Beyond Centrality and Structural Features: Learning Information Importance for Text Summarization

Markus Zopf, Eneldo Loza Mencía, and Johannes Fürnkranz
Research Training Group AIPHES / Knowledge Engineering Group
Department of Computer Science, Technische Universität Darmstadt
Hochschulstraße 10, 64289 Darmstadt, Germany
{zopf@aiphes,eneldo@ke,juffi@ke}.tu-darmstadt.de

Abstract

Most automatic text summarization systems proposed to date rely on centrality and structural features as indicators for information importance. In this paper, we argue that these features cannot reliably detect important information in heterogeneous document collections. Instead, we propose *CPSum*, a summarizer that learns the importance of information objects from a background source. Our hypothesis is tested on a multi-document corpus where we remove centrality and structural features. *CPSum* proves to be able to perform well in this challenging scenario whereas reference systems fail.

1 Introduction

The goal of text summarization is to *take an information source, extract content from it, and present the most important content to the user [...]* (Mani, 2001). Identifying important information in source documents is therefore a major goal in summarization. Most methods to date rely on structural features such as sentence position, number of upper-case words, or title words, and a wide range of measures of sentence centrality as signals for what is important in source documents.

In particular in news articles, such as those used for the DUC2002 single-document summarization and the DUC2004 multi-document summarization tasks,[1] it is quite common that the most important information is repeated most frequently. Indeed, Nenkova et al. (2006) showed that information which appears frequently in the input documents is likely to appear in a human-generated summary. Similar conclusions can be drawn for single-document news corpora, where, for example, important information is likely to be found at the beginning of the document (for impatient readers), and repeated and expanded later in the article.

Even though most research in text summarization to date focused on newswire documents, this special kind of text genre is not the only one to be considered for summarization. Recently, about 400 journalists organized by the International Consortium of Investigative Journalists (ICIJ) [2] spent more than a year to analyze 11.5 million documents in the Panama Papers repository,[3] which consists of emails, PDFs, and other text documents not belonging to the newswire genre. In such a heterogeneous collection of raw and unprocessed source documents we cannot assume that frequency information correlates with importance, and therefore cannot rely on (in)frequency as (un)importance signal.

Nevertheless, journalists are able to cope with such situations because they bring along their background knowledge about the world, which allows them to estimate what information is important and what is not. We therefore propose to incorporate world knowledge to handle more challenging summarization scenarios where centrality cannot be used as a signal for importance. Our assumption is that summarization systems which are aware of the importance of information without analyzing the structure of the source documents are able to summarize heterogeneous documents properly. The key question of the paper is whether a knowledge-based summarization system is still able to detect important information even when structural and centrality-based features cannot be used as signals for importance.

We first review well-known summarization sys-

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

[2] https://www.icij.org/

[3] https://panamapapers.icij.org/

Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning (CoNLL), pages 84–94,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics

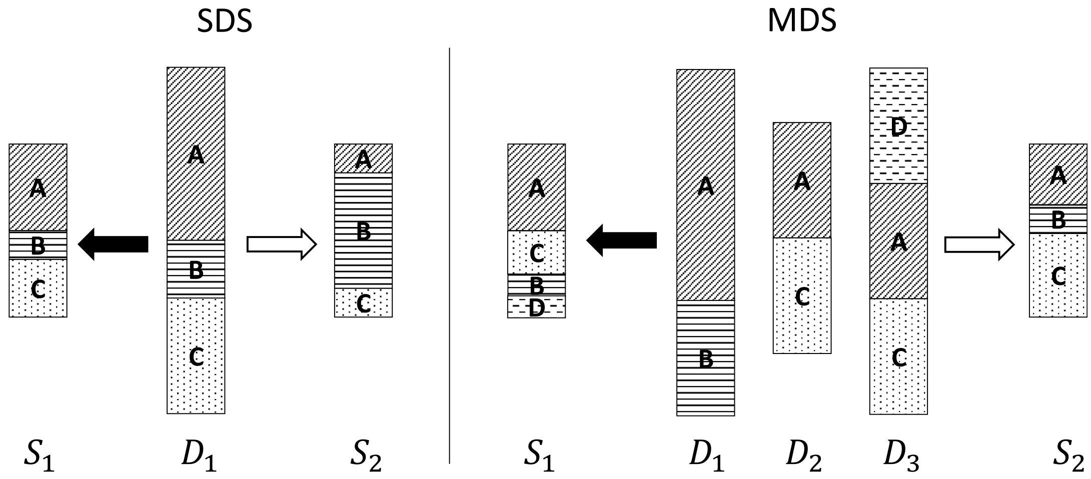


Figure 1: Differences between centrality-based summarization (black arrow) and importance-aware summarization (white arrow) in single- (left) and multi-document summarization (right).

tems for single- and multi-document summarization in Section 2. Particular emphasis is put on the methodologies used to identify important information and avoid redundancy since this is the main innovation of our knowledge-based system described in Section 3.

CPSum learns about importance by analyzing an independent background corpus of document-summary pairs and applies this knowledge in the summarization task. A major difference to previous systems is that we do not use similarity measures to compute centrality, neither for detecting importance nor for avoiding redundancy.

In order to verify our assumptions, we compare our approach on a commonly used evaluation corpus, both in its original version and in various version in which we remove redundancy and sentence order. We describe the corpus modification in Section 4. Expectedly, our experiments described in Section 5 and 6 show a substantial performance decrease for all tested reference systems, whereas the performance of *CPSum* remains essentially unchanged. The conclusions we draw from this study are summarized in Section 7.

2 Related Work

In this section, we review prior work in single- and multi-document summarization. The essential differences between these approaches and our approach are illustrated in Figure 1. On the left, a centrality-based SDS system summarizes a document D to a summary S_1. The result is a text with a similar topic distribution as in the source document. A system with a different notion of importance is able to produce a summary S_2 with a varied distribution of topics. On the right, we observe a similar situation, where a centrality-based MDS system summarizes a document collection $D_1, \ldots, D_3$ to a summary S_1 and a importance-aware summarization system produces the summary S_2 with a different topic distribution. Centrality-bases methods produce a smaller version of the source document(s) whereas an importance-aware summarization system is able to emphasis on important parts even if they are not frequent.

2.1 Single-Document Summarization

Early work in single-document summarization (SDS) by Luhn (1958) and Edmundson (1969) tries to identify salient information in documents. Luhn (1958) identifies words which are frequent in the source documents and infrequent in a background corpus. Edmundson (1969) extends this approach by using cue words and structural features such as title words and sentence position.

Several more recent methods are inspired by algorithms such as HITS (Kleinberg, 1999) and PageRank (Brin and Page, 1998) and model the source documents as graphs. TextRank (Mihalcea and Tarau, 2004) models sentences as nodes in a graph, where the strength of the connections between the nodes is determined by the similarity of the sentences measured by means of syntactic word overlap. Since Mihalcea and Tarau (2004) assume the absence of redundancy in single-document summarization there is no need for a re-ranking after selecting sentences. They also rely on sentence order by always including the first sentences in the summary.

Parveen and Strube (2015) use a bipartite graph to represent a document. This so-called topic graph has two sets of nodes, one containing sentences and one containing topics. To rank the sentences, they apply HITS . Since they deal with very long texts which may contain repetitive information they also apply a redundancy avoidance strategy by maximizing topic coverage in the summary and therefore minimizing redundancy.

All these approaches have in common that they focus on selecting the most central information from a document. However, in a noisy document with a significant amount of unimportant text, extracting the most central text may not be a good strategy. Summaries produced by these approaches will rather contain noise than important information, since the noise might be quite central (c.f. Figure 1). Most of the approaches assume that there is no redundancy in the source document and do not apply a redundancy avoidance strategy.

2.2 Multi-Document Summarization

In comparison to SDS the task in extractive multi-document summarization (MDS) is to summarize not one but a set of documents. The additional challenge in comparison to extractive SDS is that the document set may contain the same information redundantly in different documents. Therefore, in addition to detect important information, a second challenge is to avoid redundancy in the generated summary.

McKeown and Radev (1995) introduce the task of summarizing multiple news documents. Their system, called SUMMONS (SUMMarizing Online NewS articles), extends already existing template-driven message understanding systems.

Carbonell and Goldstein (1998) introduce Maximal Marginal Relevance (MMR) to reward centrality and penalize redundancy jointly with a linear combination of both attributes. In a query-based setup, sentences are greedily selected according to their similarity to the query and similarity according to already selected sentences. The similarity measure is based on the Cosine similarity between sentences.

Radev et al. (2000) use a clustering method to find a centroid. Clusters are built based on a topic detection system. For redundancy avoidance, they apply a redundancy penalty similar to the negative factor proposed by Carbonell and Goldstein (1998) and re-rank the sentences iteratively until re-ranking does not change the resulting summary.

LexRank (Erkan and Radev, 2004) is a graph-based MDS method inspired by social networks which uses intra-sentences cosine similarity to compute an adjacency matrix to represent the sentences as a graph similar to the graph-based methods in SDS. The most central sentence is considered to be the most important sentence. LexRank itself does not apply redundancy avoidance but only ranks sentences according to importance. As redundancy avoidance strategy, cross-sentence informational subsumption (CSIS) (Radev, 2000) is applied as a re-ranking strategy.

The best performing system at the DUC 2004 shared task in MDS, CLASSY (Conroy et al., 2004), uses TF-IDF scores to calculate the importance of sentences. ICSI Summ (Gillick et al., 2009), a well-performing system at TAC 2009, applies a global linear optimization to search for a set of sentences that covers relevant concepts in the source documents as well as possible. As concepts they use word bi-grams weighted by their frequency, thereby deriving importance from frequency. Since they search for a set of sentences which maximizes the sum of unique concept values, their system is able to avoid redundancy implicitly.

Lin and Bilmes (2011) treat MDS as a submodular maximization problem. By rewarding diversity rather than penalizing redundancy they created a monotone nondecreasing submodular utility function (in comparison to Carbonell and Goldstein (1998)) which has a constant factor guarantee of optimality. In contract to ICSI Summ, Yogatama et al. (2015) seek to not maximize bi-gram coverage but to maximize the semantic volume. They use embeddings to represent sentences and choose the subset of sentences that maximizes the size of the convex hull in the generated embedding space as summary.

We summarize that systems for MDS use (similar to systems for SDS) various centrality measures to detect important information. Furthermore, they apply redundancy avoidance strategies based on sentence similarity. *CPSum*, on the other hand, does not apply any similarity measure but learns from contextual preferences if something is important in the context of other information/sentences.

3 The *CPSum* Algorithm

In this section, we first define the summarization task formally. We then present the novel preference-based summarization system *CPSum* in detail. In particular, we explain our training procedure and (contextual) sentence ranking methodology.

3.1 Problem Definition

The task of a generic extractive summarization system is to create sequences of sentences (the summaries) from given sequences of sentences (the source documents) for different topics. The objective is that the selected sentences form a good summary of the source documents.

To formalize the task, we define a *sentence* s of length n as a sequence of n words $(s_1, \dots, s_n)$. For convenience we use the term *word* for all elements of a sentence identified by a sentence segmentation method. Therefore, numbers, punctuation marks, and similar elements are all considered to be words. A *document* D of length m is a sequence of m sentences $(\mathrm{s}_1, \dots, \mathrm{s}_m)$, and consequently also a sequence of words. $|X|$ denotes the length of the sequence X.

A *topic* is a pair $(\mathcal{D} = \{D_1, \dots, D_o\}, \mathcal{R} = \{R_1, \dots, R_p\})$ of input documents $\mathcal{D}$ and reference summaries $\mathcal{R}$. $|\mathcal{D}| = 1$ in single-document summarization and $|\mathcal{D}| > 1$ in multi-document summarization where $|\mathcal{D}|$ denotes the size of $\mathcal{D}$. Since we do not distinguish between different source documents we introduce $\dot{D} = D_1 \circ \cdots \circ D_o$ as the concatenation of all sentences of all source documents in $\mathcal{D}$.

The task of extractive document summarization is to find a sequence of sentences $\hat{S} \in \mathfrak{P}_{l_{max}}(\dot{D})$ that maximizes a utility function u where $\mathfrak{P}_{l_{max}}(\dot{D})$ denotes the set of all sequences of elements in $\dot{D}$ with $\sum_{\mathrm{s} \in \dot{D}} |\mathrm{s}| \leq l_{max}$. Formally, the task is to search for S_{max} with

$$S_{max} = \underset{S \in \mathfrak{P}_{l_{max}}(\dot{D})}{\arg\max}\ u(S). \qquad (1)$$

A proper utility function u is supposed to measure the quality of the summary. Approaches are usually evaluated by a comparison with given reference summaries. We refer to Section 5 where we introduce ROUGE as the utility function which is used to grade the produced summaries. The difficulty when developing summarization systems is to find an approximation of u without having access to the reference summaries.

3.2 Object Importance

The key idea of our approach is to learn the importance of objects from external sources. This assessment of importance should then be used in order to select the most relevant sentences independently of features derived directly from the source documents, such as structural information or redundancy and centrality. Hence, we believe that our system is more suitable for handling heterogeneous summarization scenarios where such features may not helpful for detecting important information.

As a proof-of-concept, we study a simple approach which learns the importance of objects from a large background corpus of document-summary pairs. Note that this corpus does not have to consist of document-summary pairs. The system could also learn from very diverse sources such as stock market prices to judge the importance of a company, the length of Wikipedia articles for learning about the importance of people, or the number of inhabitants as a signal of importance for cities. In a way this corresponds to the way humans use fast and frugal heuristics for problem solving (Gigerenzer and Todd, 1999).

We model object importance in the form of pairwise preferences (Fürnkranz and Hüllermeier, 2011). A preference $a \succ b$ models the situation that object a is preferred to object b. In this paper, we take a simple approach and model object importance in the form of pairwise preferences between bi-grams of stemmed words that occur in the documents. Preferences may be probabilistic, i.e., the probability that $a \succ b$ rather than $b \succ a$ is $\Pr(a \succ b) \in [0, 1]$, and it holds that $\Pr(a \succ b) + \Pr(b \succ a) = 1$. Due to the large number of observed preferences, each preference only provides a weak signal about the importance of an object, and object importance will be determined by aggregating probabilistic preferences (cf. Section 3.4).

Furthermore, we model the situation that an object a is preferred to object b in a *context* C with *contextual preferences* $a \succ b \mid C$. The intuition is that the preference relation between two objects may depend on a context. In summarization, this context models the information need of a reader, which depends e.g. on personal interests and already observed information. Since object preferences are context-aware they can be used to adapt to difference users and summarization situations. We use the context to model already observed information of a user. Since we select summary sentences iteratively, we model with the context knowledge which is already contained in a partial summary. Since we measure importance in a context and model the context with the partial summary we do not need an additional redundancy avoidance mechanism like most other approaches for multi-document summarization.

The fact, that object importance is learned from an external corpus, also increases the adaptiveness of our system. Since all people may have a different notion of importance, the system can be trained easily on different sources which reflect these different notions. For example, a summary generated for an engineer may look differently than a summary created for a business administration employee. Systems which do not have an adaptive notion of importance are not able to create different summaries for different information needs.

3.3 Learning of Object Importance

To learn the importance of an object we use a background corpus denoted by $\mathbb{B} = \bigcup(D_i, R_i)$ which provides a set of document-summary pairs. For the i-th topic in the corpus we observe the document D_i as well as the summary R_i. We use the same notation for the occurrence of objects in sentences and documents as for words, hence $a \in \mathbf{s}$ or $a \in D_i$ denotes that object a can be observed in $\mathbf{s}$ or D_i, respectively.

For each object pair a, b, for which it holds that a occurs in the summary as well as in the source document, and b occurs in the source document but not in the summary, we observe a preference $a \succ b$, since a was selected to be included in the summary whereas b was not. To formalize this, we first define two sets P_i and N_i for topic i. P_i contains all elements which were selected from the source document to be included in the summary and N_i contains all elements which are not included in the summary. To define the sets P_i and N_i we introduce first the notation $\sigma(D_i)$ and $\sigma(R_i)$ to reduce the sequences D_i and R_i to sets which contain each element at most once. We then define $P_i = \sigma(D_i) \cap \sigma(R_i)$ and $N_i = \sigma(D_i) \setminus P_i$. With P_i and N_i we define the number of observations for $a > b$ in the background corpus $\mathbb{B}$ as

$$n_{\mathbb{B}}(a \succ b) = \sum_{(D_i,R_i)\in\mathbb{B}} \mathbb{1}_{P_i}(a) \cdot \mathbb{1}_{N_i}(b) \qquad (2)$$

where $\mathbb{1}_X(x)$ is 1 if $x \in X$ and 0 otherwise.

To define the number of observations $a \succ b$ for a context C we extend the definitions for $\sigma(P_i)$ and $\sigma(N_i)$. First, we define $A \setminus\!\!\setminus B$ for two sequences A and B similarly to the set difference, i.e., the result is a sequence of elements where we remove elements from the first sequence which appear in the second sequence. If an element x occurs n-times in A and m-times in B, $A \setminus\!\!\setminus B$ contains the element x exactly $max(0, n - m)$-times (e.g. $(a, a, b, c) \setminus\!\!\setminus (a, b, d) = (a, c)$). We then define the set $P_i \mid C = \sigma(D_i) \cap \sigma(R_i \setminus\!\!\setminus C)$ and the set $N_i \mid C = \sigma(D_i) \setminus \sigma(R_i \setminus\!\!\setminus C)$. $P_i \mid C$ contains, similarly to P_i, all elements which are contained in the source document as well as in the reference documents without the context elements. The intuition is that these elements are important in the context of C whereas the elements in $N_i \mid C$ are not important given C.

The number of contextual preferences for the elements a and b and the sequence of context elements C in the background corpus $\mathbb{B}$ is defined as

$$n_{\mathbb{B}}(a \succ b \mid C) = \sum_{(D_i,R_i)\in\mathbb{B}} \mathbb{1}_{P_i|C}(a) \cdot \mathbb{1}_{N_i|C}(b). \qquad (3)$$

The context C models the objects which are already in a partial summary. Since our approach selects sentences sequentially, we have to detect the importance of objects according to already selected objects.

We can estimate the prior probability of observing $a \succ b$ as

$$\Pr(a \succ b) = \frac{n(a \succ b)}{n(a \succ b) + n(b \succ a)} \qquad (4)$$

and analogously for $\Pr(a \succ b \mid C)$.

3.4 Sentence Ranking

In this section, we propose a ranking methodology for all available sentences $\dot{D}$ in a multi-document summarization topic from a sentence-level utility function u. To rank the sentences, we iteratively search for the sentence $\hat{\mathbf{s}}$ with

$$\hat{\mathbf{s}} = \arg\max_{\mathbf{s}\in\dot{D}} u(\mathbf{s} \mid C) \qquad (5)$$

where $u(\mathbf{s} \mid C)$ is a utility function that encodes the importance of sentence $\mathbf{s}$ in a context C. The intuition is that the value of a sentence depends on already selected sentences which are modeled by the context C. Hence, we greedily set

$$\hat{S} \leftarrow \hat{S} \circ \arg\max_{\mathbf{s} \in \dot{D}} u(\mathbf{s} \mid \hat{S}) \tag{6}$$

as long as $\hat{S} \in \mathfrak{P}_{l_{max}}(\dot{D})$ and starting with $\hat{S} = \emptyset$.

Since we do not have access to the reference summaries when generating summaries, we use the learned knowledge from the training phase to estimate the utility of each sentence in order to find the sentence with the highest utility value. To do so, we propose a utility function u in the next section, which assigns a utility score to each individual sentence.

It is important to note at this point that we neither use any form of similarity measure between sentences nor any structural features such as sentence positions to determine the importance of a sentence, which is a crucial difference to previous works.

3.5 Individual Sentence Scoring

We obtain individual sentence scores, which means that each sentence is assigned a score independently from the other available sentences. Removing or adding sentences to the source documents will therefore not change the value of the sentences. The intuition of the score is, that we want to find the sentence which has the highest average probability that the objects in the sentence occur in the reference summary. The desired sentence $\hat{\mathbf{s}}$ is therefore selected by the utility function

$$u(\mathbf{s} \mid C) = \frac{\sum_{x \in \mathbf{s}} v(x \mid C)}{n} \tag{7}$$

where v is an object-level utility function which measures the importance of an object a in a context C. We define v for element x and a corpus $\mathbb{B}$ as

$$v(a \mid C) = \frac{1}{|V|} \sum_{x \in V} \Pr(a \succ x \mid C) \tag{8}$$

where $V = \bigcup_{(D_i,R_i) \in \mathbb{B}} \{x : x \in D_i\}$ is the set of all considered objects in the background corpus $\mathbb{B}$ (i.e. the vocabulary) and $\Pr(a \succ b)$ if $C = \emptyset$ and $\Pr(a \succ b \mid C)$ are estimated as in (4).

Note that it may happen that an object a or a preference $a \succ b \mid C$ might not have been observed in the background corpus. These cases are ignored in the computation.

4 Evaluation Corpora

The fundamental hypothesis of centrality-based summarization systems is that frequency within the source documents implies importance of information. All information which is frequent in the source documents is considered to be important and therefore extracted for the summary. While this may be a suitable assumption for some document collections (such as newswire documents), we do not believe that it is suitable for the task of summarizing heterogeneous document collections.

Since most of the work in summarization has been done for newswire data, there is a lack of evaluation data where structural and centrality signals do not provide a strong indicator for importance. We therefore modify the DUC2004 multi-document summarization corpus by *shuffling* and *oversampling* to remove the commonly used indicators for importance. By doing so, we intend to demonstrate that centrality-based document summarization algorithms break down, whereas *PLSum* will maintain its performance.

Shuffling: In order to remove order-dependency, we randomly shuffle the sentences to hide the very strong sentence position signal, which is commonly used to detect importance in news documents.

Oversampling: With oversampling we aim for hiding the important information in the corpus by increasing the frequency of unimportant information. In particular, we iteratively search for a sentence $\hat{\mathbf{s}}$ with

$$\hat{\mathbf{s}} = \arg\min_{\mathbf{s} \in \dot{D}} \sum_{\mathbf{s}_x \in \dot{D}} sim(\mathbf{s}, \mathbf{s}_x), \tag{9}$$

where sim is a similarity measure for two sentences, and add $\hat{\mathbf{s}}$ to a random document in topic $\mathcal{D}$. Since we duplicate the sentences we make sure that we do not introduce new, important information to the corpus which is not reflected in the summary. For the similarity measure we use

$$sim(\mathbf{s}_1, \mathbf{s}_2) = \frac{cos(\mathbf{s}_1, \mathbf{s}_2) + jacc(\mathbf{s}_1, \mathbf{s}_2)}{2} \tag{10}$$

in our experiments, where cos is a cosine similarity implemented in the DKPro Similarity framework (Bär et al., 2013) with TF-IDF values based on English Wikipedia articles, and $jacc$ denotes to the well-known Jaccard measure. This simple

Dataset	∅ Similarity
DUC2004	0.0877
DUC2004	0.0880
DUC2004 200%	0.0692
DUC2004 500%	0.0620
DUC2004 1000%	0.0607

Table 1: Average similarities of the sentences contained in the test corpora.

combination lead to reasonably good results on the English subtask of the SemEval2014 Semantic Textual Similarity dataset.[4]

With this methodology, we create four new corpora with 100%, 200%, 500%, and 1000% of the size of DUC2004. The bigger the corpora is, the more unimportant information has been added to it. In the 100% corpus sentences are only shuffled without duplicating sentences. With increasing size we hide the originally frequent information better and make it therefore harder to detect important information. An analysis of the result of the oversampling is displayed in Table 1. The average similarity decreases which means that we hide dense regions by adding sentences to less dense regions.

5 Evaluation

Since the DUC data provides manually written reference summaries, we can compare these gold standard summaries to the newly generated summaries of the automatic summarization systems. We provide in the evaluation ROUGE-1 (R1) and ROUGE-2 (R2) based recall scores according to Owczarzak et al. (2012) who showed that R2 provides the best agreement with manual evaluations when using stemming and without removing stopwords. As Rankel et al. (2013) showed that there is no clear winner between R1 and R2, we provide R1 as well, which is well suited to identify the better summary in a pair of summaries. Furthermore, all automatically generated summaries are truncated at a length of 100 words by the ROUGE system (Hong et al., 2014). Summarized, we use `ROUGE-1.5.5` with parameters `-a -m -n 2 -x -c 95 -r 1000 -f A -p 0.5 -t 0 -l 100 -d`.

5.1 Reference Systems

We will compare our algorithm, *CPSum*, to two baselines and to two well-known summarization algorithms.

Baselines: The first baseline *Optimal* has access to the reference summaries and is therefore no fair competitor for the remaining systems. Nevertheless, it provides useful information about the maximal reachable score for each dataset. Since computing the true optimal score is computational expensive, we only provide a pseudo-optimal value computed by a greedy search. The second baseline system, *Random*, selects sentences from the source documents randomly. It does not have access to the reference systems and is therefore the first system which can be compared with the other systems. Since most important information in news are often contained in the first sentences, just selecting the first few sentences as a summary is a strong baseline. We use *Lead* to provide evaluation scores for a system, which selects the first sentences of each document.

Summarization Systems: We use *Centroid* (Radev et al., 2000) as a representative system for centroid-based systems. To generate the summaries for this approach we apply the widely used MEAD system (Radev et al., 2004), in which *Centroid* is implemented. For *Centroid* we used the default linear feature combination, length cutoff and re-ranker. As a competitive state-of-the-art representative for graph-based approaches (Hong and Nenkova, 2014) we apply *LexRank* (Erkan and Radev, 2004), which is also implemented in the MEAD system. For *LexRank* we used the LexRank feature, standard length cutoff and the default re-ranker.

5.2 CPSum

Since *CPSum* learns about importance of objects from a background corpus, we need first a concrete instantiation for the abstract objects and second a background corpus to learn from. As instances for the objects for which we learn contextual preferences of the form $a \succ b \mid C$ we use bi-grams of stemmed words, which means that *CPSum* will learn about the importance of bi-grams. The contexts C is therefore a sequence of bi-grams. As mentioned above, any other linguistic entity like named-entities, opinions, or events would also be possible choices. Furthermore, vector representations for sentences could be applied as well. We decided to use bi-grams of stemmed words since they do not need any sophisticated pre-processing. Furthermore, showing that our approach is able to handle difficult summarization scenarios without a

[4] `http://alt.qcri.org/semeval2014/`

	ROUGE-1 Recall					ROUGE-2 Recall				
System / Dataset	D04	D04-1	D04-2	D04-5	D04-10	D04	D04-1	D04-2	D04-5	D04-10
Optimal	0.4043	0.4043	0.4046	0.4043	0.4044	0.0940	0.0941	0.0943	0.0940	0.0942
Random	0.2955	0.3095	0.2863	0.2736	0.2633	0.0435	0.0463	0.0360	0.0313	0.0322
Lead	0.3424	0.3138	0.2786	0.2636	0.2548	0.0766	0.0524	0.0382	0.0313	0.0282
Centroid	**0.3542**	0.3158	0.3082	0.2690	0.2474	**0.0867**	0.0605	0.0576	0.0396	0.0331
LexRank	0.3231	0.3219	0.3186	0.3052	0.2990	0.0659	**0.0645**	**0.0631**	0.0542	0.0522
CPSum	0.3267	**0.3247**	**0.3264**	**0.3264**	**0.3264**	0.0603	0.0604	0.0617	**0.0617**	**0.0617**

Table 2: ROUGE-1 and ROUGE-2 scores on the original and the modified DUC 2004 corpora.

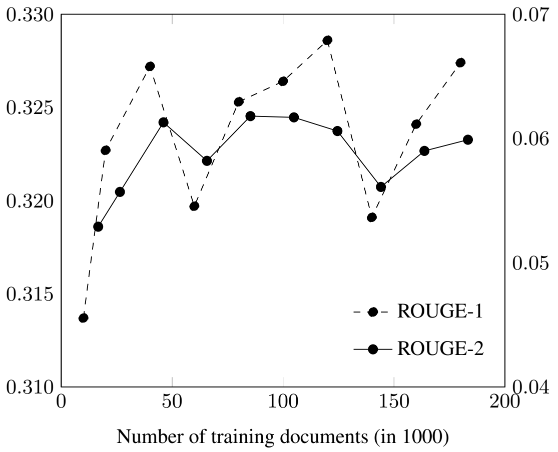


Figure 2: Learning curve of CPSum on DUC2004 for different background corpus sizes. ROUGE-1 scores (left scale) are displayed in blue and ROUGE-2 (right scale) in red.

sophisticated linguistic analysis of the data and relying solely on simple elements is an even stronger argument for the strength of *CPSum*.

For learning the importance of bi-grams we use a background corpus originally created by Hermann et al. (2015) for question answering tasks. Although this corpus does not provide well-written summaries for each article but only sentence-length bullet points summarizing the content of the article, we can use this information to derive the necessary training signals for learning object importance. The corpus contains about 100k CNN document-summary pairs crawled from CNN and about 197k pairs crawled from DailyMail. For training, we use a subset of 100k randomly selected documents in total.

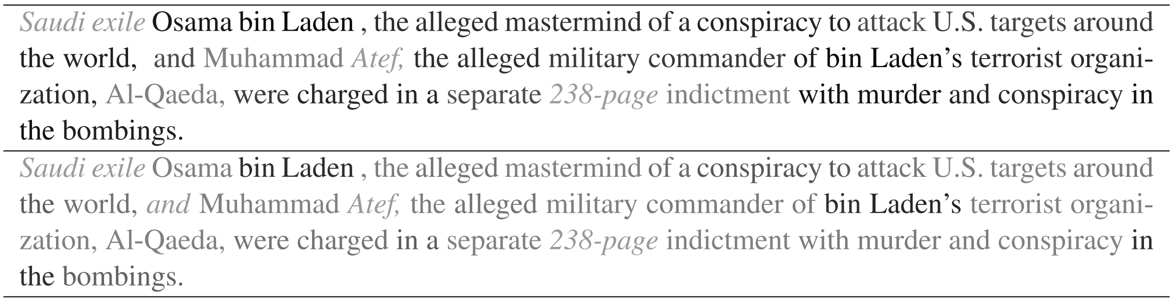
Since we will not observe most of the bi-grams from the training corpus in the test data, we apply a lazy learning strategy to only learn about elements which appear in the test data. Furthermore, we only learn preferences for contexts which we actually observe during summarization. This decreases the learning effort significantly.

6 Results

Table 2 shows the ROUGE evaluation scores of the tested systems on the test datasets. First, we see that the evaluation scores for both, ROUGE-1 and ROUGE-2 recall stays nearly constant for the oracle system *Optimal*. From this result we conclude that our modifications did neither remove from nor add important information to the corpus. After the modifications, it is still possible to generate summaries with a ROUGE-1 value of at least 0.40 and a ROUGE-2 value of at least 0.09. The performance of *Random* decreases when we add more irrelevant information to the corpus. This behavior is expected since the probability of picking an irrelevant sentences increases when more irrelevant sentences are in the corpus. The baseline *Lead*, which simply uses the first sentences of each article, but is considered to be a strong baseline in newswire documents, is able to summarize the original DUC 2004 data reasonably well. However, in the modified corpora with randomized sentence order, its performance is obviously not better than *Random*.

As expected, the two state-of-the-art reference systems work well on the original DUC 2004 corpus, where *Centroid* achieves the best results. This behavior is also expected, since it uses positional and centrality features, which provide very good signals for importance in the corpus. When these signals are more and more removed in D04-1 – D04-10, we observe a big performance decrease in both, ROUGE-1 and ROUGE-2. *LexRank* behaves similarly to *Centroid* but with a less fast decrease of performance.

CPSum performs only moderately at the original DUC 2004 dataset. This is again expected, since it does not use the strong importance signals sentence position and sentence centrality. The strength of *CPSum* can be observed in the modified corpora, where the performance stays comparable to the performance at the original corpus and does not decrease as it can be observed for all other



Saudi exile Osama bin Laden , the alleged mastermind of a conspiracy to attack U.S. targets around the world, and Muhammad *Atef,* the alleged military commander of bin Laden's terrorist organization, Al-Qaeda, were charged in a separate *238-page* indictment with murder and conspiracy in the bombings.

Saudi exile Osama bin Laden , the alleged mastermind of a conspiracy to attack U.S. targets around the world, *and* Muhammad *Atef,* the alleged military commander of bin Laden's terrorist organization, Al-Qaeda, were charged in a separate *238-page* indictment with murder and conspiracy in the bombings.

Figure 3: Example of the importance of the elements in a sentence before (top) and after (bottom) adding the sentence to the summary. The darker the font color the more important the element. Elements with less than 100 gathered preferences are displayed in italics. The importance scores are estimated by v as defined in Equation 8.

approaches. In terms of ROUGE-1 scores, *CPSum* has the best performance on the four modified corpora. In terms of ROUGE-2, its original performance is similar to the performance of *LexRank*, but lower that *Centroid*. The performance of *Centroid* drops significantly after shuffling the sentences. If we add more and more irrelevant sentences, the performance of *Centroid* drops again faster than the performance of *LexRank*. *CPSum* outperforms all systems when we increase the amount of noise in the corpora D04-5 and D04-10.

We show an example of the sentences scoring in Figure 3. We display the same sentence twice. In the top, we display the context-free scores of the elements of the sentence by using a darker font for more important information. In the bottom, we show the contextual scores of the same sentence after adding this particular sentence to the summary. We observe that the importance scores of elements such as *Osama bin Laden* are estimated properly. After adding the sentence to the summary we can see how *PLSum* discounts the scores for different elements differently.

7 Conclusions

In this paper we introduced *CPSum*, a text summarization system that learns the importance of entities from an independent background corpus of document-summary pairs. *CPSum* is able to cope with summarization scenarios where neither centrality nor structural features help to detect important information. We showed that the performance of conventional text summarization systems decreases in such a setting. Previous approaches can be confused easily by adding more and more irrelevant information whereas the performance of *CPSum* stays constant. We would argue that by relying on learned prior knowledge about what information is important for a summary, *CPSum* is able to detect important information similar to the way human experts address a summarization task.

CPSum is also different in the way it copes with redundancy. Instead of measuring the similarity to already selected sentences such as the majority of the previous systems, we estimate the score of the elements with contextual preferences. This enables *CPSum* not only to detect redundancy, but also to use synergy effects between sentences. Adding one sentence to the summary can therefore also increase the utility of other sentences. Furthermore, our system can be adapted easily to different user interest by learning from other source documents.

We intend to investigate other basis elements for the preferences as well as alternatives for modeling world knowledge in future work. Similarly, we are also working on a corpus with which we can further investigate summarization scenarios where centrality and structural features are no good signals for importance. The results of this first study make us confident that a knowledge-based approach towards importance information is necessary in order to enable summarization systems to handle difficult summarization scenarios where signals for importance cannot be inferred from the source documents.

Acknowledgments

This work has been supported by the German Research Foundation (DFG) as part of the Research Training Group Adaptive Preparation of Information from Heterogeneous Sources (AIPHES) under grant No. GRK 1994/1.

References

D. Bär, T. Zesch, and I. Gurevych. 2013. DKPro Similarity: An open source framework for text similarity. In *Proc. 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pp. 121–126, Sofia, Bulgaria. ACL.

S. Brin and L. Page. 1998. The anatomy of a large-scale hypertextual web search engine. *Computer networks*, 30(1-7):107–117.

J. Carbonell and J. Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proc. 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 335–336. ACM.

J.M. Conroy, J.D. Schlesinger, J. Goldstein, and D.P. O'Leary. 2004. Left-brain/right-brain multi-document summarization. In *Proceedings of the Document Understanding Conference (DUC 2004)*.

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

G. Erkan and D.R. Radev. 2004. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of Artificial Intelligence Research* 22:457–479.

J. Fürnkranz and E. Hüllermeier (eds.) 2011. *Preference learning*. Springer.

G. Gigerenzer and P.M. Todd. 1999. *Simple heuristics that make us smart*. Oxford University Press, USA.

D. Gillick, B. Favre, D. Hakkani-Tur, B. Bohnet, Y. Liu, and S. Xie. 2009. The ICSI/UTD summarization system at TAC 2009. In *Proc. 2nd Text Analysis Conference (TAC-09)*, Gaithersburg, MD (USA).

K.M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28 (NIPS-15)*, pp. 1684–1692.

K. Hong and A. Nenkova. 2014. Improving the estimation of word importance for news multi-document summarization. In *Proc. 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL-14)*, Gothenburg, Sweden, pp. 712–721.

K. Hong, J.M. Conroy, B. Favre, A. Kulesza, H. Lin, and A. Nenkova. 2014. A repository of state of the art and competitive baseline summaries for generic news summarization. In *Proc. 9th International Conference on Language Resources and Evaluation (LREC-14)*, Reykjavik, Iceland, pp. 1608–1616.

J.M. Kleinberg. 1999. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632.

H. Lin and J. Bilmes. 2011. A class of submodular functions for document summarization. In *Proc. 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Vol. 1*, pp. 510–520. ACL.

C.-Y. Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of the ACL-04 Text Summarization Workshop*, pages 74–81.

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

I. Mani. 2001. *Automatic summarization*, Vol. 3. John Benjamins Publishing.

K. McKeown and D.R. Radev. 1995. Generating summaries of multiple news articles. In *Proc. 18th annual international ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 74–82. ACM.

R. Mihalcea and P. Tarau. 2004. TextRank: Bringing order into texts. In *Proc. 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP-04)*, pp. 404–411. ACL.

A. Nenkova, L. Vanderwende, and K. McKeown. 2006. A compositional context sensitive multi-document summarizer: exploring the factors that influence summarization. In *Proc. 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 573–580. ACM.

K. Owczarzak, J.M. Conroy, H. Trang Dang, and A. Nenkova. 2012. An assessment of the accuracy of automatic evaluation in summarization. In *Proc. Workshop on Evaluation Metrics and System Comparison for Automatic Summarization*, pp. 1–9. ACL.

D. Parveen and M. Strube. 2015. Integrating importance, non-redundancy and coherence in graph-based extractive summarization. In *Proc. 24th International Conference on Artificial Intelligence (AAAI-15)*, pp. 1298–1304. AAAI Press.

D.R. Radev. 2000. A common theory of information fusion from multiple text sources—Step one: cross-document structure. In *Proceedings of the 1st SIGdial workshop on Discourse and dialogue (SIGDIAL-00), Vol. 10*, pp. 74–83. ACL.

D.R. 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, M. Topper, A. Winkel, and Z. Zhang. 2004. MEAD-a platform for multidocument multilingual text summarization. In *Proc. 4th International Conference on Language Resources and Evaluation (LREC-04)*. European Language Resources Association.

D.R. Radev, H. Jing, and M. Budzikowska. 2000. 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*, pp. 21–30. ACL.

P.A. Rankel, J.M. Conroy, E.V. Slud, and D.P. O'Leary. 2011. Ranking human and machine summarization systems. In *Proc. 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP-11)*, pp. 467–473. ACL.

P.A. Rankel, J.M. Conroy, H. Trang Dang, and A. Nenkova. 2013. A decade of automatic content evaluation of news summaries: Reassessing the state of the art. In *Proc. 51st Annual Meeting of the Association for Computational Linguistics (ACL-13)*, Sofia, Bulgaria, pp. 131–136.

J. Sjöbergh. 2007. Older versions of the rougeeval summarization evaluation system were easier to fool. *Information Processing & Management*, 43(6):1500–1505.

X. Wan. 2010. Towards a unified approach to simultaneous single-document and multi-document summarizations. In *Proc. 23rd International Conference on Computational Linguistics (ACL-10)*, pp. 1137–1145. ACL.

D. Yogatama, F. Liu, and N.A. Smith. 2015. Extractive summarization by maximizing semantic volume. In *Proc. Conference on Empirical Methods in Natural Language Processing (EMNLP-15)*, pp. 961–1966, Lisbon, Portugal. ACL.