SUMDocS: Surrounding-aware Unsupervised Multi-Document Summarization

Qi Zhu *† Fang Guo *† Jingjing Tian ‡ Yuning Mao † Jiawei Han †

Abstract

Multi-document summarization, which summarizes a set of documents with a small number of phrases or sentences, provides a concise and critical essence of the documents. Existing multi-document summarization methods ignore the fact that there often exist many relevant documents that provide surrounding background knowledge, which can help generate a salient and discriminative summary for a given set of documents. In this paper, we propose a novel method, SUMDocS (*Surrounding-aware Unsupervised Multi-Document Summarization*), which incorporates rich surrounding (*topically related*) documents to help improve the quality of extractive summarization *without human supervision*. Specifically, we propose a joint optimization algorithm to unify global novelty (i.e., *category-level frequent and discriminative*), local consistency (i.e., *locally frequent, co-occurring*), and local saliency (i.e., *salient from its surroundings*) such that the obtained summary captures the characteristics of the target documents. Extensive experiments on news and scientific domains demonstrate the superior performance of our method when the unlabeled surrounding corpus is utilized.

1 Introduction

With the ubiquity of massive text data in today's world, text summarization (*i.e.*, identifying summarative terms [19] and sentences [31] of a given set of documents) has become a cornerstone application for text understanding and document (e.g., online news) recommendation.

This paper studies extractive multi-document summarization, that is, extracting summarative text units (phrases or sentences) from multiple documents of the same topic. Recently, neural methods [21] have been extensively used in supervised text summarization and the fine-tuning [33] of pre-trained language model like BERT [7] further improves the summary quality with the help of large unlabeled corpora. However, these models are not well-suited for multi-document summarization due to its different nature and limited supervision. Traditional unsupervised multi-document summarization systems are mainly built upon co-occurrence of text units [31, 8] or objectives regarding summary coverage and saliency [16]. These methods utilize information solely from the collection of documents to be summarized, ignoring the fact that related documents beyond the collection could be useful for identifying *salient* information. This contrasts with the summaries written by humans who have the *background* knowledge on similar topics. For example, to summarize articles about **"Ethiopian Airlines Crash"** in March 2019, a traditional multi-document summarization method may generate the following result:

Summary A. The Ethiopian Airlines Boeing 737 MAX 8 bound for Nairobi, Kenya crashed ... Boeing is deeply saddened to learn of the passing of the passengers ... Boeing officials have pledged to correct the erroneous activation ...

The above summary, though reasonable, does not make the most salient point explicitly: the *unusual cause* of the crash. Different from many other crashes, where pilots' improper behaviors or severe weather conditions are to blame, this particular crash was mainly caused by the defective parts in Boeing 737 MAX. Equipped with commonsense knowledge, a human reader can quickly grasp two key points: first, this is about a plane crash disaster: frequent and distinctive keywords (*e.g.*, "pilot", or "black box") are important while other keywords (*e.g.*, "government", or "reporter") should be ruled out; second, comparing with other plane crashes, the distinctive aspects of this specific accident (*e.g.*, "Boeing 737 Max", "faulty sensor") are important and should be included in the summary. By utilizing background knowledge, a new summary that points out the *cause* of the crash ("faulty sensor") can be generated as follows.

Summary B. The Ethiopian Airlines Boeing 737

*These two authors contribute equally.

†Department of Computer Science, University of Illinois, Email: qiz3@illinois.edu, fangguo1@illinois.edu, yuningm2@illinois.edu, hanj@illinois.edu

‡Department of Computer Science, Peking University, Email: tianjj97@pku.edu.cn

MAX 8 bound for Nairobi, Kenya crashed ... The doomed Ethiopian Airlines jet suffered from faulty readings by a key ... Boeing officials have pledged to correct the erroneous activation ...

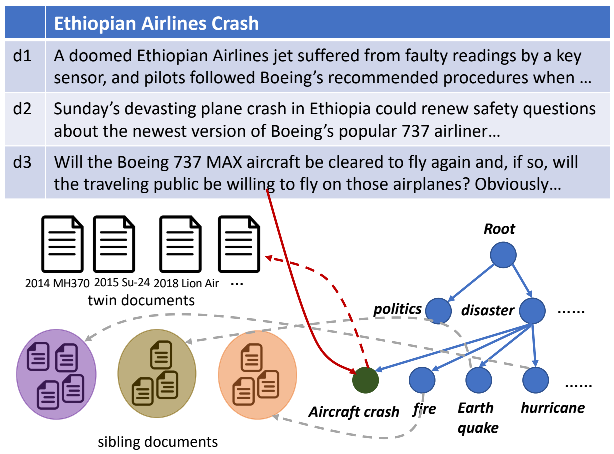


Figure 1: Examples of Surrounding Documents

Using surrounding documents, though appealing, poses challenges on how to identify and contrast against appropriate surrounding knowledge. There may be millions of unstructured text documents in a background corpus, which makes accurate identification of useful surrounding knowledge necessary. In our method, we define surrounding documents as a subset of the background corpus that is either semantically close (*twins*) or orthogonal (*siblings*) to the target documents. In the previous example, *twins* are the similar documents under category "Aircraft crash" in Figure 1. *Siblings* are representative documents under orthogonal categories like "fire" and "earthquake". We use category name-guided embedding [18] to allocate the documents in the background corpus along a given category and then identify the surrounding documents of the target document set in the embedding space. SUMDocS features a phrase selection module (section 3.2) to pick salient phrases that are both discriminative **w.r.t.** *twins* and representative **w.r.t.** *siblings.* The summary is selected via a submodular set function (section 3.2.1). On both news and scientific datasets, our method beat the other unsupervised methods easily and even on par with supervised method trained on the same domain.

To summarize, our main contributions are as follows:

1. We recognize the benefits of utilizing background corpus in the problem of multi-document summarization and formulate the surrounding-aware multi-document summarization problem.
2. We propose an unsupervised extractive summarization methodology SUMDocS that captures *salient* information in the target documents by utilizing background corpus.

2 Problem Definition & Preliminary

A target document set $\mathcal{T} = \{d_1, d_2, \ldots, d_n\}$ for multi-document summarization is defined as a collection of correlated articles on the same event or topic. Given a background corpus $\mathcal{D}$ (*i.e.*, corpus in the same domain), surrounding document set $\mathcal{S}$ is a subset of documents $\mathcal{S} \in \mathcal{D}$, which is semantically related to the target documents $\mathcal{T}$. Given a background corpus $\mathcal{D}$ and a target document set $\mathcal{T}$, we assume their category names $\mathcal{C} = \{c_1, c_2, \ldots, c_n\}$ are provided as guidance to identify the surrounding documents $\mathcal{S}$. The task of surrounding-aware multi-document summarization aims to comparatively summarize $\mathcal{T}$ against retrieved surrounding documents $\mathcal{S}$ into a list of extractive sentences $s_1, s_2, \ldots, s_m$ from text indicating the (1) **saliency** among documents $\mathcal{T}$; and (2) **novelty** beyond information in the surrounding documents $\mathcal{S}$.

3 Method

SUMDocS consists of two major components: background corpus categorization (Sec. 3.1) and comparative summarization (Sec. 3.2). Using category names only, we adopt the category-name guided text embedding [18] to obtain the document and category (label) embeddings. Articles of the unlabeled corpus are assigned into different categories such as *politics*, *business* in news domain. For each target document set, we retrieve the most similar documents in the same category and representative documents in other categories, namely, *twin* and *sibling* documents. For the comparative summarization, we proposed a graph-based manifold ranking algorithm to calculate the phrase salient scores regarding: (1) whether it's a frequent word in target documents but not *siblings* (2) whether it's a relatively fresh term comparing with twin.

3.1 Background corpus categorization

3.1.1 Modeling category-name guided text embedding

We build a category-name guided text embedding model to help identify the twin and *sibling* documents in the latent embedding space. It embeds documents d, category name c and word w into the same space as u_d, u_c and u_t, respectively. Similar with [18, 20], we conduct the embedding learning via capturing the co-occurrence between category-document (C-D), document-word (D-W) and word-word (W-W).

Although the category of each document is unknown, we re-write $p(d|c_d)$ by marginalizing the probability of observing each word in the document under category c_d,

$$p(d|c_d) \propto p(c_d|d)p(d) \propto p(c_d|d) \propto \prod_{w \in d} p(c_d|w)$$

where $p(c_d|w)$ is computed between category and word embeddings as $p(c|w) \propto \exp(u_c^\intercal u_w)$

$$\mathcal{L}_{\text{C-D}} = -\sum_{d \in \mathcal{D}} \log p(d|c_d) = -\sum_{c \in \mathcal{C}} \sum_{w \in c} p(c|w) + const.$$

The document and word co-occurrence probability $p(w|d)$ is computed as embedding similarity between document and corresponding word.

$$\mathcal{L}_{\text{D-W}} = -\sum_{d \in \mathcal{D}} \sum_{t_i \in d} \log p(w_i|d), \; p(w_i|d) \propto \exp({u_{w_i}}^\intercal u_d)$$

The co-occurrence between words are modeled as same as skip-gram objective[20],

$$\mathcal{L}_{\text{W-W}} = -\sum_{d \in \mathcal{D}} \sum_{t_i \in d} \sum_{\substack{w_{i+j} \in d \\ -h \leq j \leq h, j \neq 0}} \log p(w_{i+j}|w_i).$$

where $p(w_{i+j}|w_i) \propto exp({u_{w_i}}^\intercal u_{w_{i+j}})$ and h is the size of the context window. Combining the aforementioned three objectives, the overall embedding training loss $\mathcal{L} = \mathcal{L}_{\text{C-D}} + \mathcal{L}_{\text{D-W}} + \mathcal{L}_{\text{W-W}}$. More details can be found in the original paper [18].

3.1.2 Twin and sibling documents identification

To categorize target documents, we aim to obtain the category distribution on target documents $\mathcal{T}$, *i.e.* $p(c_d|d)$. Similar with the embedding training, we transform the $p(d|c_d)$ into marginalized word-category distribution $p(c|w)$.

$$p(c_d|d) \propto \prod_{w \in d} p(c_d|w), \tag{3.1}$$

Note that target document set $\mathcal{T}$ contains multiple documents, we infer its category as the major label of each document $d \in \mathcal{T}$. As stated in Section 1, we utilize surrounding documents in two ways, *i.e.*, sibling documents A and twin documents B. Once target documents $\mathcal{T}$ are categorized under category c_t, we will retrieve *sibling* documents from sibling categories of c_t as sibling documents $A_1, A_2, ...A_n$. For instance, the sibling documents of "Ethiopian Airlines Crash" are from categories like "fire", "earthquake", *etc.*

Although the document embedding for target documents are missing from the embedding procedure in section 3.1, we are able to retrieve the *twin* documents from the embedding space. We calculate the pseudo document embedding $V_d, d \in \mathcal{T}$ of target documents as weighted average of its word embeddings $u_w, w \in d$. Meanwhile, we use the same method to compute the pseudo document embedding of documents in the background corpus $\mathcal{D}$. The twin documents B are $|\mathcal{T}|$-most similar documents among documents categorized under c_t in the embedding space.

3.2 Surrounding-aware summarization

Now we describe how to do comparative summarization with *twin* and *sibling*. We adopt three hypotheses to incorporate sibling A and twin B documents.

1. **global novelty**: category-level frequent and discriminative phrases are likely to be *salient* phrases, *e.g.* crash and Boeing in Figure. 2.
2. **local consistency:** frequently co-occurred phrases should have similar *salient* score.
3. **local saliency:** phrases that are *salient* in target documents but less *salient* in twin documents should be prompted. For example, faulty reading, MCAS and Ethiopian Airlines in "Ethiopian Airlines Crash" are less *salient* in other air crashes.

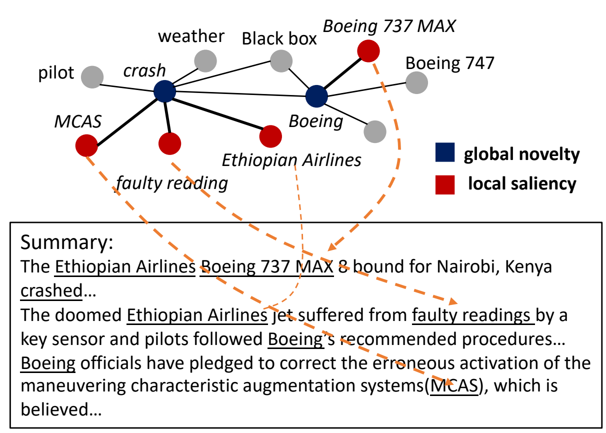


Figure 2: Text co-occurrence graph of the target documents. Global novelty are binary labels obtained from sibling documents. The scores are calculated as the difference between target and twin documents.

Our algorithm enforces global novelty and local consistency on two text co-occurrence graphs (Figure. 2) from target documents $\mathcal{T}$ and twin documents B. Then the local saliency of the phrase is calculated between two graphs as the criterion to select summarization terms.

Formally, suppose there are n different phrases in target documents $\mathcal{T}$, where $|\mathcal{T}| = m$. We use $W \in \mathbb{R}^{n \times n}$

to represent the edge weights between phrase. F_i and F'_i are phrase p_i's score in target document and twin documents. We denote $sim(i,j,d)$ as times of co-occurrence of phrase p_i and p_j in document d. We have $W_{i,j} = \sum_k^m \min(\delta, sim(i,j,d_k))$, where δ prevents one single document dominating the adjancency matrix.

Assuming c_d is the category of the target documents, category-level frequent phrases $\mathcal{P}^+$ are selected based on their representativeness $r(p, c_d)$ between target documents $\mathcal{T}$ and sibling documents $A_1, A_2, ..., A_n$. We denote $G \in \{0,1\}^n$ as the indicator vector of novelty, *i.e.*, $g_i = 1$ if $p_i \in \mathcal{P}^+$, $\mathcal{P}^+ = \text{argmax}_{|P|=k} \sum_P r(p, c_d)$. We use $k = 10$, namely, ten phrases as global novelty in our experiments. Several different representativeness scores can be used here, *e.g.* tf-idf. In our experiment, we use the phrase scores calculated in [30].

With $\mathcal{L}_{d\in\mathcal{T},\mu}$ and $\mathcal{L}_{d'\in B,\mu}$ denoting graph manifold ranking objective in target documents $\mathcal{T}$ and twin documents B, we have,

$$\mathcal{L}_{d,\mu}(F) = \sum_{i,j}^{n} W_{i,j} \| \frac{F_i}{\sqrt{D_i}} - \frac{F_j}{\sqrt{D_j}} \|^2 + \mu \sum_{i}^{n} \|F_i - g_i\|^2, \tag{3.2}$$

where D_i is the i-th row-wise sum of W, μ is a non-negative parameter controlling the global novelty weight. g_i is the binary global novelty label calculated above. The first term imposes the local consistency between neighboring phrases across documents,

Then we denote $Y \in \{0,1\}^n$ as the indicator vector of output words and we define the measure of the local saliency $\Phi(p_i, \mathcal{T}, B)$ of phrase p_i as score difference between two graphs, $\Phi_i = F_i - F'_i$. Phrases with $Y_i = 1$ are our selected summarization terms. Finally, we combine the local saliency Φ into the following joint optimization between target documents and twin documents.

$$\begin{aligned} L = \tfrac{1}{2}\mathcal{L}_{d,\mu}(F) + \tfrac{1}{2}\mathcal{L}_{d',\mu}(F') - \lambda \cdot \sum_{i=1}^{n} Y_i \cdot (F_i - F'_i). \\ s.t. \quad Y_i \cdot (F_i - F'_i) \geq Y_i \frac{\sum_{i=1}^{n}(F_i - F'_i)}{n}, \sum_{i=1}^{n} Y_i \leq K \end{aligned}$$

where we further control size $|Y|_1$ by two constraints: (1) above average salience score among all phrases (2) number of salience phrases are bounded by constant K.

However, directly optimization of this mix-integer programming problem is NP-hard. We approximate it by optimizing $\{F, F'\}$ and $\{Y\}$ iteratively. During each step, the above optimization has closed-form solution as follows:

$$\begin{aligned} F^* &= (1-\alpha)(I - \alpha S)^{-1} \cdot (G + Y), \\ F'^* &= (1-\alpha)(I - \alpha S')^{-1} \cdot (G - Y), \\ S &= D^{-\frac{1}{2}} W D^{-\frac{1}{2}}, S' = D'^{-\frac{1}{2}} W' D'^{-\frac{1}{2}}. \end{aligned} \tag{3.3}$$

3.2.1 Submodular selection module At last we follow the recent study to generate extractive summary from phrases considering both coverage and diversity [3, 31]. We formulate sentence selection as a submodular function over coverage $\mathcal{C}$ and diversity $\mathcal{D}$.

$$\mathcal{F}(S) = \mathcal{C}(S) + \lambda \mathcal{D}(S) \tag{3.4}$$

where $\mathcal{C}(S) = \sum_{i\in S} n_i \Phi_i$ measures the quality of current summary as sum of phrase *salient* score, and $\mathcal{D}(S) = N_{i\in S}/N_{\#keywords}$ measures the summary diversity. To push forward a fair comparison, for different phrase-based summarization systems, we use same $N_{\#keywords}$. The above objective can be solved greedily with $(1 - 1/e)$-approximation [24].

4 Experiments

In this section, we compare with other multi-document summarization systems to examine our three major claims:

- SUMDocS consistently outperforms other unsupervised multi-document summarization methods on both lexical and semantic measures.
- Background documents are beneficial for the task of unsupervised multi-document summarization.
- The proposed algorithm can produce sensible summaries in different domains with the help of background documents.

4.1 Experimental Settings

4.1.1 Datasets We use two large-scale multi-document summarization datasets from two different domains(*i.e.* News and Scientific) to evaluate the effectiveness of proposed algorithm. (1) Multi-News [9] collects news articles and human-written summaries pairs from the site newser.com. It includes news from over 1500 different sources and total 44,972/5,622/5,622 document sets as train, validation and test, respectively. Most of the target document set has only two documents. As shown in Table 1, we construct a smaller but more challenging subset of the Multi-news that filters target documents with less than five articles. (2) Scientific-NLP are scientific papers are collected from top natural language processing

Dataset	#corpus (background)	#test (docs)	document length	summary length
Multi-News*	44972	400	5071	358
Scientific-NLP	1892	120	4459	152

Table 1: Datasets statistics in average number of words. Multi-News* is a subset of original Multi-News [9].

conferences between 2016 and 2019. The original pdf files are parsed into json files using Science-Parse[1]. For each json file, we remove less-relevant parts like "acknowledgement","bibliography" and noisy texts like "et al." We evaluate methods with remaining sections from the paper as input and utilize the abstract of the paper as the ground-truth summary. The detailed statistics of these two datasets can be found in Table. 1. In both datasets, length of target documents are ∼20X (5,071 v.s 264) larger than traditional single document summarization dataset like NYT[2], which can not be scaled by most generative/abstractive seq2seq models.

4.1.2 Baselines Since our proposed method is unsupervised, we mainly compare the performance among several major unsupervised multi-document summarization systems. Besides, we also provide some recent sequence to sequence summarization models in the benchmark dataset [9, 5] as a comprehensive comparison between supervised and unsupervised methods.

We first introduce the unsupervised multi-document summarization baselines used in the experiments. TextRank [19] represents text units as nodes in a graph and rank phrases based on the centrality. LexRank [8] is a graph-based method that measures lexical importance among different sentences. Graph Degeneracy Summarization (GraphDegen) [31] is a recent graph-based method that identifies summary terms as highly influential spreaders in the dense subgraph structure. DensityPeak [34] is a clustering-based methods that models representativeness and diversity simultaneously.

We also consider one of the most recent abstractive summarization method, Hi-MAP [9]. It incorporates hierarchical MMR-attention in the pointer-generator network and achieves the state-of-the-art performance on Multi-News dataset.

Our ablations. SUMDocS is the proposed method utilizing the category-guided embedding on the corpus to locate twin documents and a joint graph-based optimization to rank the input sentences. To test the effectiveness of the background corpus in our algorithm, we build two ablations: SUMDocS-**NoBKG** and SUMDocS-**NoTwin**, in order to study the importance of sibling documents and twin documents respectively.

4.1.3 Experiment Details We pre-process both testing and background corpus using AutoPhrase [28] to recognize quality phrases in the text. In our background corpus categorization (Sec. 3.1), we use negative sample ratio k as 5 and the number of seed words per topic is selected using 5 keywords with the highest tf-idf scores. In News corpus, we use five different category names: *science*, *politics*, *disaster*, *business* and *sports*. In the Scientic-NLP dataset, we use eight different topics under natural language processing as categories: *text embedding*, *text classification*, *language model*, *machine translation*, *question answering*, *entity recognition*, *sentence matching* and *relation extraction*. The embedding model is optimized using Adam, learning rate is initialized as 0.001. We set the number of negative samples and window size both at 5 in the embedding learning. For SUMDocS and our ablations, we choose top-100 most representative documents from categories other than predicted class for target documents. Number of seed words as global novelty Y is set as 10. For all the methods using submodular sentence selection (Equation 3.4) including SUMDocS, we use the same control parameter $\lambda = 2$ for for diversity measure. The code and data are released in Github repository[3].

4.1.4 Evaluation Metrics ROUGE [15] score is commonly used in document summarization tasks. It measures the lexical overlap (*e.g.* unigram, bi-gram) between the system and reference summaries. However, people have been arguing that ROUGE is not capable of capturing synonyms, namely, semantic similarity. Earth mover distance [4] are proposed recently to capture the semantic similarity with word (WMD) and sentence embedding (SMD). Thus, we use both ROUGE and embedding mover distance measure in our experiments.

4.2 Experiments and Performance Study On Multi-News and Scientific-NLP test set, we compare SUMDocS with other baselines under ROUGE and embedding mover distance. The results are shown in Table 2.

"*Does* SUMDocS *outperforms other methods across different domains?*"

Compared with other unsupervised extractive methods, SUMDocS yield the best performance across two datasets on R-1, R-2 and all of the semantic measures by a wide margin. Sometimes, other extrac-

[1] https://github.com/allenai/science-parse
[2] https://catalog.ldc.upenn.edu/LDC2008T19
[3] https://github.com/GentleZhu/text_summarization

tive baselines like GraphDegen (Scientific-NLP) and SUMDocS-**NoBkg** (on Multi-News) achieves slightly better R-L scores. It has also been discovered by the previos work [32] that ROUGE-1,2 tend to measure the informativeness of the summary but longest common subsequence (ROUGE-L) captures fluency more. When compared with pre-trained neural abstractive and extractive baselines, SUMDocS still outperform on both measures only with exception against Hi-MAP on Multi-News. It is because Hi-Map, as a pre-trained model, is optimized on Multi-News training corpus with massive training data. When being applied to different domain, Hi-Map suffers a lot on performance. SUMDocS, however, as an unsupervised method, enjoy both effectiveness and efficiency on different domains.

"*Does background corpus help?*"

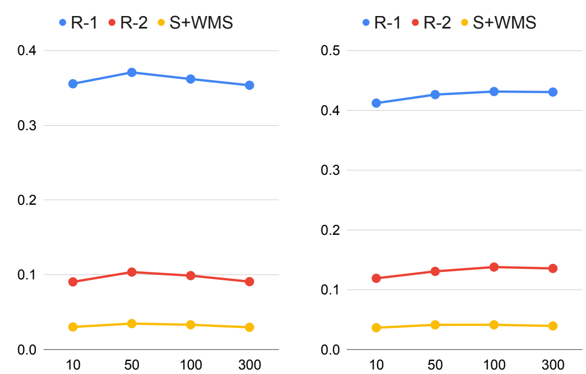
In this paper, we introduce the problem of surrounding-aware multi-document summarization and we want to validate that background documents are beneficial to the task. We have two different ablations that either ignores the *twin* documents or both *twin* and *sibling* documents. In Table 2, the result clearly shows SUMDocS beat these ablations on all of the measures. Specifically, SUMDocS-**NoTwin** is slightly better than the no background version. It reveals the necessity of global novelty (*i.e.* siblings) and local saliency (*i.e.* twins) used in our Equation 3.2. As expected, performance of SUMDocS-**NoBkg** is comparable with other baselines and indicates improvements of SUMDocS are mainly from the introduce of background corpus. We also observes a larger gap on the Scientific-NLP corpus, which is probably due to the underlying topic distribution is more distinctive than general news.

"*Does* SUMDocS *produces sensible summaries with background corpus?*"

In order to answer the above question, we present top-scored phrases that appear only in SUMDocS or SUMDocS-**NoBkg** in Table 3 and 4. In the same table, we also present the ground truth summary of the target documents. For the scientific-NLP dataset, we choose "BERT" as an example. As shown in the Table 4, SUMDocS is able to give higher ranking to those phrases that relate to the characteristics of BERT, such as "left-to-right", "mlm (masked language model)", "bidirectional", *etc.* Meanwhile, our ablation without background information mainly captures words generally seen in NLP papers like "model" and "fine-tuning". In the News domain, we study the news of former associate justice's death at Table 3. Main content in the ground truth summary are the cause of the death and the statement. SUMDocS captures these information in both intermediate keywords and extracted summary. Apparently, our method generates high quality summary with the acquired background knowledge. The improvement happens as early as the salient phrases are selected.

4.3 Parameter Study

4.3.1 Varying number of salient keywords We are interested that whether the number of key phrases returned by the graph optimization algorithm in SUMDocS affect the performance by a large amount. Hence, we study the performance of SUMDocS by varying the number of key words used in submodular sentence selection. In Figure 3, we demonstrate the results on two datasets and two different measures. In general, SUMDocS perform better with 50 or 100 output keywords. Fewer or more number of keywords will make the submodular selection module either contains limited information or flat out the important information.



(a) Multi-News **(b)** Scientific-NLP

Figure 3: Performance of SUMDocS varying number of keywords used in submodular selection.

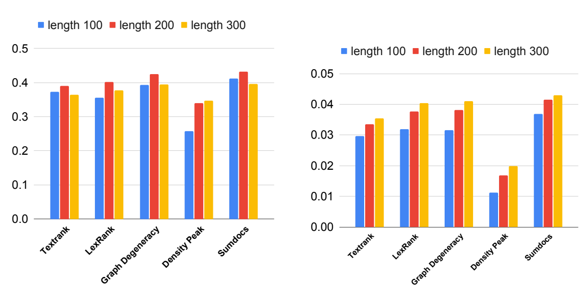
4.3.2 Varying length of the output summary Then we study the performance variance on Scientific-NLP between different methods when the output summary length varies. As shown in Figure 4, SUMDocS consistently perform better at various output lengths. Moreover, even the length-100 summary generated by ours beat the quite a few longer summary generated by other baselines. In Figure 4a, almost every algorithm performs best at length-200 because the average length of ground truth is about 200 words and F1 ROUGE score is used in our experiments. The sentence and word mover distance measure in Figure 4b is not penalized by precision and longer summary would always be better on score.

Table 2: Performance on Multi-News and Scientific-NLP dataset. Hi-MAP is trained on Multi-News, thus good performance on its test data is as expected. SUMDocS performs almost best or second best all the time.

Methods	Multi-News						Scientific-NLP					
	RG-1	RG-2	RG-L	WMD	SMD	S+WMD	RG-1	RG-2	RG-L	WMD	SMD	S+WMD
TextRank	36.34	8.84	30.86	0.81	13.77	3.27	39.06	10.27	21.82	0.85	13.94	3.36
LexRank	35.55	8.68	30.47	0.86	13.66	3.35	40.20	11.02	20.87	1.02	**14.61**	3.76
GraphDegen	35.16	9.12	33.04	0.89	10.32	2.99	42.54	13.00	**25.51**	1.25	11.80	3.81
DensityPeak	31.02	6.84	29.39	0.67	5.27	1.88	34.00	6.94	21.34	0.52	5.30	1.68
Hi-MAP*	**38.05**	**11.20**	**34.03**	0.93	10.67	3.12	32.29	7.42	24.44	0.43	8.87	1.92
SUMDocS-**NoBkg**	35.79	9.20	32.88	0.88	10.30	3.32	41.76	12.42	24.75	1.19	11.73	3.68
SUMDocS-**NoTwin**	36.57	9.64	31.12	0.98	**13.09**	**3.52**	41.83	12.53	22.60	1.16	14.34	3.99
SUMDocS	37.07	10.37	32.34	**1.02**	12.35	3.48	**43.14**	**13.80**	25.03	**1.35**	13.33	**4.15**

Table 3: Qualitative analysis on Multi-News. We compare SUMDocS and our ablations without background corpus. We present the different top-scored phrases selected by each method and their appearance in the ground truth summary.

	SUMDocS	SUMDocS-**NoBkg**	ground truth
keywords	79, abbott, god, february, patriot, statement, 13, appeared, natural, 2016	death, obama	N.A.
summary	breaking : u.s. supreme court justice antonin scalia found dead at west texas ranch at 79 cbs news (@cbsnews) february 13, 2016 cbs news reported scalia appeared to die of natural causes, according to a u.s. marshals service spokesperson. bush said scalia will be missed. scalia was nominated to the u.s. supreme court in 1986 by president ronald reagan. abbott said scalia set an example for citizens. scalia's legacy is enormous. greg abbott released a statement saturday afternoon, calling scalia a man of god, a patriot and...	bush said scalia will be missed. scalia's legacy is enormous. scalia was nominated to the u.s. supreme court in 1986 by president ronald reagan. scalia was just as ready for combat outside the court. similarly, scalia redefined and popularized originalism. abbott said scalia set an example for citizens. mr. obama was informed of scalia's death saturday afternoon. cbs news tweeted scalia was found dead at a west texas ranch. scalia was the longest-serving justice on the current supreme court at the time of his death.	supreme court justice antonin scalia was found dead saturday at a resort outside of marfa , texas , kvia reports .according to the san antonio express-news , the 79-year-old appears to have died from natural causes . scalia was the longest-serving justice currently on the supreme court , having been nominated by ronald reagan in 1986. in a statement , texas gov. greg abbott called scalia " a man of god, a patriot, and an unwavering defender of the written constitution and the rule of law ." we mourn his passing , and we pray that his successor on the supreme court



(a) Rouge-1 score **(b)** S+WMD score

Figure 4: Performance of different methods on Scientific-NLP dataset varying length of output summary.

5 RELATED WORK

Multiple Documents Summarization. The previous study of unsupervised multiple document summarization mainly spans in three categories: 1) graph-based ranking algorithms 2) summarization via submodular optimization 3) clustering based summarization. Graph-based ranking algorithms can be traced back to TextRank [19] and LexRank [8], where both methods construct text graph based on sentence similarity or phrase co-occurrence and determine the salience of sentence or phrase by eigenvector centrality like PageRank. ClusterRank [10] clusters similar sentences and uses clusters as nodes in the text graph. The family of submodular optimization [6, 16] towards documents summarization is designed to balance between summarization coverage and dispersion with a sub-optimal approximation. Recent advance [31] combines the strength of graph-ranking and submodular selec-

Table 4: Qualitative analysis on Scientific-NLP. We compare SUMDocS and our ablations without background corpus. We present the different top-scored phrases selected by each method and their appearance in the ground truth summary.

	SUMDocS	SUMDocS-**NoBkg**	ground truth
keywords	left-to-right, representation, mlm, context, bidirectional, state-of-the-art, left, feature-based	model, fine-tuning, score, f1, final, pre-trained, answer, embeddings	N.A.
summary	Unlike left-to-right language model pre-training, the mlm objective enables the representation to fuse the left and the right context, which allows us to pretrain a deep bidirectional Transformer. both bert-base and bertlarge outperform all systems on all tasks by a substantial margin , obtaining 4.5% and 7.0% respective average accuracy improvement over the prior state-of-the-art. input/output representations to make bert handle a variety of down-stream tasks , our input representation is able to unambiguously represent both a single sentence and a pair of sentences in one token sequence.	in this section, we explore the effect of model size on fine-tuning task accuracy. additionally, this model was pre-trained without the nsp task. in this section, we present bert fine-tuning results on 11 nlp tasks. during pre-training, the model is trained on unlabeled data over different pre-training tasks. we use a simple approach to extend the squad v1.1 bert model for this task. the final model achieves 97% - 98% accuracy on nsp. in fact, our single bert model outperforms the top ensemble system in terms of f1 score. gpt uses a sentence separator and classifier token which are only introduced at fine-tuning time;	we introduce a new language representation model called bert , which stands for bidirectional encoder representations from transformers . unlike recent language representation models, bert is designed to pretrain deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers . as a result , the pre-trained bert model can be finetuned with just one additional output layer to create state-of-the-art models for a wide range of tasks , such as question answering and language inference , without substantial task specific architecture modifications .

tion. Clustering based summarization methods [11] origin from keyphrase extraction task, which groups the topical keyphrases using techniques like hierarchical clustering. Many of them [34] introduce term or sentence relatedness scoring as a preprocessing step.

Recently, deep neural network based supervised summarization methods start to achieve competitive performance on supervised single document summarization. The most relevant ones to our work are extractive summarization methods, which model the summarization as classification [29, 21] and reinforcement learning problem [23]. There are also abstractive algorithms [22, 25] train a neural seq2seq model to generate summary. PointerNetwork [27] mix abstractive generation and extractive copy mechanism. Regarding multi-document summarization, the excessive length of the articles poses challenge to these methods. Most of the successes landed on single document summarization with desirable training data. Several [9, 14] recent multi-document summarization methods adopt the traditional extractive sentence ranking to select the importance sentences and reduce the space complexity. However, as shown in our experiments, the performance of these models drops a lot when applied on corpus that is different from its training data.

Context-aware Summarization. Similar with the proposed method, various researchers are motivated to improve the quality of summarization with context or background knowledge. Based on existing ontologies, *e.g.* wordnet, wikipedia, yago, [26, 1, 12] first map the sentences onto the ontology node and use either hand-crafted features or graph-based summarization objective to select the summary. Besides directly matching the sentences, recent studies start to use vector representations to score the sentences [13] or jointly learn the summarization and classification [2].

The introduction of neural language modeling [7] facilitates the downstream task learning with rich semantic information in the pre-trained encoders. It is also used in text summarization as a form of contextual information [33, 17]. However, the general language model may not adapt to the specific domain like scientific papers without supervised fine-tuning. SUMDocS captures the domain-specific information from the unlabeled corpus via category name guided embedding.

6 Conclusions and Future Work

In this paper, we proposed SUMDocS that identifies surrounding documents from background corpus and summarizes the target documents comparatively. We also validate the benefits of introducing background corpus on both lexical and semantic metric. In the future, it is promising to incorporate surrounding documents into abstractive summarization model like seq2seq .

7 Acknowledgement

Research was sponsored in part by US DARPA KAIROS Program No. FA8750-19-2-1004 and SocialSim Program No. W911NF-17-C-0099, National Science Foundation IIS-19-56151, IIS-17-41317, IIS 17-04532, and IIS 16-18481, and DTRA HDTRA11810026.

References

[1] E. Baralis, L. Cagliero, S. Jabeen, A. Fiori, and S. Shah. Multi-document summarization based on the yago ontology. *Expert Systems with Applications*.

[2] Z. Cao, W. Li, S. Li, and F. Wei. Improving multi-document summarization via text classification. In *AAAI*.

[3] J. G. Carbonell and J. Goldstein. The use of mmr and diversity-based reranking for reodering documents and producing summaries. In *SIGIR*.

[4] E. Clark, A. Celikyilmaz, and N. A. Smith. Sentence mover's similarity: Automatic evaluation for multi-sentence texts. In *ACL*, 2019.

[5] A. Cohan, F. Dernoncourt, D. S. Kim, T. Bui, S. Kim, W. Chang, and N. Goharian. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint*, 2018.

[6] A. Dasgupta, R. Kumar, and S. Ravi. Summarization through submodularity and dispersion. In *ACL*, 2013.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 2019.

[8] G. Erkan and D. R. Radev. Lexrank: Graph-based lexical centrality as salience in text summarization. *JAIR*, 2004.

[9] A. R. Fabbri, I. Li, T. She, S. Li, and D. Radev. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. In *ACL*.

[10] N. Garg, B. Favre, K. Reidhammer, and D. Hakkani-Tür. Clusterrank: a graph based method for meeting summarization. In *INTERSPEECH*, 2009.

[11] M. Grineva, M. Grinev, and D. Lizorkin. Extracting key terms from noisy and multitheme documents. In *WWW*, 2009.

[12] L. Hennig, W. Umbrath, and R. Wetzker. An ontology-based approach to text summarization. In *WI-IAT*, volume 3, 2008.

[13] C. Khatri, G. Singh, and N. Parikh. Abstractive and extractive text summarization using document context vector and recurrent neural networks. 2018.

[14] L. Lebanoff, K. Song, and F. Liu. Adapting the neural encoder-decoder framework from single to multi-document summarization. In *EMNLP*, 2018.

[15] C.-Y. Lin. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*, 2004.

[16] H. Lin and J. Bilmes. A class of submodular functions for document summarization. In *ACL*, 2011.

[17] Y. Liu and M. Lapata. Text summarization with pretrained encoders. In *EMNLP*, 2019.

[18] Y. Meng, J. Huang, G. Wang, Z. Wang, C. Zhang, Y. Zhang, and J. Han. Discriminative topic mining via category-name guided text embedding. In *WWW*, 2020.

[19] R. Mihalcea and P. Tarau. Textrank: Bringing order into text. In *EMNLP*, 2004.

[20] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS*, 2013.

[21] R. Nallapati, F. Zhai, and B. Zhou. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *AAAI*, 2017.

[22] R. Nallapati, B. Zhou, C. dos Santos, cC. Gulccehre, and B. Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *CONLL*, 2016.

[23] S. Narayan, S. B. Cohen, and M. Lapata. Ranking sentences for extractive summarization with reinforcement learning. In *NAACL*, 2018.

[24] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—i. *Mathematical programming*, 1978.

[25] A. M. Rush, S. Chopra, and J. Weston. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*, 2015.

[26] Y. Sankarasubramaniam, K. Ramanathan, and S. Ghosh. Text summarization using wikipedia. *Information Processing & Management*, 2014.

[27] A. See, P. J. Liu, and C. D. Manning. Get to the point: Summarization with pointer-generator networks. In *ACL*, 2017.

[28] J. Shang, J. Liu, M. Jiang, X. Ren, C. R. Voss, and J. Han. Automated phrase mining from massive text corpora. *TKDE*, 2018.

[29] J. Shi, C. Liang, L. Hou, J. Li, Z. Liu, and H. Zhang. Deepchannel: Salience estimation by contrastive learning for extractive document summarization. In *AAAI*, 2019.

[30] F. Tao, H. Zhuang, C. W. Yu, Q. Wang, T. Cassidy, L. M. Kaplan, C. R. Voss, and J. Han. Multi-dimensional, phrase-based summarization in text cubes. *IEEE Data Eng. Bull.*, 2016.

[31] A. Tixier, P. Meladianos, and M. Vazirgiannis. Combining graph degeneracy and submodularity for unsupervised extractive summarization. In *Workshop on New Frontiers in Summarization*, 2017.

[32] W. Xiao and G. Carenini. Extractive summarization of long documents by combining global and local context. In *EMNLP*, 2019.

[33] X. Zhang, F. Wei, and M. Zhou. Hibert: Document level pre-training of hierarchical bidirectional transformers for document summarization. In *ACL*, 2019.

[34] Y. Zhang, Y. Xia, Y. Liu, and W. Wang. Clustering sentences with density peaks for multi-document summarization. In *NAACL-HLT*, 2015.