the text graph leads to superior results over plain TF*IDF cosine similarity [13].

Using different weighting schemes for links between sentences that belong to the same article and sentences from different articles can help separate the notions of topicality within a document and recurrent topics across documents. This distinction can be easily integrated in the graph-based models for summarization [96].

Graph representations for summarization had been explored even before the PageRank models became popular. For example, the purpose of an older graph-based system for multi-document summarization [55] is to identify salient regions of each story related to a topic given by a user, and compare the stories by summarizing similarities and differences. The vertices in the graph are *words*, *phrases* and *named entities* rather than sentences and their initial weight is assigned using TF*IDF. Edges between vertices are defined using synonym and hypernym links in WordNet, as well as coreference links. Spreading activation is used to assign weights to non-query terms as a function of the weight of their neighbors in the graph and the type of relation connecting the nodes.

In order to avoid problems with coherence that may arise with the selection of single sentences, the authors of another approach [78] argue that a summarizer should select full paragraphs to provide adequate context. Their algorithm constructs a text graph for a document using cosine similarity between each pair of paragraphs in the document. The shape of the text graph determines which paragraphs to extract. In their experiments, they show that two strategies, selecting paragraphs

that are well connected to other paragraphs or first paragraphs of topical text segments within the graph, both produce good summaries.

A combination of the subsentential granularity of analysis where nodes are words and phrases rather than sentences and edges are syntactic dependencies has also been explored [44]. Using machine learning techniques, the authors attempt to learn what portions of the input graph would be included in a summary. In their experiments on single document summarization of news articles, properties of the graph such as incoming and outgoing links, connectivity and PageRank weights are identified as the best class of features that can be used for content selection. This work provides an excellent example of how machine learning can be used to combine a range of indicators of importance rather than committing to a single one.

4.2 Machine Learning for Summarization

Edmundson's early work [23] set the direction for later investigation of applying machine learning techniques for summarization [43]. He proposed that rather than relying on a single representation of topics in the input, many different indicators of importance can be combined. Then a corpus of inputs and summaries written by people can be used to determine the weight of each indicator.

In supervised methods for summarization, the task of selecting important sentences is represented as a binary classification problem, partitioning all sentences in the input into summary and non-summary sentences. A corpus with human annotations of sentences that should be included in the summary is used to train a statistical classifier for the distinction, with each sentences represented as a list of potential indicators of importance. The likelihood of a sentence to belong to the summary class, or the confidence of the classifier that the sentence should be in the summary, is the score of the sentence. The chosen classifier plays the role of a sentence scoring function, taking as an input the intermediate representation of the sentence and outputting the score of the sentence. The most highly scoring sentences are selected to form the summary, possibly after skipping some because of high similarity to already chosen sentences.

Machine learning approaches to summarization offer great freedom because the number of indicators of importance is practically endless [40, 70, 104, 44, 27, 37, 99, 51]. Any of the topic representation approaches discussed above can serve as the basis of indicators. Some common features include the position of the sentence in the document (first sentences of news are almost always informative), position in the

paragraph (first and last sentences are often important), sentence length, similarity of the sentence with the document title or headings, weights of the words in a sentence determined by any topic representation approach, presence of named entities or cue phrases from a predetermined list, etc.

It is hardly an exaggeration to say that every existing machine learning method has been applied for summarization. One important difference is whether the classifier assumes that the decision about inclusion in the summary is independently done for each sentence. This assumption is apparently not realistic, and methods that explicitly encode dependencies between sentences such as Hidden Markov Models and Conditional Random Fields outperform other learning methods [14, 30, 83].

A problem inherent in the supervised learning paradigm is the necessity of labeled data on which classifiers can be trained. Asking annotators to select summary-worthy sentences is a reasonable solution [93] but it is time consuming and even more importantly, annotator agreement is low and different people tend to choose different sentences when asked to construct an extractive summary of a text [76]. Partly motivated by this issue and partly because of their interest in ultimately developing abstractive methods for summarization many researchers have instead worked with abstracts written by people (often professional writers). Researchers concentrated their efforts on developing methods for automatic alignment of the human abstracts and the input [56, 42, 104, 4, 17] in order to provide labeled data of summary and non-summary sentences for machine learning. Some researchers have also proposed ways to leverage the information from manual evaluation of content selection in summarization in which multiple sentences can be marked as expressing the same fact that should be in the summary [16, 27]. Alternatively, one could compute similarity between sentences in human abstracts and those in the input in order to find very similar sentences, not necessarily doing full alignment [12].

Another option for training a classifier is to employ a semi-supervised approach. In this paradigm, a small number of examples of summary and non-summary sentences are annotated by people. Then two classifiers are trained on that data, using different sets of features which are independent given the class [100] or two different classification methods [99]. After that one of the classifiers is run on unannotated data, and its most confident predictions are added to the annotated examples to train the other classifier, repeating the process until some predefined halting condition is met.

Several modifications to standard machine learning approaches are appropriate for summarization. In effect formulating summarization as

a binary classification problem, which scores individual sentences, is not equivalent to finding the best summary, which consists of several sentences. This is exactly the issue of selecting a summary that we discuss in the next section. In training a supervised model, the parameters may be optimized to lead to a summary that has the best score against a human model [1, 49].

For generic multi-document summarization of news, supervised methods have not been shown to outperform competitive unsupervised methods based on a single feature such as the presence of topic words and graph methods. Machine learning approaches have proved to be much more successful in single document or domain or genre specific summarization, where classifiers can be trained to identify specific types of information such as sentences describing literature background in scientific article summarization [90], utterances expressing agreement or disagreement in meetings [30], biographical information [105, 6, 80], etc.

5. Selecting Summary Sentences

Most summarization approaches choose content sentence by sentence: they first include the most informative sentence, and then if space constraints permit, the next most informative sentence is included in the summary and so on. Some process of checking for similarity between the chosen sentences is also usually employed in order to avoid the inclusion of repetitive sentences.

5.1 Greedy Approaches: Maximal Marginal Relevance

indexGreedy Approach to Summarization One of the early summarization approaches for both generic and query focused summarization that has been widely adopted is *Maximal Marginal Relevance* (MMR) [9]. In this approach, summaries are created using greedy, sentence-by-sentence selection. At each selection step, the greedy algorithm is constrained to select the sentence that is maximally relevant to the user query (or has highest importance score when a query is not available) and minimally redundant with sentences already included in the summary. MMR measures relevance and novelty separately and then uses a linear combination of the two to produce a single score for the importance of a sentence in a given stage of the selection process. To quantify both properties of a sentence, Carbonell and Goldstein use cosine similarity. For relevance, similarity is measured to the query, while for novelty, similarity is measured against sentences selected so far. The MMR approach was originally proposed for query-focused summarization in the

context of information retrieval, but could easily be adapted for generic summarization, for example by using the entire input as a user [33]. In fact any of the previously discussed approaches for sentence scoring can be used to calculate the importance of a sentence. Many have adopted this seminal approach, mostly in its generic version, sometimes using different measures of novelty to select new sentences [91, 101, 65].

This greedy approach of sequential sentence selection might not be that effective for optimal content selection of the entire summary. One typical problematic scenario for greedy sentence selection (discussed in [57]) is when a very long and highly relevant sentence happens to be evaluated as the most informative early on. Such a sentence may contain several pieces of relevant information, alongside some not so relevant facts which could be considered noise. Including such a sentence in the summary will help maximize content relevance at the time of selection, but at the cost of limiting the amount of space in the summary remaining for other sentences. In such cases it is often more desirable to include several shorter sentences, which are individually less informative than the long one, but which taken together do not express any unnecessary information.

5.2 Global Summary Selection

Global optimization algorithms can be used to solve the new formulation of the summarization task, in which the best overall summary is selected. Given some constraints imposed on the summary, such as maximizing informativeness, minimizing repetition, and conforming to required summary length, the task would be to select the best summary. Finding an exact solution to this problem is NP-hard [26], but approximate solutions can be found using a dynamic programming algorithm [57, 103, 102]. Exact solutions can be found quickly via search techniques when the sentence scoring function is local, computable only from the given sentence [1].

Even in global optimization methods, informativeness is still defined and measured using features well-explored in the sentence selection literature. These include word frequency and position in the document [103], TF*IDF [26], similarity with the input [57], and concept frequency [102, 32]. Global optimization approaches to content selection have been shown to outperform greedy selection algorithms in several evaluations using news data as input, and have proved to be especially effective for extractive summarization of meetings [77, 32].

In a detailed study of global inference algorithms [57], it has been demonstrated that it is possible to find an exact solution for the op-

timization problem for content selection using Integer Linear Programming. The performance of the approximate algorithm based on dynamic programming was lower, but comparable to that of the exact solutions. In terms of running time, the greedy algorithm is very efficient, almost constant in the size of the input. The approximate algorithm scales linearly with the size of the input and is thus indeed practical to use. The running time for the exact algorithm grows steeply with the size of the input and is unlikely to be useful in practice [57]. However, when a monotone submodular function is used to evaluate the informativeness of the summary, optimal or near optimal solution can be found quickly [48, 47].

6. Conclusion

In this chapter we have attempted to give a comprehensive overview of the most prominent recent methods for automatic text summarization. We have outlined the connection to early approaches and have contrasted approaches in terms of how they represent the input, score sentences and select the summary. We have highlighted the success of KL divergence as a method for scoring sentences which directly incorporates an intuition about the characteristics of a good summary, as well as the growing interest in the development of methods that globally optimize the selection of the summary. We have shown how summarization strategies must be adapted to different genres, such as web pages and journal articles, taking into account contextual information that guides sentence selection. These three recent developments in summarization complement traditional topics in the field that concern intermediate representations and the application of appropriate machine learning methods for summarization.

References

[1] A. Aker, T. Cohn, and R. Gaizauskas. Multi-document summarization using a* search and discriminative training. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP'10, pages 482–491, 2010.

[2] E. Amitay and C. Paris. Automatically summarizing web sites - is there a way around it? In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 173–179, 2000.

[3] R. Barzilay and M. Elhadad. Text summarizations with lexical chains. In Inderjeet Mani and Mark Maybury, editors, *Advances in Automatic Text Summarization*, pages 111–121. MIT Press, 1999.

[4] R. Barzilay and N. Elhadad. Sentence alignment for monolingual comparable corpora. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 25–32, 2003.

[5] R. Barzilay and L. Lee. Catching the drift: Probabilistic content models, with applications to generation and summarization. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 113–120, 2004.

[6] F. Biadsy, J. Hirschberg, and E. Filatova. An unsupervised approach to biography production using wikipedia. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 807–815, 2008.

[7] S. Blair-Goldensohn, K. McKeown, and A. Schlaikjer. Defscriber: a hybrid system for definitional qa. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 462–462, 2003.

[8] D. Blei, T. Griffiths, M. Jordan, and J. Tenenbaum. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, page 2003, 2004.

[9] J. Carbonell and J. Goldstein. The use of mmr, diversity-based rerunning for reordering documents and producing summaries. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 335–336, 1998.

[10] G. Carenini, R. Ng, and X. Zhou. Summarizing email conversations with clue words. In *Proceedings of the international conference on World Wide Web*, pages 91–100, 2007.

[11] 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*, pages 815–824, 2010.

[12] Y. Chali, S. Hasan, and S. Joty. Do automatic annotation techniques have any impact on supervised complex question answering? In *Proceedings of the Joint Conference of the Annual Meeting of the ACL and the International Joint Conference on Natural Language Processing of the AFNLP*, pages 329–332, 2009.

[13] Y. Chali and S. Joty. Improving the performance of the random walk model for answering complex questions. In *Proceedings of the*

Annual Meeting of the Association for Computational Linguistics, Short Papers, pages 9–12, 2008.

[14] J. Conroy and D. O'Leary. Text summarization via hidden markov models. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 406–407, 2001.

[15] J. Conroy, J. Schlesinger, and D. O'Leary. Topic-focused multi-document summarization using an approximate oracle score. In *Proceedings of the International Conference on Computational Linguistics and the annual meeting of the Association for Computational Linguistics*, pages 152–159, 2006.

[16] T. Copeck and S. Szpakowicz. Leveraging pyramids. In *Proceedings of the Document Understanding Conference*, 2005.

[17] H. Daumé III and D. Marcu. A phrase-based HMM approach to document/abstract alignment. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 119–126, 2004.

[18] H. Daumé III and D. Marcu. Bayesian query-focused summarization. In *Proceedings of the International Conference on Computational Linguistics and the annual meeting of the Association for Computational Linguistics*, pages 305–312, 2006.

[19] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, pages 391–407, 1990.

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

[21] R. Donaway, K. Drummey, and L. Mather. A comparison of rankings produced by summarization evaluation measures. In *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization - Volume 4*, pages 69–78, 2000.

[22] T. Dunning. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74, 1994.

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

[24] N. Elhadad, M.-Y. Kan, J. Klavans, and K. McKeown. Customization in a unified framework for summarizing medical literature. *Journal of Artificial Intelligence in Medicine*, 33:179–198, 2005.

[25] G. Erkan and D. Radev. Lexrank: Graph-based centrality as salience in text summarization. *Journal of Artificial Intelligence Research*, 2004.

[26] E. Filatova and V. Hatzivassiloglou. A formal model for information selection in multi-sentence text extraction. In *Proceedings of the International Conference on Computational Linguistic*, pages 397–403, 2004.

[27] M. Fuentes, E. Alfonseca, and H. Rodríguez. Support vector machines for query-focused summarization trained and evaluated on pyramid data. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, Companion Volume: Proceedings of the Demo and Poster Sessions*, pages 57–60, 2007.

[28] P. Fung and G. Ngai. One story, one flow: Hidden markov story models for multilingual multidocument summarization. *ACM Transactions on Speech and Language Processing*, 3(2):1–16, 2006.

[29] S. Furui, M. Hirohata, Y. Shinnaka, and K. Iwano. Sentence extraction-based automatic speech summarization and evaluation techniques. In *Proceedings of the Symposium on Large-scale Knowledge Resources*, pages 33–38, 2005.

[30] M. Galley. A skip-chain conditional random field for ranking meeting utterances by importance. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 364–372, 2006.

[31] M. Galley and K. McKeown. Improving word sense disambiguation in lexical chaining. In *Proceedings of the international joint conference on Artificial intelligence*, pages 1486–1488, 2003.

[32] D. Gillick, K. Riedhammer, B. Favre, and D. Hakkani-Tur. A global optimization framework for meeting summarization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4769–4772, 2009.

[33] Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 19–25, 2001.

[34] S. Gupta, A. Nenkova, and D. Jurafsky. Measuring importance and query relevance in topic-focused multi-document summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, Demo and Poster Sessions*, pages 193–196, 2007.

[35] B. Hachey, G. Murray, and D. Reitter. Dimensionality reduction aids term co-occurrence based multi-document summarization. In *SumQA '06: Proceedings of the Workshop on Task-Focused Summarization and Question Answering*, pages 1–7, 2006.

[36] A. Haghighi and L. Vanderwende. Exploring content models for multi-document summarization. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370, 2009.

[37] D. Hakkani-Tur and G. Tur. Statistical sentence extraction for information distillation. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages IV–1 –IV–4, 2007.

[38] 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*, SIGIR'05, pages 202–209, 2005.

[39] V. Hatzivassiloglou, J. Klavans, M. Holcombe, R. Barzilay, M. Kan, and K. McKeown. Simfinder: A flexible clustering tool for summarization. In *Proceedings of the NAACL Workshop on Automatic Summarization*, pages 41–49, 2001.

[40] E. Hovy and C.-Y. Lin. Automated text summarization in summarist. In *Advances in Automatic Text Summarization*, pages 82–94, 1999.

[41] M. Hu, A. Sun, and E.-P. Lim. Comments-oriented blog summarization by sentence extraction. In *Proceedings of the ACM Conference on Information and Knowledge Management*, pages 901–904, 2007.

[42] H. Jing. Using hidden markov modeling to decompose human-written summaries. *Computational linguistics*, 28(4):527–543, 2002.

[43] J. Kupiec, J. Pedersen, and F. Chen. A trainable document summarizer. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 68–73, 1995.

[44] J. Leskovec, N. Milic-frayling, and M. Grobelnik. Impact of linguistic analysis on the semantic graph coverage and learning of document extracts. In *Proceedings of the national conference on Artificial intelligence*, pages 1069–1074, 2005.

[45] C.-Y. Lin, G. Cao, J. Gao, and J.-Y. Nie. An information-theoretic approach to automatic evaluation of summaries. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics (HLT-NAACL'06)*, pages 463–470, 2006.

[46] C.-Y. Lin and E. Hovy. The automated acquisition of topic signatures for text summarization. In *Proceedings of the International Conference on Computational Linguistic*, pages 495–501, 2000.

[47] H. Lin and J. Bilmes. Multi-document summarization via budgeted maximization of submodular functions. In *North American chapter of the Association for Computational Linguistics/Human Language Technology Conference (NAACL/HLT-2010)*, 2010.

[48] H. Lin, J. Bilmes, and S. Xie. Graph-based submodular selection for extractive summarization. In *Proc. IEEE Automatic Speech Recognition and Understanding (ASRU)*, 2009.

[49] S.-H. Lin, Y.-M. Chang, J.-W. Liu, and B. Chen. Leveraging evaluation metric-related training criteria for speech summarization. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010*, pages 5314–5317, 2010.

[50] S.-H. Lin and B. Chen. A risk minimization framework for extractive speech summarization. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 79–87, 2010.

[51] A. Louis, A. Joshi, and A. Nenkova. Discourse indicators for content selection in summarization. In *Proceedings of the Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 147–156, 2010.

[52] A. Louis and A. Nenkova. Automatically evaluating content selection in summarization without human models. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 306–314, 2009.

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

[54] M. Mana-López, M. De Buenaga, and J. Gómez-Hidalgo. Multidocument summarization: An added value to clustering in interactive retrieval. *ACM Transactions on Informations Systems*, 22(2):215–241, 2004.

[55] I. Mani and E. Bloedorn. Summarizing similarities and differences among related documents. *Information Retrieval*, 1(1-2):35–67, April 1999.

[56] D. Marcu. The automatic construction of large-scale corpora for summarization research. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 137–144, 1999.

[57] R. McDonald. A study of global inference algorithms in multi-document summarization. In *Proceedings of the European Conference on IR Research*, pages 557–564, 2007.

[58] K. McKeown, L. Shrestha, and O. Rambow. Using question-answer pairs in extractive summarization of email conversations. In *Proceedings of the International Conference on Computational Linguistics and Intelligent Text Processing*, pages 542–550, 2007.

[59] K. McKeown, J. Klavans, V. Hatzivassiloglou, R. Barzilay, and E. Eskin. Towards multidocument summarization by reformulation: progress and prospects. In *Proceedings of the national conference on Artificial intelligence*, pages 453–460, 1999.

[60] Q. Mei and C. Zhai. Generating impact-based summaries for scientific literature. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 816–824, 2008.

[61] R. Mihalcea and P. Tarau. Textrank: Bringing order into texts. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 404–411, 2004.

[62] R. Mihalcea and P. Tarau. An algorithm for language independent single and multiple document summarization. In *Proceedings of the International Joint Conference on Natural Language Processing*, pages 19–24, 2005.

[63] G.A. Miller, R. Beckwith, C. Fellbaum, D. Gross, and K. J. Miller. Introduction to wordnet: An on-line lexical database. *International Journal of Lexicography (special issue)*, 3(4):235–312, 1990.

[64] H. Murakoshi, A. Shimazu, and K. Ochimizu. Construction of deliberation structure in email conversation. In *Proceedings of the Conference of the Pacific Association for Computational Linguistics*, pages 570–577, 2004.

[65] G. Murray, S. Renals, and J. Carletta. Extractive summarization of meeting recordings. In *Proc. 9th European Conference on Speech Communication and Technology*, pages 593–596, 2005.

[66] A. Nenkova and A. Bagga. Facilitating email thread access by extractive summary generation. In *Proceedings of the Recent Advances in Natural Language Processing Conference*, 2003.

[67] A. Nenkova and K. McKeown. Automatic Summarization. In *Foundations and Trends in Information Retrieval* 5(2–3), pages 103–233, 2011.

[68] A. Nenkova, L. Vanderwende, and K. McKeown. A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 573–580, 2006.

[69] P. Newman and J. Blitzer. Summarizing archived discussions: a beginning. In *Proceedings of the international conference on Intelligent user interfaces*, pages 273–276, 2003.

[70] M. Osborne. Using maximum entropy for sentence extraction. In *Proceedings of the ACL Workshop on Automatic Summarization*, pages 1–8, 2002.

[71] J. Otterbacher, G. Erkan, and D. Radev. Biased lexrank: Passage retrieval using random walks with question-based priors. *Information Processing and Management*, 45:42–54, January 2009.

[72] M. Ozsoy, I. Cicekli, and F. Alpaslan. Text summarization of turkish texts using latent semantic analysis. In *Proceedings of the 23rd International Conference on Computational Linguistics (COLING 2010)*, pages 869–876, August 2010.

[73] D. Radev, H. Jing, M. Sty, and D. Tam. Centroid-based summarization of multiple documents. *Information Processing and Management*, 40:919–938, 2004.

[74] D. Radev, S. Teufel, H. Saggion, W. Lam, J. Blitzer, H. Qi, A. Çelebi, D. Liu, and E. Drabek. Evaluation challenges in large-scale document summarization. In *Proceedings of the 41st Annual Meeting on Association for Computational Linguistics (ACL'03)*, pages 375–382, 2003.

[75] O. Rambow, L. Shrestha, J. Chen, and C. Lauridsen. Summarizing email threads. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, 2004.

[76] G. Rath, A. Resnick, and R. Savage. The formation of abstracts by the selection of sentences: Part 1: sentence selection by man and machines. *American Documentation*, 2(12):139–208, 1961.

[77] K. Riedhammer, D. Gillick, B. Favre, and D. Hakkani-Tur. Packing the meeting summarization knapsack. In *Proceedings of the Annual Conference of the International Speech Communication Association*, pages 2434–2437, 2008.

[78] G. Salton, A. Singhal, M. Mitra, and C. Buckley. Automatic text structuring and summarization. *Information Processing and Management*, 33(2):193–208, 1997.

[79] G. Salton and C. Buckley. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24:513–523, 1988.

[80] C. Sauper and R. Barzilay. Automatically generating wikipedia articles: A structure-aware approach. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 208–216, 2009.

[81] B. Schiffman, I. Mani, and K. Concepcion. Producing biographical summaries: Combining linguistic knowledge with corpus statistics. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 458–465, 2001.

[82] B. Schiffman, A. Nenkova, and K. McKeown. Experiments in multidocument summarization. In *Proceedings of the international conference on Human Language Technology Research*, pages 52–58, 2002.

[83] D. Shen, J.-T. Sun, H. Li, Q. Yang, and Z. Chen. Document summarization using conditional random fields. In *Proceedings of the 20th international joint conference on Artifical intelligence*, pages 2862–2867, 2007.

[84] L. Shrestha and K. McKeown. Detection of question-answer pairs in email conversations. In *Proceedings of the International Conference on Computational Linguistic*, 2004.

[85] A. Siddharthan, A. Nenkova, and K. McKeown. Syntactic simplification for improving content selection in multi-document summarization. In *Proceedings of the International Conference on Computational Linguistic*, pages 896–902, 2004.

[86] H. Silber and K. McCoy. Efficiently computed lexical chains as an intermediate representation for automatic text summarization. *Computational Linguistics*, 28(4):487–496, 2002.

[87] K. Sparck Jones. A statistical interpretation of term specificity and its application in retrieval. *Journal of Documentation*, 28:11–21, 1972.

[88] J. Steinberger, M. Poesio, M. A. Kabadjov, and K. Jeek. Two uses of anaphora resolution in summarization. *Information Processing and Management*, 43(6):1663–1680, 2007.

[89] W. Yih, J. Goodman, L. Vanderwende, and H. Suzuki. Multi-document summarization by maximizing informative content-words. In *Proceedings of the international joint conference on Artificial intelligence*, pages 1776–1782, 2007.

[90] S. Teufel and M. Moens. Summarizing scientific articles: experiments with relevance and rhetorical status. *Computational Linguisics.*, 28(4):409–445, 2002.

[91] D. Radev, T. Allison, S. Blair-goldensohn, J. Blitzer, A. Celebi, S. Dimitrov, E. Drabek, A. Hakim, W. Lam, D. Liu, J. Otterbacher, H. Qi, H. Saggion, S. Teufel, A. Winkel, and Z. Zhang. Mead - a platform for multidocument multilingual text summarization. In *Proceedings of the International Conference on Language Resources and Evaluation*, 2004.

[92] A. Turpin, Y. Tsegay, D. Hawking, and H. Williams. Fast generation of result snippets in web search. In *Proceedings of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 127–134, 2007.

[93] J. Ulrich, G. Murray, and G. Carenini. A publicly available annotated corpus for supervised email summarization. In *Proceedings of the AAAI EMAIL Workshop*, pages 77–87, 2008.

[94] L. Vanderwende, H. Suzuki, C. Brockett, and A. Nenkova. Beyond sumbasic: Task-focused summarization with sentence simplification and lexical expansion. *Information Processing and Managment*, 43:1606–1618, 2007.

[95] R. Varadarajan and V. Hristidis. A system for query-specific document summarization. In *Proceedings of the ACM Conference on Information and Knowledge Management*, 2006.

[96] X. Wan and J. Yang. Improved affinity graph based multi-document summarization. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 181–184, 2006.

[97] 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*, pages 297–300, 2009.

[98] R. Weischedel, J. Xu, and A. Licuanan. A hybrid approach to answering biographical questions. In Mark Maybury, editor, *New Directions In Question Answering*, pages 59–70, 2004.

[99] K. Wong, M. Wu, and W. Li. Extractive summarization using supervised and semi-supervised learning. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 985–992, 2008.

[100] S. Xie, H. Lin, and Y. Liu. Semi-supervised extractive speech summarization via co-training algorithm. In *INTERSPEECH, the 11th Annual Conference of the International Speech Communication Association*, pages 2522–2525, 2010.

[101] S. Xie and Y. Liu. Using corpus and knowledge-based similarity measure in maximum marginal relevance for meeting summarization. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 4985–4988, 2008.

[102] S. Ye, T.-S. Chua, M.-Y. Kan, and L. Qiu. Document concept lattice for text understanding and summarization. *Information Processing and Management*, 43(6):1643 – 1662, 2007.

[103] W. Yih, J. Goodman, L. Vanderwende, and H. Suzuki. Multi-document summarization by maximizing informative content-words. In *Proceedings of the international joint conference on Artificial intelligence*, pages 1776–1782, 2007.

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

[105] L. Zhou, M. Ticrea, and E. Hovy. Multi-document biography summarization. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 434–441, 2004.