Improving Unsupervised Extractive Summarization by Jointly Modeling Facet and Redundancy

Xinnian Liang, Jing Li, Shuangzhi Wu, Mu Li, and Zhoujun Li

***Abstract*—Unsupervised extractive summarization aims to extract salient sentences from documents without labeled corpus. Existing methods are mostly graph-based by computing sentence centrality. These methods have two main problems: facet bias and redundant problems. Facet bias problem leads summarization models tend to select sentences within the same facet, which often leads to the ignoring of other vital facets, especially on long-document and multi-documents. First, to address the facet bias problem, we proposed a novel Facet-Aware centrality-based Ranking model (FAR). We let the model pay more attention to different facets by introducing a sentence-document weight. The weight is added to the sentence centrality score. FAR can alleviate redundancy to some extent. Then, to further reduce redundancy, we proposed a novel Redundancy- and Facet-Aware Ranking model (RFAR) which jointly models facet and redundancy by incorporating Determinantal Point Process (DPP) into the previous proposed FAR. We evaluate our FAR and RFAR on a wide range of summarization tasks that include 8 representative benchmark datasets. Experimental results show that FAR and RFAR consistently outperforms strong baselines, especially in long- and multi-document scenarios, and even perform comparably to some supervised models. Besides, we find that our methods can alleviate the position bias problem.**

***Index Terms*—Determinantal point process, facet-bias problem, redundancy, unsupervised extractive summarization.**

I. Introduction

DOCUMENT summarization is the task of transforming a long document or multi-documents into a shorter version while retaining the most salient content [1]. Most summarization methods can be divided into abstractive and extractive. Abstractive methods generate summary word by word like human writing based on the understanding of document, making generated summary more fluent and human-like. Extractive methods extract several sentences from documents as summary. While abstractive methods can be more concise and flexible, extractive methods can guarantee correct grammar and are more consistent factually [2].

Existing extractive or abstractive methods are mostly in supervised fashion which rely on large amounts of annotated corpora [3]–[9]. However, this is not available for different summarization styles, domains, and languages. Fortunately, recent work has shown successful practices on unsupervised extractive summarization [10]–[12]. Compared with supervised ones, unsupervised methods 1) remove the dependency on large-scale annotated document-summary pairs; 2) are more general for various scenarios.

Graph-based models are commonly used in unsupervised extractive methods [13]–[15]. They represent documents as a graph. Nodes in the graph represent sentences in documents, edges mean relation between sentences. The core idea of graph-based model is to compute node centrality by degree or PageRank-based [16] algorithms [14], [15] and select top-ranked sentence as summary. Zheng *et al.* [10] proposed a directed centrality-based method named PacSum by assuming that the contribution of any two nodes to their respective centrality is influenced by their relative position in a document. Dong *et al.* [12] further improved PacSum by incorporating hierarchical and positional information into the directed centrality method. The theoretical basis of centrality-based models is that the more similar a sentence is to others, the more vital it is [13]. They usually work well for documents with a single facet (i.e. topic, aspect)[1]. However, there is always more than one facet, especially in long-document or multi-documents. Fig. 1 shows an example of a document with 3 facets. We highlight the key phrases of each facet in different colors. Current centrality-based models often select sentences from one facet which is supported by more similar sentences. For example, the baseline model selects 3 sentences from the facet 1. We call this the facet bias problem. We can see that our model can cover three facets in this example. However, the baseline and our model all tend to select position-forward sentences due to the layout bias of existing summarization datasets, which need to be solved in future work.

Fig. 2 shows an intuitive explanation of the facet bias problem. The nodes are sentence representations, the star is the document representation, and rhombuses are the centers of selected summary sentences. Sentences supporting the same facet appear in the same circle. Recent centrality-based models tend to select

Manuscript received June 25, 2021; revised September 24, 2021 and November 1, 2021; accepted December 13, 2021. Date of publication December 28, 2021; date of current version May 2, 2022. This work was supported in part by the National Natural Science Foundation of China under Grants U1636211, 61672081, and 61370126, in part by 2020 Tencent Wechat Rhino-Bird Focused Research Program, and in part by the Fund of the State Key Laboratory of Software Development Environment under Grant SKLSDE-2021ZX-18. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Nancy F. Chen. *(Corresponding author: Zhoujun Li.)*

Xinnian Liang and Zhoujun Li are with the State Key Lab of Software Development Environment, School of Computer Science, and Engineering (SCSE), Beihang University, Beijing 100089, China (e-mail: xnliang@buaa.edu.cn; lizj@buaa.edu.cn).

Jing Li is with the School of Information, Renmin University of China, Beijing 100089, China (e-mail: lijinginfo@ruc.edu.cn).

Shuangzhi Wu and Mu Li are with the Department of Tencent Cloud Xiaowei, Beijing 100089, China (e-mail: frostwu@tencent.com; ethanlli@tencent.com).

Digital Object Identifier 10.1109/TASLP.2021.3138673

[1]The concept of the facet is very similar to topic and aspect. We employ the definition of facet: each summary sentence contains one facet [17].

Document

Facet 1: Lampard was fired.
1. As Chelsea's winter had turned bleak, as whispers that Lampard, its inexperienced coach, might be drifting toward the edge grew louder …
2. Not for the manager — that Lampard was fired so soon after he was given such public backing illustrates, quite neatly, how little power fans have — but for the public itself.
3. Chelsea might, in truth, have fired Lampard earlier. His colleagues, certainly, have been fearing it for weeks.
4. That was Sunday afternoon. He was fired on Monday morning …

Facet 2: Fans support Lampard.
5. "In Frank We Trust," it read, white letters on a blue field … And underneath, three simple words: "Then. Now. Forever." …
6. But none — not even Mourinho — have retained the support of the fans quite so unanimously as Lampard.
7. Lampard's association with Chelsea runs long and deep enough that he has deep-seated, well-established connections with Chelsea's fans.

Facet 3: Many managers were fired of Abramovich's ruthless impatience.
8. Lampard was not the first manager at Roman Abramovich's Chelsea to come under what seemed, on the surface, to be an undue, premature sort of pressure. …
9. Some of those who have gone before Lampard have done so with sympathy, perceived as victims of Abramovich's ruthless impatience.

Baseline

2. Not for the manager — that Lampard was fired so soon after he was given such public backing illustrates, quite neatly, how little power fans have — but for the public itself.
3. Chelsea might, in truth, have fired Lampard earlier. His colleagues, certainly, have been fearing it for weeks.
4. That was Sunday afternoon. He was fired on Monday morning …

Our Model

2. Not for the manager — that Lampard was fired so soon after he was given such public backing illustrates, quite neatly, how little power fans have — but for the public itself.
6. But none — not even Mourinho — have retained the support of the fans quite so unanimously as Lampard.
8. Lampard was not the first manager at Roman Abramovich's Chelsea to come under what seemed, on the surface, to be an undue, premature sort of pressure. …

Gold Reference

3. Chelsea might, in truth, have fired Lampard earlier. His colleagues, certainly, have been fearing it for weeks.
6. But none — not even Mourinho — have retained the support of the fans quite so unanimously as Lampard.
9. Some of those who have gone before Lampard have done so with sympathy, perceived as victims of Abramovich's ruthless impatience.

Fig. 1. An example from New York Times. We selected part of the vital sentences from the source document to show in this table. "..." refers to the omissions of context sentences due to the space limitation.

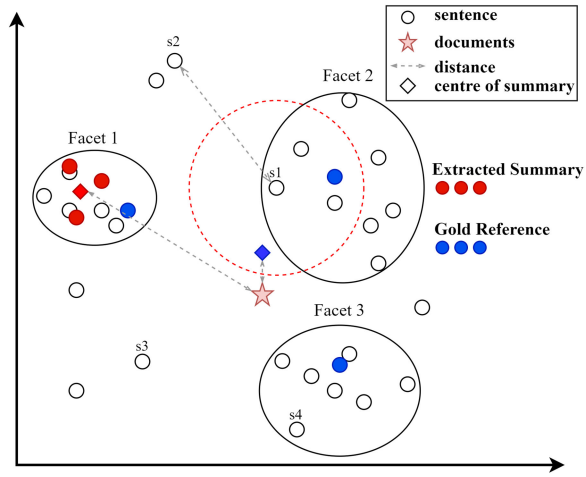


Fig. 2. Visualization of facet bias. Nodes refer to sentence representations and star is the document representation. Black solid circles mean facets. Red dashed circle means threshold in Section III-A. The dashed bi-direction arrows denote the sentence similarities. Sentences s1-4 are used for explaining the filter of negligible sentences in Section III-A. The centrality score of s1 should not consider sentences outside the red dashed circle (e.g. s2), which is highly dissimilar with s1.

sentences from facet 1 (red nodes). Because these sentences are more similar to each other which leads to a higher centrality score of all nodes in facet 1. However, the ideal summary should consist of support sentences from different facets (blue nodes). To address the facet bias problem, in this paper, we proposed a facet-aware centrality-based model, which is called Facet-Aware Rank (FAR). First, we introduce a modified graph-based ranking method to filter irrelevant sentences. Then we encode the whole document into vector space, which is used to capture all facets in the document. For each candidate summary, we calculate a similarity score between the summary sentences and the document. This sentence-document similarity aims at measuring the relevance between summary and document, whereas the sentence centrality measures the sentence-level importance. In the ranking phase, we combine the sentence-document similarity and the sentence centrality to guarantee selected sentences are important and cover all facets. As shown in Fig. 2, by incorporating the sentence-document similarity, we are more likely to select the blue ones, that are closer to the star, instead of the red ones. Experiments confirm that the performance gains indeed come from alleviating the facet bias problem.

Besides, we surprisingly find that our FAR model can tackle redundancy in summary to some extent, which is a widely focused problem in extractive methods. To further reduce redundancy, we proposed a novel ranking model which jointly models facet and redundancy by introducing the Determinantal Point Process (DPP) into our previous proposed FAR model to balance the quality (importance) and redundancy of extracted summary. The whole model is called the Redundancy- and Facet-Aware Ranking model (RFAR), which can take the importance, relevance, and redundancy of selected summary sentences into consideration at the same time. We evaluate our methods on eight representative datasets. The results show that our models can surpass strong unsupervised baselines on most datasets and are comparable to supervised models on some datasets. FAR achieved stable performance with an automatic evaluation metric. Compared with FAR, RFAR achieved better human-evaluation results and still maintained great automatic evaluation results.

Our contributions can be summarized as follows:

- We discovered the facet bias problem in graph-based models which make the system tend to select sentences around one facet. We proposed a novel unsupervised Facet-Aware Ranking model (FAR) to tackle the facet bias problem.
- To jointly solve the redundant problem, we proposed a novel Redundancy- and Facet-Aware Ranking model (RFAR) which introduces Determinantal Point Process (DPP) into our previous proposed FAR model and can take the importance, relevance, and redundancy of selected summary sentences into consideration at the same time.
- Our models surpass strong unsupervised baselines on most datasets and are comparable to supervised models on some datasets. We also confirm that the performance gains of RFAR and FAR indeed come from alleviating facet bias and redundant problems.

- We conduct many studies to analyze the characteristics of our method on existing datasets and find that our methods can alleviate the position bias problem.

II. Background: Graph-Based Ranking

Given a document D, it contains a set of sentences $\{s_1, .., s_i, .., s_j, .., s_n\}$. Graph-based algorithms treat D as a graph $\mathcal{G} = (V, E)$. $V = \{v_1, v_2, \ldots, v_n\}$ is the vertex set where v_i is the representation of sentence s_i. E is the edge set, which is an $n \times n$ matrix. Each $= \{e_{i,j}\} \in E$ denotes the weight between vertex v_i and v_j.

The key idea of graph-based ranking is to calculate the centrality score of each sentence (or vertex). Traditionally, this score is measured by degree or ranking algorithms [14], [15] based on PageRank [16]. Then the sentences with the top score are extracted as a summary. The undirected graph algorithm computes the sentence centrality score as follows:

$$\texttt{Centrality}(s_i) = \sum_{j=1}^{N} e_{ij} \tag{1}$$

This is based on the assumption that the contribution of the sentence's importance in the document is not affected by the order of the sentence. In contrast, the directed graph-based ranking algorithm considers the positional feature, based on the assumption that the previous content of the current sentence and the later contexts have a different impact on the current sentence's centrality score [18]. Then (1) is reformulated as

$$\texttt{DC}(s_i) = \lambda_1 \sum_{i>j} e_{ij} + \lambda_2 \sum_{i<j} e_{ij} \tag{2}$$

Where $\lambda_1 + \lambda_2 = 1$. Hyper-parameters λ_1 and λ_2 were used to adjust the influence of previous and last content. Our method is built based on the directed graph-based ranking algorithm.

III. Facet-Aware Centrality-Based Model

A. Modified Directed Graph-Based Ranking

We propose a variation of directed graph-based ranking in this section. We modify (2) in terms of filtering negligible sentences. We take s_1 in Fig. 2 as an example to give an intuitive explanation. There usually exist many unrelated sentences especially in long documents for s_1 i.e. s_2, s_3, s_4. As shown in (2), all these sentences have a contribution in computing s_1's centrality score. We regard sentences like them as noise of s_1 and propose a modified directed graph-based ranking to filter them. To this end, we simply introduce a threshold ϵ to (2). For s_1, ϵ can be seen as a diameter, s_1 is the center. The centrality score of s_1 only considers nodes in a red dashed circle. We further rewrite 2 as:

$$\begin{aligned}\texttt{DC}(s_i) = \lambda_1 \sum_{i>j} \texttt{Max}((e_{ij} - \epsilon), 0) \\ + \lambda_2 \sum_{i<j} \texttt{Max}((e_{ij} - \epsilon), 0)\end{aligned} \tag{3}$$

where $\epsilon = \beta \cdot (\max(e_{ij}) - \min(e_{ij}))$. β is a Hyper-parameter to control the scale of diameter. As shown in (3), if the similarity between s_i and s_j is lower than ϵ, s_j is neglected. We find this modification is very effective but the model is very sensitive to the selection of β, so we carefully tune β on the development set. We finally rank and select sentences with (4).

$$\texttt{summary} = \texttt{topK}(\{\texttt{DC}(s_i)\}_{i=1,..,n}) \tag{4}$$

Where top-ranked k sentences will be extracted as the summary and k is pre-defined with the average length of summary in training data.

B. Facet-Aware Centrality Scoring

In this section, we introduce how to implement (3) and how we incorporate the facet into centrality-based ranking in detail. We propose a simple method to model the facets in a document by a special representation based on the whole document.

Specifically, based on (4), we add a sentence-document similarity, which computes the similarity between sentences in candidate summary C and document to measure the relevance between candidate summary and document. Candidate summary is pre-selected sentences from top-ranked K sentences with score $\texttt{DC}(s_i)$ to reduce search range. We combine sentence-document similarity with sentence centrality and obtain the best candidate summary by (5).

$$\texttt{summary} = \texttt{arg}\max_{\mathcal{C}}(\texttt{sim}(d, \hat{v}) \cdot \sum_{s_i \in \mathcal{C}} \texttt{DC}(s_i)^{\alpha}) \tag{5}$$

where α is a hyper-parameter to control the influence of directed centrality. $\texttt{sim}(d, \hat{v})$ refers to the sentence-document similarity, where d is the document representation and $\hat{v}$ is the candidate summary representation. $\hat{v}$ is obtained by $\frac{\sum_{i \in C}(v_i)}{|C|}$ which is the mean representation of summary sentences. We select the cosine similarity for $sim(\cdot)$.

A candidate summary C is the subset of top-ranked K sentences after ranking with $\texttt{DC}(s_i)$, which satisfy the following two conditions: 1) the length of sentences in candidate summary is predefined L, which is related to the summary length of dataset training data; 2) the total length of top-ranked K sentences is $t \times L$, where t is empirically set as 3. For the sentence representations v_i, we employ BERT as the encoder which maps each word into a hidden state. Specifically, the sentence representation v_i is obtained by $\texttt{sigmoid}(h_i)$, where h_i is the hidden state of "[CLS]". Each e_{ij} in E is calculated by the dot product of the two sentences $v_i^{\top} v_j$. For document representation, we first collect all the sentence representations $\{v_1, v_2, \ldots, v_n\}$. To compress all the valuable information in the document, we apply a max-pooling function to sentence representations. The document representation d is computed as:

$$d = \texttt{Maxpooling}(\{v_1, v_2, \ldots, v_n\}) \tag{6}$$

Empirically, the max-pooling function can represent document semantic information better than the mean-pooling function or the hidden state of "[CLS]". We will prove this through contrast experiments.

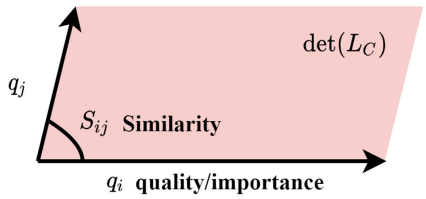


Fig. 3. The DPP specifies the probability of squared volume of the space spanned by sentence i and j vectors.

IV. Redundancy- and Facet-Aware Ranking Model

In this section, we will introduce the incorporation of the determinantal point process for modeling redundancy.

A. The DPP Framework

Let $\mathcal{C}$ as all subsets of document $D = \{s_1, \ldots, s_n\}$. The goal of the DPP framework is to identify the best subset $C \in \mathcal{C}$ as an extracted summary of the document. A determinantal point process [19] defines a probability measure over all subsets $C \in \mathcal{C}$ s.t.

$$\mathcal{P}(C; L) = \frac{\det(L_C)}{\det(L + I)}$$
$$\det(L + I) = \sum_{C \in \mathcal{C}} \det(L_C) \tag{7}$$

where $\det(\cdot)$ is the determinant of a matrix; I is the identity matrix; $L \in \mathbb{R}^{n \times n}$ is a positive semi-definite matrix and L_{ij} means the correlation between sentence i and sentence j; L_C is a sub-matrix of L which only contain entries in subset C. The probability of subset C to be summary can be obtained by (7). Kulesza and Taskar [20] provided a composition of L: $L_{ij} = q_i \cdot S_{ij} \cdot q_j$ where $q_i \in \mathbb{R}^+$ is a positive real number indicating the importance or quality of sentence i and S_{ij} is the similarity between sentence i and j. For example, if candidate summary $C = \{s_i, s_j\}$ only contains 2 sentences, we can compute the probability of it by:

$$\begin{aligned}\mathcal{P}(C; L) &\propto \det(L_C) \\ &= \begin{vmatrix} q_i S_{ii} q_i & q_i S_{ij} q_j \\ q_j S_{ji} q_i & q_j S_{jj} q_j \end{vmatrix} \\ &= q_i^2 \cdot q_j^2 \cdot (1 - S_{ij}^2)\end{aligned} \tag{8}$$

From (8), we can see that if sentence i is very important, denoted by q_i, then any summary containing sentence i will have a high probability. However, if sentence i and j are similar to each other, denoted by S_{ij}, then summary contains them will have low probability. A proper summary C should have salient sentences and keep diversity among them.

From the geometric view, the quality of the sentence can be seen as the length of vector i and j; the similarity between sentences can be seen as an angle between vectors. The final probability is the squared volume of space spanned by sentence vector i and j as shown in Fig. 3.

B. Redundancy- and Facet-Aware Ranking Model

In the previous section, we introduced the DPP framework. To jointly model facet and redundancy, we can replace the measure of sentence quality (importance) q_i in (8) with our directed centrality score $\texttt{DC}(s_i)$ and the similarity between sentences S_{ij} can be computed with cosine similarity between sentence vectors. Finally, we can rewrite (5) as (9).

$$\texttt{summary} = \arg\max_{C \in \mathcal{C}} (\texttt{sim}(d, \hat{v}) \cdot \mathcal{P}(C; L)^{\alpha}) \tag{9}$$

Where $\texttt{sim}(d, \hat{v})$ measures the relevance between candidate summary C and document; $\mathcal{P}(C; L)$ measures the importance and redundancy of candidate summary sentences.

C. Improved Sentence Representation

Sentence representations play a crucial role in our ranking model. The previous study shows that improving the quality of sentence representations helps improve the ranking performance [10], [12]. We post-train BERT on a sentence-level task constructed based on the corpus of a specific task. The idea is that its representation is affected not only by the words in it, but also the sentences around it. For a sentence in a document, we take its previous sentence and its following sentence to be positive examples and random sample sentences from documents as negative examples. The objective function follows that used in [21]. Specifically, for sentence s_i, a positive sentence s_j, and a negative sentence s_k, the BERT is trained to minimize the following equation:

$$\max(\| v_i - v_j \| - \| v_i - v_k \| + \mu, 0) \tag{10}$$

where v is the sentence representation, and μ is margin which ensures that v_j is at least μ closer to s_i than s_k. The hidden state vector of "[CLS]" is used as sentence representations and we set μ to 1 following [21] in post-training phase.

V. Experiments

A. Datasets

We introduce eight datasets in this section.

[1] CNN/DM dataset contains 93 k articles from CNN, and 220 k articles from Daily Mail newspapers [22]. We use the non-anonymous version. Following [10], documents whose length of summaries are shorter than 30 tokens are filtered out.

[2] NYT dataset contains articles published by the New York Times between January 1, 1987 and June 19, 2007 [23]. The summaries are written by library scientists. Different from CNN/DM, salient sentences distribute evenly in an article [24]. We filter out documents whose length of summaries are shorter than 50 tokens [10].

[3] MultiNews dataset consists of news articles and human-written summaries. The dataset is the first large-scale Multi-Documents Summarization (MDS) news dataset and comes from a diverse set of news sources (over 1500 sites) [25].

[4-5] arXiv&PubMed datasets are two long document datasets of scientific publications from arXiv.org (113 k) and PubMed (215 k) [26]. The task is to generate the abstract from the paper body.

TABLE I
INFORMATION OF DATASETS

Datasets	Sources	Type	#Pairs			#Tokens		#Sentences	
			Train	Valid	Test	Doc.	Sum.	Doc.	Sum.
CNN/DM	News	SDS	287,227	13,368	11,490	788	63	33	3
NYT	News	SDS	36,735	5,531	4,375	1,291	80	50	3
MultiNews	News	MDS	44,972	5,622	5,622	2,104	264	167	7
arXiv	Scientific Paper	LDS	202,914	6,436	6,440	4,938	220	205	10
PubMed	Scientific Paper	LDS	117,108	6,631	6,658	3,016	203	107	8
WikiSum	Wikipedia	MDS	1,579,360	38,144	38,144	2,800	139	184	8
WikiHow	Wikipedia	SDS	157,252	5,599	5,577	581	63	30	4
BillSum	US Legislation	LDS	17,054	1,895	3,269	2,148	209	168	10

The Data in Doc. And Sum. Indicates the Average Length of Document and Summary Respectively. SDS Represents Single-Document Summarization, MDS Represents Multi-Documents Summarization, and LDS Represents Single Long Document Summarization (#tokens of Document $\geq$ 3,000).

TABLE II
BEST SETTINGS OF HYPER-PARAMETERS FOR DIFFERENT DATASETS

Datasets	α	β	λ_1	λ_2
CNN/DM	1	0.0	0.7	0.3
NYT	1	0.6	0.6	0.4
arXiv	2	0.7	0.5	0.5
PubMed	2	0.3	0.5	0.5
MultiNews	1	0.4	0.5	0.5
WikiSum	1	0.0	0.5	0.5
BillSum	1	0.5	0.5	0.5
WikiHow	1	0.8	0.5	0.5

[6] WikiSum dataset is a multi-documents summarization dataset from Wikipedia [27]. We use the version provided by [6], which selects ranked top-40 paragraphs as input. For this dataset, we filter out documents whose summary length is less than 100 tokens. After the process, WikiSum test set contains 15,795 examples and the average length of summaries is 198.

[7] WikiHow dataset is a large-scale dataset of instructions from the online WikiHow.com website [28]. The task is to generate the concatenated summary-sentences from the paragraphs.

[8] BillSum dataset contains US Congressional bills and human-written reference summaries from the 103rd-115th (1993-2018) sessions of Congress [29].

These datasets differ in scale, domain and task type. We collect details of the 8 corpus in Table I.

B. Implementation Details and Metrics

Our proposed two models RFAR and FAR have 4 hyper-parameters and the best set of them are chosen from the following setting: $\alpha \in \{1, 2\}$, $\beta \in \{0.0, 0.1, \ldots, 0.9\}$, $\lambda_1 + \lambda_2 = 1, \lambda_1 \in \{0.0, 0.1, \ldots, 1.0\}$. In most case, FAR with the default setting $(\alpha = 1, \beta = 0.5, \lambda_1 = 0.5, \lambda_2 = 0.5)$ can achieve satisfied performance on all datasets. We select best hyper-parameters by sampling 1,000 examples from the validation set [10]. The best settings of hyper-parameters is shown in Tables II.

The implementation of encoder model is based on the PyTorch implementation of BERT.[2] In the post-training stage, we employ a basic BERT model to initialize our sentence encoder. We use Adam [30] as our optimizer with a learning-rate of $2e^{-5}$. We sample documents from the training set of all datasets. The max length of the input sentence is set to 60. A linear warm-up for the first 10% of steps followed by a linear decay to 0 is used. The BERT encoder is post-trained on six Tesla V100 GPUs.

We employ `ROUGE-1.5.5.pl` script[3] to evaluated summarization quality automatically with ROUGE F1 [31]. We report ROUGE-1/2/L and BertScore [32] to measure the quality of extracted summaries. Besides, we also do a human evaluation for the facet bias and redundancy of extracted summaries.

C. Results of ROUGE-1/2/L on 8 Datasets

Tables III– V report the results of datasets with 3 types. In each table, we present the results of **Oracle** and supervised extractive models in the first block. **Oracle** can be seen as the upper bound of extractive models, which extracts gold standard summaries by greedily selecting sentences to optimize the mean of ROUGE-1 and ROUGE-2 [4]. We compare our approach with strong unsupervised baselines **Lead**, **TextRank** [14], **LexRank** [15], **MMR** [33] in the second block of each table. We also implement a strong baseline **BERT+MMR** to compare with our model. **Lead** selects the first k tokens as a summary. We also report previous best centrality-based model **PacSum** [10] in the second block of each table. The results of our FAR and RFAR are reported in the third block.

Overall, our FAR and RFAR outperform all unsupervised baselines on most datasets, especially on long-document and multi-documents datasets. Results prove that our models are more generalized than them for different types, domains datasets. We can see that the model of redundancy with DPP improves performance obviously on multi-documents datasets. We also find that the performance of Lead on many datasets (i.e. CNN/DM, MultiNews, WikiSum) is very high, which means these datasets have the position bias (lead bias) problem. The representation from the BERT model also improves the performance of MMR, which makes BERT+MMR achieve satisfactory results on some datasets. We do not reproduce TextRank with BERT due to it can not have better performance by employing sentence representations from BERT [10].

1) Results on SDS: Table III reports the results on single document summarization (SDS) datasets CNN/DM, NYT and WikiHow. **REFRESH** [34] and **BertExt** [7] are supervised extractive models. **STAS** [11] is state-of-the-art unsupervised

[2][Online]. Available: https://github.com/huggingface/transformers

[3][Online]. Available: https://github.com/andersjo/pyrouge

TABLE III
RESULTS ON SINGLE DOCUMENT SUMMARIZATION DATASETS: CNNDM, NYT AND WIKIHOW TEST SETS

Methods	CNN/DM			NYT			WikiHow		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
supervised extractive methods									
Oracle	52.59	17.62	36.67	61.63	41.54	58.11	39.80	14.85	36.90
REFRESH	41.30	18.40	35.70	**41.30**	**22.00**	**37.80**	-	-	-
BertExt	**43.25**	**20.24**	**39.63**	-	-	-	**30.31**	**8.17**	**28.24**
unsupervised extractive methods									
Lead	40.49	17.66	36.75	35.50	17.20	32.00	24.31	5.52	22.53
TextRank	33.85	13.61	30.14	33.24	14.74	29.92	21.64	5.34	19.68
LexRank	34.68	12.82	31.12	30.75	10.49	26.58	25.46	5.89	23.63
MMR	31.63	10.02	28.55	27.16	6.41	25.32	22.02	4.40	20.22
BERT+MMR	33.12	11.05	29.56	28.18	8.11	24.20	24.15	6.48	23.96
PacSum	40.70	17.80	36.90	41.40	21.70	37.50	-	-	-
PacSum (Ours)	40.69	17.82	36.91	41.37	21.65	37.35	27.46	6.13	25.40
STAS	**40.90**	**18.02**	**37.21**	41.46	21.80	37.57	-	-	-
FAR	40.83	17.85	36.91	**41.61**	**21.88**	**37.59**	**27.54**	**6.17**	**25.46**
RFAR	40.64	17.49	36.01	41.42	21.68	37.36	27.38	6.02	25.37

TABLE IV
RESULTS ON LONG DOCUMENT SUMMARIZATION DATASETS: ARXIV, PUBMED AND BILLSUM TEST SETS

Methods	arXiv			PubMed			BillSum		
	R-1	R-2	R-L	R-1	R-2	R-L	R-1	R-2	R-L
supervised extractive methods									
Oracle	53.88	23.05	34.9	55.05	27.48	38.66	56.22	38.77	51.25
SummaRuNNer	42.81	16.52	28.23	43.89	18.78	30.36	-	-	-
GlobalLocalCont	**43.62**	**17.36**	**29.14**	**44.85**	**19.7**	**31.43**	-	-	-
unsupervised extractive methods									
Lead	33.66	8.94	22.19	35.63	12.28	25.17	35.10	16.76	30.31
TextRank	24.38	10.57	22.18	38.66	15.87	34.53	36.10	15.00	30.35
LexRank	33.85	10.73	28.99	39.19	13.89	34.59	38.28	16.02	32.44
MMR	29.75	6.14	26.41	37.65	10.61	33.71	36.73	12.45	32.13
BERT+MMR	33.41	7.87	29.42	38.13	11.50	34.00	36.43	13.40	31.95
PacSum	39.33	12.19	34.18	39.79	14.00	36.09	38.34	16.64	33.36
HipoRank	39.34	12.56	34.89	**43.58**	**17.00**	**39.31**	-	-	-
FAR	40.92	13.75	35.56	41.98	16.74	37.58	38.37	16.69	33.40
RFAR	**41.02**	**13.96**	**35.87**	42.23	16.98	38.53	**38.76**	**17.04**	**33.63**

TABLE V
RESULTS ON MULTI-DOCUMENT SUMMARIZATION DATASETS: MULTINEWS AND WIKISUM TEST SETS

Methods	MultiNews			WikiSum		
	R-1	R-2	R-L	R-1	R-2	R-L
supervised extractive methods						
Oracle	55.40	29.91	50.51	49.43	27.18	45.04
FT (2019)	**44.32**	**15.11**	**20.50**	40.56	25.35	34.73
HT (2019)	42.36	15.27	22.08	**41.53**	**26.52**	**35.76**
T-DMCA (2018)	-	-	-	40.77	25.60	34.90
HiMAP (2019)	44.17	16.05	21.38	-	-	-
unsupervised extractive methods						
Lead	39.41	11.77	14.51	37.63	14.75	34.76
TextRank	38.44	13.10	13.50	23.66	7.79	21.23
LexRank	38.27	12.70	13.20	36.12	11.67	22.52
MMR	38.77	11.98	12.91	31.22	10.24	22.48
BERT+MMR	40.11	11.63	35.28	34.70	12.64	28.74
PacSum (2019)	43.27	14.16	38.25	36.85	12.94	33.64
FAR	43.48	16.87	44.00	38.11	14.54	35.01
RFAR	**44.26**	**17.84**	**44.91**	**40.02**	**17.15**	**35.88**

extractive model on CNN/DM and NYT, which design two auxiliary tasks to help BERT measure the salience of sentences.

From the results, we can see that our FAR outperforms all baselines in the second block and PacSum in terms of ROUGE-1/2/L on 3 SDS datasets. Especially on NYT, our FAR outperforms the previous best unsupervised extractive system STAS and supervised method REFERSH. However, the modeling of redundancy hurt the performance on SDS datasets due to these datasets only extracting little sentences from documents as the summary. The improvement of PacSum, STAS, and our FAR on WikiHow dataset is limited. We analyze the dataset and find that the WikiHow dataset is more adaptive for abstractive models [35].

2) Results on LDS: Table IV reports the results on long document summarization (LDS) datasets arXiv, PubMed and BillSum. We compare with the best unsupervised model **HipoRank** [12] on arXiv and PubMed datasets, which is specially designed for the scientific long document. For supervised extractive models, we compare with **SummaRuNNer** [4] and **GlobalLocalCont** [36]. We also compare with supervised abstractive models **Discourse-aware** [26] and **PRT-GEN**.

As shown in Table IV, our models have obviously higher ROUGE-1/2/L score (+1.89 +1.56 +1.38) on arXiv and (+2.22 +1.55 +1.45) on PubMed than PacSum. HipoRank achieved a remarkable result on PubMed due to the boundary function which is very suitable for long scientific documents. Compared with HipoRank, our FAR performs better on arXiv and is more adaptive and generalized for different datasets. However, unsupervised models still have a gap with supervised extractive models on LDS datasets. The reason for this gap is that

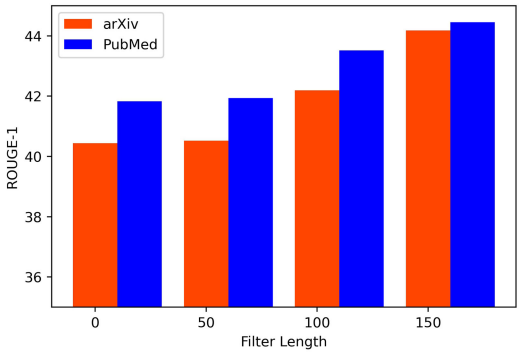


Fig. 4. Performance on arXiv and PubMed, when we filter examples in the test set with summary length.

supervised extractive models can extract sentences with dynamic length through training with labeled corpus, but unsupervised models need to pre-define the length of the extracted summary.

To prove the summary length hurts the performance of unsupervised models, we filter examples on the test set of arXiv and PubMed with summary length and report the results in Fig. 4. We filter examples which summary length is less than 50, 100, and 150. We can see that the performance is improved obviously when we remove short summary examples. When the average length of the summary is closer to the predefined length of the extracted summary, the performance of unsupervised methods is competitive with strong supervised models. So the research on dynamic summary length selection is crucial and meaningful for future work. Besides, we can see that the improvement on BillSum is tiny, due to the document in BillSum contains many short and uncommon sentences.

3) Results on MDS: Table V reports the results on multi-documents summarization datasets MultiNews and WikiSum. **T-DMCA** and **HiMAP** are proposed with the construction of WikiSum and MultiNews. **FT** (Flat Transformer) and **HT** (Hierarchical Transformer) are two supervised extractive models which are proposed by [6].

From results in Table V, we can see that PacSum, FAR, and RFAR have a strong performance on MultiNews, which may result from the characteristic of news datasets and the high-quality human-written documents-summary pairs of MultiNews. Our FAR and RFAR are better than PacSum on WikiSum, especially RFAR. We also can observe that the performance of PacSum and RFAR are far less than supervised models. Because 1) the length of a multi-document summary has a fluctuation and unsupervised methods are hard to decide the length of extracted sentences; 2) multi-document inputs always have many redundant sentences which describe the same content. PacSum and FAR did not especially consider the redundant problem. The advantage of RFAR was distinct on multi-document summarization datasets.

D. Results of BertScore

We also report BertScore [32] of our models on eight datasets in Table VI. Overall, the performance of our methods with BertScore is similar to with ROUGE scores. We can see that

TABLE VI
RESULTS OF BERTSCORE ON EIGHT DATASETS

	PacSum	FAR	RFAR	Oracle
cnndm	**86.74**	86.25	85.76	90.07
nyt	**86.75**	85.74	84.28	90.69
wikihow	84.81	**84.92**	84.86	88.96
arxiv	81.78	82.32	**83.63**	86.91
pubmed	83.39	83.48	**83.75**	87.05
billsum	75.85	76.47	**77.11**	81.26
multinews	84.01	84.33	**84.79**	88.64
wikisum	82.52	82.67	**83.28**	87.45

TABLE VII
ABLATION STUDY ON ARXIV AND NYT

arXiv	R-1	R-2	R-L
FAR	**40.92**	**13.75**	**35.56**
-facet-aware scoring	39.61	12.45	34.37
-modified DC	38.32	11.53	33.35
-post-training	40.02	12.79	34.67
NYT	R-1	R-2	R-L
FAR	**41.67**	**21.93**	**37.68**
-facet-aware scoring	40.82	21.10	36.81
-modified DC	39.90	20.47	36.02
-post-training	40.93	21.38	36.99

our FAR and RFAR surpass PacSum on most datasets, except CNN/DM and NYT. FAR mainly promoted on the long-document, and RFAR further promoted on multi-documents. That is consistent with our previous conclusions and the purpose of our systems.

VI. DISCUSSION

In this section, we present a series of analyses and tests to understand the improvements of our FAR and RFAR. For some complex experiments, we choose NYT from SDS and arXiv from LDS to analyze the performance of FAR. These two datasets are typical and cover the situation of short and long document inputs.

A. Ablation Study

In order to access the contribution of each component in our FAR – modified DC, facet-aware scoring, and post-training encoder. We remove each of them and report the results in Table VII. We can see that modified DC and facet-aware scoring are indispensable to the performance of FAR. The performance of FAR drops sharply when removing them. The results also confirm that post-training is usable. The impact of the DPP for RFAR is shown on the main results from Table III–V.

B. Human Evaluation

To evaluate the ability of FAR and RFAR in reducing facet bias and redundancy, we asked 3 human annotators to evaluate the extracted summaries of PacSum and FAR with the gold reference summary. Three annotators were asked to give 0-2 scores for facet bias and redundancy of 100 random sampled examples from test sets of NYT and 100 random sampled examples from test sets of arXiv. The results of PacSum in terms of facet bias is 1.44 and redundancy is 1.14. Our FAR performs significantly better than PacSum whose facet bias is **0.98** and redundancy is

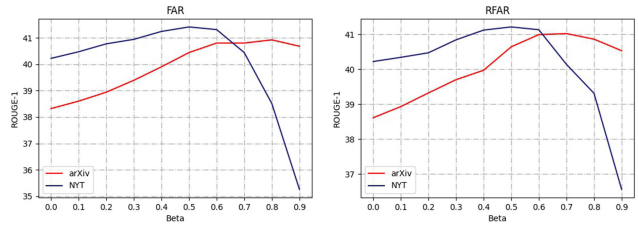


Fig. 5. The performance of FAR and RFAR against different values of β on arXiv and NYT datasets.

TABLE VIII
IMPACT OF DIFFERENT SIMILARITY MEASURE METHODS

arXiv	R-1	R-2	R-L
Dot-Product Similarity	**40.92**	**13.75**	**35.56**
Cosine Similarity	37.94	11.38	33.07
Euclidean Distance	39.12	12.06	34.64
NYT	R-1	R-2	R-L
Dot-Product Similarity	**41.67**	**21.93**	**37.68**
Cosine Similarity	38.93	18.70	35.94
Euclidean Distance	39.97	20.39	36.16

0.83. The facet bias of RFAR is **0.95** and redundancy is **0.71**. Human evaluation results indicated that FAR and RFAR can extract high-quality summaries by facet-aware modeling and reducing redundancy.

C. Impact of Hyper-Parameters

We mainly discuss the hyper-parameter β, which is used to filter out noise sentences in documents. We fixed other hyper-parameters and observed the change of ROUGE-1 from 0.1 to 0.9 with β in Fig. 5. The figure show that Hyper-parameter β has great impact on the performance FAR and RFAR, especially on NYT dataset. This proved that noise sentences truly exists and hurt the performance of centrality-based models.

We can see the best setting of other hyper-parameters in Table II. They also influence models' performance. However, in the experiment, we find that they do not need many adjustments. We designed them for more generic scenarios.

D. Impact of Different Similarity Measure Methods

We empirically employ the dot-product to measure the similarity between sentences. In Table VIII, we show the comparison of different similarity measure methods: dot-product, cosine similarity, and euclidean distance. We can see that cosine similarity leads to a sharp drop on both datasets and dot-product similarity is the best choice for sentence similarity measure.

E. Impact of Different Document Representations

We empirically employ the max-pooling function to extract document representation from the output of the BERT encoder. In Table IX, we show the comparison of different document representation methods: max-pooling, mean-pooling, the "[CLS]". We can see that the max-pooling function is the best choice for document representation in our model and the "[CLS]" has the worst performance.

TABLE IX
IMPACT OF DIFFERENT DOCUMENT REPRESENTATIONS

arXiv	R-1	R-2	R-L
max-pooling	**40.92**	**13.75**	**35.56**
mean-pooling	40.02	12.86	34.95
"[CLS]"	39.64	12.27	34.07
NYT	R-1	R-2	R-L
max-pooling	**41.67**	**21.93**	**37.68**
mean-pooling	40.12	20.32	36.54
"[CLS]"	39.96	20.01	36.18

TABLE X
RATIO OF DIVERSITY SCORE GREATER THAN THE DIVERSITY THRESHOLD

	PacSum	FAR	RFAR
NYT	67.88	56.23	**39.93**
CNNDM	77.53	77.03	**65.41**
WikiHow	54.35	53.51	**37.82**
arXiv	34.48	23.76	**18.97**
PubMed	32.21	29.85	**23.92**
BillSum	13.38	12.98	**11.25**
MultiNews	78.56	53.81	**28.32**
WikiSum	49.61	40.33	**30.94**

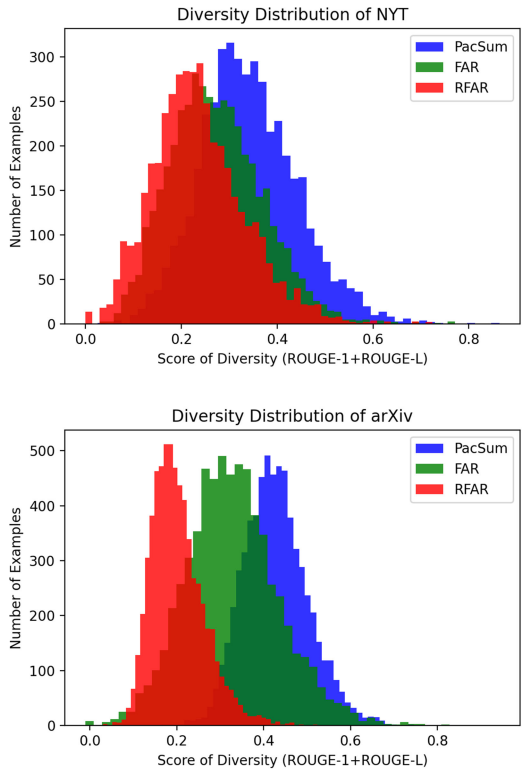


Fig. 6. Distribution of diversity score on NYT and arXiv datasets.

F. Analysis of Diversity

We proposed a simple method, which computes the average of ROUGE scores (ROUGE-1 and ROUGE-L) between summary sentences, to measure the diversity of the extracted summary.[4] We call it diversity score. The summary has a smaller diversity score means more diversity. If the diversity score is greater than the threshold σ, we consider this summary is focused on one facet. We employ the upper quartile value of the gold reference

[4]We also attempt to employ BertScore to measure diversity. However, the value is inconsistent on different datasets.

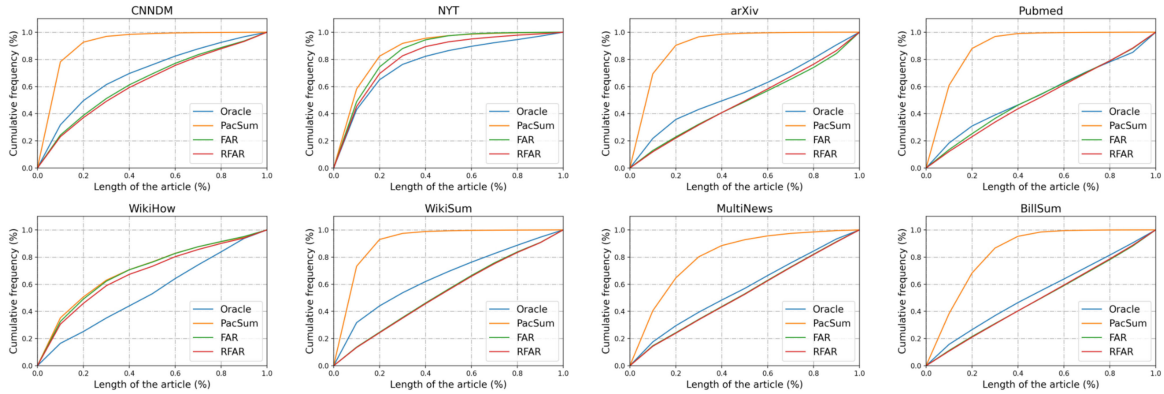


Fig. 7. The cumulative distribution of selected sentences over the length of the article. The X-axis is the ratio of article length. Y-axis is the cumulative percentage of summary sentences.

diversity score as the threshold σ. The ratio of diversity score greater than the threshold can show the facet bias problem in the extracted summary to some extent. We report the ratio in Table X. The results show that 1) the facet bias problem is serious on the News Articles (CNN/DM, NYT, and MultiNews), 2) the low ratio on long document demonstrate that they contain more than one facets. Specially, some ratio value is less than 25, which means extracted summary sentences are dissimilar to each other than gold reference. Because gold reference is written by human, which have better context correlation.

We also plot the distribution of diversity score on NYT and arXiv in Fig. 6. We can see that our FAR and RFAR are more diverse, which means our models cover more facets of the input document.

G. Analysis of Position Distribution

We analyze the position distribution of extracted sentences in the input document and plot the cumulative distribution over the length of the article [37], [38] in the Fig. 7. Overall, the position of the Oracle system is well-distributed. Our two methods are closer to the Oracle distribution curve. The position distribution shows that PacSum tends to select lead sentences, which is called lead bias problem [39], [40]. We can see that our method can select sentences evenly in all positions based on ensuring the quality of abstracts.

For news article datasets, we can see that the oracle curve and PacSum curve both rapid upstrokes before 20% length of the input document, which demonstrates that news articles have the position bias problem. Due to the input of MultiNews dataset is the concatenation of multi-documents, the distribution is dissimilar with CNN/DM and NYT.

H. Analysis of Inference Time

We sample 1,000 examples from NYT and arXiv to test the inference time of three systems on 16xCPUs. All three systems need sentence representations from the BERT model. So we ignore this time and only focus on the algorithm after the obtain of sentence representations. The results are shown in Fig. 8. The inference time of PacSum only increases with the number of the input sentences n. Our methods have another computation

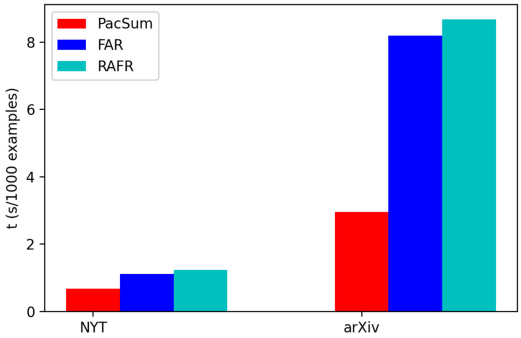


Fig. 8. Inference time of 1,000 examples from NYT and arXiv test set.

related to the number of selected sentences k and the whole time is the plus of them. We can see that our methods are not much slower than PacSum on short input and output dataset NYT. With the increase of input sentences and selected sentences, our methods need more time. However, our methods achieve more robust and effective performance. The increased time is acceptable.

VII. Related Work

A. Automatic Document Summarization

Automatic document summarization is a crucial sub-task of natural language processing. With the development of the neural networks and availability of large-scale parallel datasets, supervised document summarization methods develop sharply [41]–[50]. However, supervised methods need high-quality parallel datasets and are not adaptive to different domains, languages, and lengths of documents. Unsupervised summarization models can tackle these problems via getting the summary based on structural information and features from the input document itself. Summarization models also can be divided into extractive, which selects some sentences from input documents as the summary, and abstractive, which generate summary word by word like a human. Unsupervised abstractive summarization is more challenging than extractive. There are also many interesting works [51]–[56] on unsupervised abstractive summarization. However, the performance of unsupervised abstractive models

is not yet compatible with unsupervised extractive models [13]–[15], [33], [57]–[60]. In this paper, we mainly focus on unsupervised extractive summarization.

B. Unsupervised Extractive Summarization

Graph-based models are effective and widely concerned with unsupervised extractive summarization. Graph-based models represent the input document as a graph, where each sentence in the document is a graph node with a weighted edge which is computed by nodes' similarity. Then, graph-based models select salient sentences from the document through computing centrality of nodes with a degree or PageRank [16]. Typical graph-based models [13]–[15] are based on the assumption that sentence with different order contribute equally to each other (graph is undirected). Different from undirected graph rank models, Zheng *et al.* [10] proposed the directed centrality method, which is based on the Rhetorical Structure Theory (RST) [18] assumption. Dong *et al.* [12] points out that PacSum has position bias, which makes it not suitable for long document summarization, and proposed hierarchical position-based model HipoRank for scientific document summarization. STAS [11] design two summarization tasks related to pretraining tasks to improve sentence representation. Then they proposed a rank method that combines attention weight with reconstruction loss to measure the centrality of sentences.

Most existing works do not consider the diversity problem of extracted summary which is proved vital for quality of summary by [37], [38]. In this paper, we reinforce the diversity of summary through tackle the redundant and facet bias problem.

VIII. Conclusion

In order to tackle the facet bias and redundancy problem in unsupervised extractive summarization, in this paper, we proposed a novel redundancy- and facet-aware centrality-based ranking model RFAR, which is based on our previous proposed FAR. To balance the importance and redundancy of the summary, we introduce the DPP into FAR. Experimental results on a wide range of summarization tasks show that our methods consistently outperform strong baselines, especially in long- and multi-document scenarios. Extensive analyses confirmed that our model also could tackle position bias to some extent.

Acknowledgment

The authors would like to thank all editors and reviewers for their careful reading of our paper and their many insightful comments and suggestions.

References

[1] A. Nenkova and K. McKeown, *Automatic Summarization*. Boston, MA, USA: Now Publishers Inc., 2011.

[2] W. Kryscinski, B. McCann, C. Xiong, and R. Socher, "Evaluating the factual consistency of abstractive text summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 9332–9346.

[3] J. Cheng and M. Lapata, "Neural summarization by extracting sentences and words," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany, 2016, pp. 484–494.

[4] R. Nallapati, F. Zhai, and B. Zhou, "SummaRuNNer: A recurrent neural network based sequence model for extractive summarization of documents," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 3075–3081.

[5] S. Gehrmann, Y. Deng, and A. Rush, "Bottom-up abstractive summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Brussels, Belgium, 2018, pp. 4098–4109.

[6] Y. Liu and M. Lapata, "Hierarchical transformers for multi-document summarization," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 5070–5081.

[7] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," in *Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, Hong Kong, China, 2019, pp. 3730–3740.

[8] X. Zhang, F. Wei, and M. Zhou, "HIBERT: Document level pre-training of hierarchical bidirectional transformers for document summarization," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 5059–5069.

[9] D. Wang, P. Liu, Y. Zheng, X. Qiu, and X. Huang, "Heterogeneous graph neural networks for extractive document summarization," in *Proc. 58th Annu*. Meeting Assoc. Comput. Linguistics, 2020, pp. 6209–6219.

[10] H. Zheng and M. Lapata, "Sentence centrality revisited for unsupervised summarization," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 6236–6247.

[11] S. Xu, X. Zhang, Y. Wu, F. Wei, and M. Zhou, "Unsupervised extractive summarization by pre-training hierarchical transformers," in *Proc. Findings Assoc. Comput. Linguistics, EMNLP*, 2020, pp. 1784–1795.

[12] Y. Dong, A. Romascanu, and J. C. Cheung, "Discourse-Aware Unsupervised Summarization for Long Scientific Documents," in *Proc. 16th Conf. Eur. Chapter Assoc. Comput. Linguistics*, 2021, pp. 1089–1102.

[13] D. R. Radev, H. Jing, and M. Budzikowska, "Centroid-based summarization of multiple documents: Sentence extraction, utility-based evaluation, and user studies," in *Proc. NAACL-ANLPWorkshop Autom. Summarization*, 2000, pp. 21–30.

[14] R. Mihalcea and P. Tarau, "TextRank: Bringing order into text," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Barcelona, Spain, 2004, pp. 404–411.

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

[16] S. Brin and L. Page, "The anatomy of a large-scale hypertextual web search engine," *Comput. Netw.*, vol. 30, pp. 107–117, 1998.

[17] Y. Mao, L. Liu, Q. Zhu, X. Ren, and J. Han, "Facet-aware evaluation for extractive summarization," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 4941–4957.

[18] W. C. Mann and S. A. Thompson, "Rhetorical structure theory: Toward a functional theory of text organization," *Text*, vol. 8, no. 3, pp. 243–281, 1988.

[19] A. Kulesza and B. Taskar, "Learning determinantal point processes," in *Proc. 27th Conf. Uncertainty Artif. Intell.*, Arlington, VA, USA, 2011, pp. 419–427.

[20] A. Kulesza, "Determinantal point processes for machine learning," Found. Trends *Mach. Lear.*, vol. 5, no. 2/3, pp. 123–286, 2012.

[21] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, 2019, pp. 3980–3990.

[22] K. M. Hermann *et al.*, "Teaching machines to read and comprehend," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 1693–1701.

[23] J. J. Li, K. Thadani, and A. Stent, "The role of discourse units in near-extractive summarization," in *Proc. 17th Annu. Meeting Special Interest Group Discourse Dialogue*, 2016, pp. 137–147.

[24] G. Durrett, T. Berg-Kirkpatrick, and D. Klein, "Learning-based single-document summarization with compression and anaphoricity constraints," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, Berlin, Germany, 2016, pp. 1998–2008.

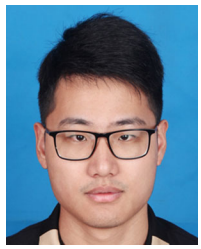
[25] A. Fabbri, I. Li, T. She, S. Li, and D. Radev, "Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 1074–1084.

[26] A. Cohan *et al.*, "A discourse-aware attention model for abstractive summarization of long documents," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, New Orleans, Louisiana, 2018, pp. 615–621.

[27] P. J. Liu *et al.*, "Generating wikipedia by summarizing long sequences," in *Proc. Int. Conf. Learn. Representations*, 2018. [Online]. Available: https://openreview.net/forum?id=Hyg0vbWC-

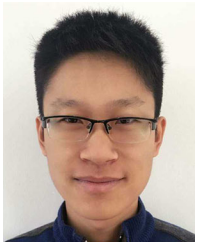
[28] M. Koupaee and W. Y. Wang, "*WikiHow: A large scale text summarization dataset*," 2018, *arXiv:1810.09305*.

[29] A. Kornilova and V. Eidelman, "BillSum: A corpus for automatic summarization of US legislation," in *Proc. 2nd Workshop New Frontiers Summarization*, Hong Kong, China, 2019, pp. 48–56.
[30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *in Proc. Int. Conf. Learn. Representations*, 2014. [Online]. Available: https://arxiv.org/abs/1412.6980
[31] C.-Y. Lin and E. Hovy, "Automatic evaluation of summaries using N-gram co-occurrence statistics," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics Hum. Lang. Technol.*, 2003, pp. 150–157.
[32] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, "BERTScore: Evaluating text generation with bert," in *Proc. Int. Conf. Learn. Representations*, 2020. [Online]. Available: https://openreview.net/forum?id=SkeHuCVFDr
[33] J. Carbonell and J. Goldstein, "The use of MMR, diversity-based reranking for reordering documents and producing summaries," in *Proc. 21st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retriev*, New York, NY, USA, Association for Computing Machinery, 1998, pp. 335–336.
[34] S. Narayan, S. B. Cohen, and M. Lapata, "Ranking sentences for extractive summarization with reinforcement learning," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, New Orleans, LA, USA, 2018, pp. 1747–1759.
[35] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, "PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, 11 328–11 339.
[36] W. Xiao and G. Carenini, "Extractive summarization of long documents by combining global and local context," in *Proc. Conf. Empir. Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, Hong Kong, China, 2019, pp. 3011–3021.
[37] W. Kryscinski, N. S. Keskar, B. McCann, C. Xiong, and R. Socher, "Neural text summarization: A critical evaluation," in *Proc. Conf. Empir. Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, Hong Kong, China, 2019, pp. 540–551. [Online]. Available: https://aclanthology.org/D19-1051
[38] Z. Liu, K. Shi, and N. Chen, "Conditional neural generation using sub-aspect functions for extractive news summarization," in *Proc. Findings Assoc. Comput. Linguistics: EMNLP*, 2020, pp. 1453–1463.
[39] M. Grenander, Y. Dong, J. C. K. Cheung, and A. Louis, "Countering the effects of lead bias in news summarization via multi-stage training and auxiliary losses," in *Proc. Conf. Empir. Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, Hong Kong, China, 2019, pp. 6019–6024.
[40] L. Xing, W. Xiao, and G. Carenini, "Demoting the lead bias in news summarization via alternating adversarial learning," in *Proc. 59th Annu. Meeting Assoc. Comput. Linguistics 11th Int. Joint Conf. Natural Lang. Process.*, 2021, pp. 948–954. [Online]. Available: https://aclanthology.org/2021.acl-short.119
[41] X. Zhang, M. Lapata, F. Wei, and M. Zhou, "Neural latent extractive document summarization," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, Brussels, Belgium, 2018, pp. 779–784.
[42] J. Zhang, Y. Zhou, and C. Zong, "Abstractive cross-language summarization via translation model enhanced predicate argument structure fusing," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 24, no. 10, pp. 1842–1853, Oct. 2016.
[43] K.-Y. Chen, S.-H. Liu, B. Chen, and H.-M. Wang, "An information distillation framework for extractive summarization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 1, pp. 161–170, Jan. 2018.
[44] J. Zhang, Y. Zhao, H. Li, and C. Zong, "Attention with sparsity regularization for neural machine translation and summarization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 3, pp. 507–518, Mar. 2019.
[45] M. Zhong, P. Liu, D. Wang, X. Qiu, and X. Huang, "Searching for effective neural extractive summarization: What works and what's next," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 1049–1058.
[46] S. Cho, L. Lebanoff, H. Foroosh, and F. Liu, "Improving the similarity measure of determinantal point processes for extractive multi-document summarization," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 1027–1038.
[47] Q. Zhou, N. Yang, F. Wei, S. Huang, M. Zhou, and T. Zhao, "A joint sentence scoring and selection framework for neural extractive document summarization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 671–681, 2020.
[48] Y. Cao, H. Liu, and X. Wan, "Jointly learning to align and summarize for neural cross-lingual summarization," in *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.*, 2020, pp. 6220–6231.
[49] H. Jin, T. Wang, and X. Wan, "Multi-granularity interaction network for extractive and abstractive multi-document summarization," in *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.*, 2020, pp. 6244–6254.
[50] M. Zhong, P. Liu, Y. Chen, D. Wang, X. Qiu, and X. Huang, "Extractive summarization as text matching," in *Proc. 58th Annu. Meet. Assoc. Comput. Linguist.*, 2020, pp. 6197–6208.
[51] C. Baziotis, I. Androutsopoulos, I. Konstas, and A. Potamianos, "SEQˆ3: Differentiable sequence-to-sequence-to-sequence autoencoder for unsupervised abstractive sentence compression," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, Minneapolis, MN, USA, 2019, pp. 673–681.
[52] Y. Jernite, "Unsupervised text summarization via mixed model back-translation," 2019, *arXiv:1908.08566*.
[53] J. Zhou and A. Rush, "Simple unsupervised summarization by contextual matching," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, Florence, Italy, 2019, pp. 5101–5106.
[54] P. West, A. Holtzman, J. Buys, and Y. Choi, "BottleSum: Unsupervised and self-supervised sentence summarization using the information bottleneck principle," in *Proc. Conf. Empir. Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, Hong Kong, China, 2019, pp. 3752–3761.
[55] E. Chu and P. Liu, "MeanSum: A neural model for unsupervised multi-document abstractive summarization," in *Proc. Int. Conf. Learn. Representations*, Long Beach, CA, USA, 2019, pp. 1223–1232.
[56] Z. Yang, C. Zhu, R. Gmyr, M. Zeng, X. Huang, and E. Darve, "TED: A pretrained unsupervised summarization model with theme modeling and denoising," in *Proc. Findings Assoc. Comput. Linguistics, EMNLP*, 2020, pp. 1865–1874.
[57] S. Yan and X. Wan, "SRRank: Leveraging semantic roles for extractive multi-document summarization," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 22, no. 12, pp. 2048–2058, Dec. 2014.
[58] X. Wan, "An exploration of document impact on graph-based multi-document summarization," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, Honolulu, Hawaii, 2008, pp. 755–762.
[59] N. Schluter and A. Søgaard, "Unsupervised extractive summarization via coverage maximization with syntactic and semantic concepts," in *Proc. 53th Annu. Meet. Assoc. Comput. Linguist. 7th Int. Joint Conf. Natural Lang.*, Beijing, China, 2015, pp. 840–844.
[60] J. Zhao *et al.*, "SummPip: Unsupervised multi-document summarization with sentence graph compression," in *Proc. 20st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retriev.*, New York, NY, USA, 2020, pp. 1949–1952.



Xinnian Liang received the bachelor's degree in July 2019 from the School of Computer Science and Software, Nanjing University of Information Science & Technology, Nanjing, China, where he is currently working toward the Ph.D. degree. His current research interests include text summarization, information extraction, and natural language generation.

Jing Li received the bachelor's degree in July 2019 from the School of Computer Science and Software, Nanjing University of Information Science & Technology, Nanjing, China, where she is currently working toward the master's degree. Her current research interests include natural language processing and graph data mining.



Shuangzhi Wu received the Ph.D. degree in July 2019 from the Department of Computer Science, Harbin Institute of Technology, Harbin, China. Then, he joined Tencent as an NLP Researcher. His current research interests include machine translation, generation and large-scale pre-training models.

Mu Li is currently an NLP Researcher with Tencent. He is also working on fundamental NLP problems, models, algorithms and innovations. Before joining Tencent, he was a Principal Researcher with Microsoft Research Asia. His research interests include language modeling, syntactic parsing, machine translation, and deep learning methods for other NLP tasks.

Zhoujun Li received the M.Sc. and Ph.D. degrees in computer science from the National University of Defence Technology, Changsha, China, in 1984 and 1999, respectively. He is currently with the School of Computer, Beihang University, Beijing, China, where he has been a Professor since 2001. He has authored or coauthored more than 150 papers in international journals, such as the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, and *Information Sciences*, and international conferences, such as SIGKDD, ACL, SIGIR, AAAI, IJCAI, SDM, CIKM, and WSDM. His current research interests include data mining, information retrieval, and database. Dr. Li is a PC Member of several international conferences, such as SDM 2015, CIKM 2013, and PRICAI 2012.