IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 7 March 2024, accepted 5 April 2024, date of publication 15 April 2024, date of current version 22 April 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3388493

RESEARCH ARTICLE

Mining Both Commonality and Specificity From Multiple Documents for Multi-Document Summarization

BING MA
Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China
University of Chinese Academy of Sciences, Beijing 101408, China

e-mail: mabing17z@ict.ac.cn

ABSTRACT The multi-document summarization task requires the designed summarizer to generate a short text that covers the important information of original multiple documents and satisfies content diversity. To fulfill the dual requirements of coverage and diversity in multi-document summarization, this study introduces a novel method. Initially, a class tree is constructed through hierarchical clustering of documents. Subsequently, a sentence selection method based on class tree is proposed for generating a summary. Specifically, a top-down traversal is performed on the class tree, during which sentences are selected from each node based on their similarity to the centroid of the documents within the node and their dissimilarity to the centroid of documents not belonging to the node. Sentences selected from the root node reflect the commonality of all document, and sentences selected from the sub nodes reflect the distinct specificity of the respective subclasses. Experimental results on standard text summarization datasets DUC'2002, DUC'2003, and DUC'2004 demonstrate that the proposed method significantly outperforms the variant method that considers only commonality of all documents, achieving average improvements of up to 1.54 and 1.42 in ROUGE-1 and ROUGE-L scores, respectively. Additionally, the method demonstrates significant superiority over another variant method that considers only the specificity of subclasses, achieving average improvements of up to 2.16 and 2.01 in ROUGE-1 and ROUGE-L scores, respectively. Furthermore, extensive experiments on DUC'2004 and Multi-News datasets show that the proposed method outperforms lots of competitive supervised and unsupervised multi-document summarization methods and yields considerable results.

INDEX TERMS Class tree, commonality and specificity, hierarchical clustering of documents, multi-document summarization, pre-trained embedding representation.

I. INTRODUCTION

Automatic text summarization is becoming much more important because of the exponential growth of digital textual information on the web. Multi-document summarization, which aims to generate a short text containing important and diverse information of original multiple documents, is a challenging focus of NLP research. A well-organized summary of multiple documents needs to cover the main information of all documents comprehensively and simultaneously satisfy content diversity. Extractive summarization methods, which generate a summary by selecting a few important sentences from original documents, attract much attention because of its simplicity and robustness. This paper focuses on extractive multi-document summarization.

Most extractive multi-document summarization methods splice all sentences contained in original multiple documents into a larger text, and then generate a summary by selecting sentences from the larger text [1], [2], [3]. However, the task of summarizing multiple documents is more difficult than the task of summarizing a single document. Simply transforming multi-document summarization task into summarizing a single larger text completely breaks the constraints of documents on their sentences and lacks comparisons

The associate editor coordinating the review of this manuscript and approving it for publication was Antonio J. R. Neves.

between documents, which results in the inability to mine the relevant information between documents, including mining the common information (i.e., commonality) of all documents and the important specific information (i.e., specificity) of some subclasses of documents.

The centroid-based summarization methods focus on the commonality of all documents or all sentences and they select sentences based on the centroid words of all documents [4], [5] or the centroid embedding of all sentences [1]. The clustering-based summarization methods divide sentences into multiple groups and select sentences from each group [2], [6]. These methods do not take into account the commonality and specificity of documents simultaneously.

Think about the process of humans summarizing multiple documents: we would first describe the common information of all documents, and then describe the important specific information of some subclasses of these documents respectively, so as to satisfy the coverage and diversity requirements of multi-document summarization.

Inspired by the idea of humans summarizing multiple documents, this paper proposes a novel multi-document summarization method based on the class tree constructed by hierarchical clustering of documents. Firstly, a class tree is constructed by hierarchically clustering the documents in a top-down manner. Each node on the class tree consists of a group of documents, where the root node contains all documents, and each sub node contains a subclass of all documents. Next, a sentence selection method based on class tree is proposed. Specifically, a top-down traversal of nodes is performed on the class tree, with the aim of selecting sentences from each node, until the cumulative length of the selected sentences reaches a pre-specified value. Within each node, sentences are selected based on their similarity to the centroid of the documents within the node, as well as their dissimilarity to the centroid of the documents not belonging to the node. The sentences selected from the root node reflect the commonality of all documents, and the sentences selected from sub nodes at different hierarchical levels highlight the distinct specificity of the respective subclasses. Finally, all selected sentences are arranged according to the order of their corresponding nodes on the class tree to form a summary.

To summarize, the main contributions of this study are as follows:

1) This study applies hierarchical clustering of documents to multi-document summarization, aiming to organize input documents into a class tree, where the root node contains all input documents and the sub nodes at different hierarchical levels contain different subclasses of all documents. This approach endows the proposed model with the capability to extract not only the sentences that summarize the overall content of all input documents but also the sentences that highlight the distinct characteristics of different subclasses. Because different nodes at different hierarchical levels enable the capture of distinct characteristics of the input documents.
2) This study introduces a novel sentence scoring method, which assigns a score to each sentence within a node based on its similarity to the centroid of the documents within the node and its dissimilarity to the centroid of documents not belonging to the node. The scoring mechanism enables sentences within the root node that emphasize the commonalities across all documents to obtain higher scores, and it enables sentences within each sub node that emphasize the unique characteristics of the documents within the sub node to obtain higher scores. The innovative combination of the class tree structure and this sentence scoring method enhance the coverage and diversity of the generated summary.
3) This paper assesses the effectiveness of the proposed method for multi-document summarization task by comparing it with distinct variant methods on the standard multi-document summarization datasets, including DUC'2002, DUC'2003, and DUC'2004, and provide an empirical analysis on the results. Experimental results show that the method, which considers both commonality of all documents and distinct specificity of different subclasses of documents, significantly outperforms the variant methods considering only commonality of all documents (achieving average improvements of up to 1.54 in ROUGE-1 score) or only specificity of different subclasses (achieving average improvements of up to 2.16 in ROUGE-1 score), and outperforms the variant method which is based on sentences hierarchical clustering (achieving average improvements of up to 1.25 in ROUGE-1 score).
4) This paper compares the proposed method with several state-of-the-art methods, including both supervised and unsupervised multi-document summarization methods, using the DUC'2004 and Multi-News datasets as benchmarks [7]. Experimental results show that this method outperforms lots of competitive supervised and unsupervised methods and yields considerable results.

In general, the proposed method is unsupervised and easy to implement, and can be used as a strong baseline for evaluating multi-document summarization systems.

The rest of this paper is organized as follows. Section II discusses the related work. Section III presents the proposed method in detail. Section IV describes the conducted experiments, presents the obtained results, and provide a thorough discussion of the obtained results. Finally, Section V concludes this paper and outlines future work.

II. RELATED WORK

Extractive multi-document summarization method involves selecting some critical sentences from original documents to generate a concise summary, and the length of the summary is determined by the compression rate specified by human users [8]. Various extractive summarization techniques have been proposed, including centroid-based methods, clustering-based methods, statistical based methods, graph-based methods, and so on. Among these techniques, the most pertinent works include centroid-based and clustering-based summarization methods.

The centroid-based methods score each sentence in documents by calculating the similarity between the sentence and the centroid of all documents or all sentences, so as to identify the most central sentences to generate a summary [1], [4], [5].

MEAD [4] scores each sentence based on the centroid words it contains and two other metrics (positional value and first-sentence overlap). The centroid words in MEAD method correspond to the words that are statistically important to multiple documents. The method in [5] improves the original MEAD method by exploiting the word embedding representations to represent the centroid of documents and each sentence, and scoring each sentence based on the cosine similarity between the sentence embedding and the centroid embedding. The method in [1] exploits various sentence embedding models to represent each sentence and the centroid of all sentences (i.e., the mean of all sentence embeddings), and scores each sentence based on the cosine similarity between the sentence embedding and the centroid embedding, as well as two other metrics, sentence novelty and sentence position. These centroid-based methods focus on the commonality property of all documents or all sentences, and take no account of the important specificity property of some subclasses of these documents.

Many clustering-based extractive summarization methods cluster all sentences in documents and then select sentences from each sentence cluster to form a summary [2], [9], [10], [11].

Wang et al. [9] groups sentences into clusters by sentence-level semantic analysis and symmetric non-negative matrix factorization, and selects the most informative sentences from each sentence cluster. Mohd et al. [10] represents each sentence as a big-vector using the Word2Vec model and applies the k-means algorithm to cluster sentences, and then scores sentences in each sentence cluster based on various statistical features (e.g. sentence length, position, etc.). Rouane et al. [11] also uses the k-means algorithm to cluster sentences, and then scores each sentence in each cluster based on the frequent itemsets of the cluster contained by the sentence. Yang et al. [2] proposes a ranking-based sentence clustering framework to generate sentence clusters, and uses a modified MMR-like approach to select the highest scored sentences from the sentence clusters arranged in descending order to form the summary. These clustering-based methods take no account of the commonality property of all documents, which is important for multi-document summarization because the input of multi-document summarization tasks is usually a set of related documents.

Additionally, statistical-based methods [12] and graph-based methods [3], [13] are two widely used extractive text summarization methods. The statistical-based methods calculate the score of each sentence by leveraging statistical features of texts, such as Term Frequency, Inverse Document Frequency, and sentence position information. Subsequently, the sentences with higher scores are selected to compose the summary. The primary distinction among various statistical extractive summarization methods lies in the sentence scoring methodologies employed. The graph-based methods transform the original documents into a graph representation, with sentences serving as nodes and the similarity between sentences serving as the weights of the links connecting the corresponding nodes. These node weights are iteratively updated based on the link weights. Subsequently, sentences with higher scores in the graph are chosen to compose the summary.

III. METHODS

The proposed method takes a set of documents and a pre-given summary length as input, and outputs a multi-document summary. It consists of three steps: (1) pre-processing of documents, (2) hierarchical clustering of documents for constructing a class tree of documents, and (3) sentence selection from the constructed class tree and summary generation. Each of these steps will be presented in detail in the subsequent three subsections.

A. PRE-PROCESSING

Pre-trained models are widely used in Natural Language Processing tasks. There are usually two ways to use the pre-trained models: (1) Feature Extraction based approach, which uses the pre-trained model learned from a large amount of textual data to encode texts of arbitrary length into vectors of fixed length; (2) Fine-Tuning based approach, which trains the downstream tasks by fine-tuning the pre-trained model's parameters. This paper adopts the feature extraction based approach, where the pre-trained model is applied on the input documents to obtain the embedding representations of sentences and documents.

This study uses pre-trained sentence embeddings model to encode sentences. To obtain document embedding vectors, two ways can be used: one is to directly obtain the document embedding vectors by taking each document as the input of the pre-trained embedding model; the other is to obtain the document embedding vectors based on sentence embedding vectors, e.g., a document embedding vector can be represented as the average of the sentence embedding vectors of all sentences it contains.

Many pre-trained embedding models can be used in the proposed method to obtain the sentence embedding vectors and document embedding vectors. This section focuses on the elaboration of the proposed hierarchical clustering-based multi-document summarization method. The selection of pre-trained embedding models and document embedding approaches will be discussed in the Experiment section.

Formally, given a set of documents D containing n documents $D = \{d_1, d_2, \cdots, d_n\}$. Firstly, each document $d_i \in D$ is split into sentences (denoted as $d_i = \{s_1^i, s_2^i, \cdots, s_{|d_i|}^i\}$) using the Natural Language Toolkit (NLTK).[1] Next, each sentence in each document ($s_k^i \in d_i$) is mapped to a fixed-length vector (denoted as $\boldsymbol{s}_k^i$) using the pre-trained embedding model, and each document ($d_i \in D$) is mapped to a vector of the same length (denoted as $\boldsymbol{d}_i$).

B. HIERARCHICAL CLUSTERING OF DOCUMENTS

The proposed top-down hierarchical clustering algorithm for constructing the class tree of documents includes the following steps:

1) **Generate the root node of class tree.**
All documents in D form the root node. The root node constitutes the first layer of class tree. (After step 1, the root node becomes the latest layer of the class tree.)

2) **Generate the next layer of class tree.**
For each node of the latest layer of class tree, the k-means algorithm[2] is used to divide the documents in the node into k sub nodes (also called k subclasses). All new sub nodes generated in this step constitute the new latest layer of class tree.

3) **Repeat step 2 until one of the following conditions is satisfied.**
Condition 1: There are no nodes on the latest layer of class tree that can be divided using the k-means algorithm.
Condition 2: The total number of nodes on the class tree exceeds the number of sentences required for the summary, where the required number is specified or estimated according to the pre-given summary length. Because the proposed method will select sentences from each node on the class tree top-down until the pre-given value is reached.

C. SENTENCES SELECTION AND SUMMARY GENERATION

After the construction of the class tree, the proposed method traverses the nodes on the class tree from top to bottom and selects sentences from each node to generate a summary until the summary length reaches the pre-given length.

This section provides a detailed introduction of the three primary components involved in sentences selection and summary generation: (1) the overall flow of traversing the nodes on class tree for sentences selection, (2) the details of sentences scoring and selection within each node, and (3) the process of sorting the selected sentences to form a summary.

1) OVERALL FLOW OF TRAVERSING CLASS TREE

Fig. 1 displays the overall flow chart of traversing the nodes on the class tree for sentences selection.

The order of traversing the nodes on class tree follows two principles: (1) For different layers of class tree, the method traverses the layers from top to bottom; (2) For the nodes on the same layer, the method traverses the nodes in descending order of the number of documents contained in the nodes. Because under the limitation of the pre-given summary length, the method hopes that the selected sentences can cover as many documents as possible while increasing diversity.

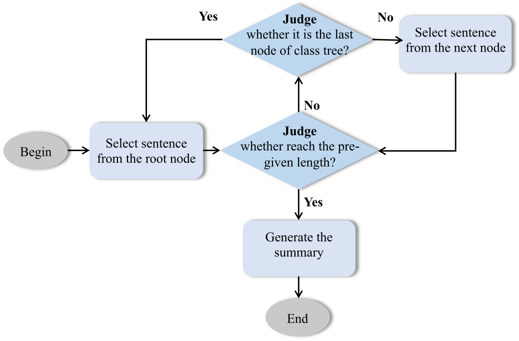

FIGURE 1. The flow chart of selecting sentences from the class tree.

As shown in Fig. 1, if the total length of the selected sentences does not reach the pre-given summary length after selecting sentence from the last node on the last layer of class tree, the method goes back to the first layer of class tree (i.e., the root node) to start the next iteration of selecting sentences, until the total length of the selected sentences in all iterations reaches the pre-given summary length.

2) SENTENCE SCORING AND SELECTION IN EACH NODE

Each node N_t on the class tree consists of multiple documents, denoted as $N_t = \{d_1^t, \cdots, d_{|N_t|}^t\}$. Each document $d_i^t \in N_t$ consists of multiple sentences, denoted as $d_i^t = \{s_1^i, s_2^i, \cdots, s_{|d_i^t|}^i\}$. The proposed method calculates the score of each sentence s_k^i in each node N_t ($s_k^i \in d_i^t$ and $d_i^t \in N_t$).

a: COMMONALITY-SPECIFICITY SCORE

The centroid of all documents in N_t represents the common core of these documents. It is reasonable to think that the sentences that are more similar to the centroid of N_t are more relevant to the documents in N_t, and the sentences that are less similar to the centroid of N_t are less relevant to the documents in N_t. Therefore, the Commonality-Specificity score of each sentence s_k^i in N_t can be calculated as the combination of its similarity to the centroid of N_t and its dissimilarity to the centroid of the documents not in N_t.

The centroid embedding vector of N_t (denoted as $\boldsymbol{C}_{N_t}$) is built by averaging all document embedding vectors in N_t (as shown in Eq. (1)).

$$\boldsymbol{C}_{N_t} = \frac{1}{|N_t|}\sum_{i=1}^{|N_t|}\boldsymbol{d}_i^t \qquad (1)$$

where $|N_t|$ denotes the number of documents in N_t, and $\boldsymbol{d}_i^t$ is the document embedding vector of the i^{th} document in N_t.

Similarly, the centroid embedding vector of the documents not in N_t (denoted as $\boldsymbol{C}_{\overline{N_t}}$, where $\overline{N_t} = \{d \mid d \in D\, and\, d \notin N_t\}$) is built as the average of all document embedding vectors not in N_t.

[1] nltk.tokenize.
[2] sklearn.cluster.KMeans.

The Commonality-Specificity score of each sentence s_k^i in N_t is calculated as follows:

$$\begin{aligned}\text{score}^{\text{CS}}(s_k^i, N_t) &= \delta \cdot \text{Similarity}(\boldsymbol{s_k^i}, \boldsymbol{C_{N_t}}) \\ &\quad + (1-\delta)\cdot(1-\text{Similarity}(\boldsymbol{s_k^i}, \boldsymbol{C_{\overline{N_t}}})) \quad (2)\end{aligned}$$

The value of $\delta \in [0, 1]$. The larger value of δ illustrates more attention to the relevance with the documents in N_t, and the smaller value of δ illustrates more attention to the irrelevance with the documents not in N_t. When $\delta = 1$, the score$^{\text{CS}}$ of each sentence in N_t only focuses on the relevance with the documents in N_t. The method uses the cosine similarity[3] (denoted as Similarity) to calculate the similarity between vectors.

The value of score$^{\text{CS}}$ is bounded in [0, 1]. The sentences with higher score$^{\text{CS}}$ are considered to be more relevant to the documents in N_k and more irrelevant to the documents not in N_k.

b: SENTENCES SCORING AND SELECTION

The Commonality-Specificity score can be used alone to score and select sentences, or combined with other scoring metrics to score and select sentences.

Non-Redundant Score: To reduce the redundancy of the generated summary, the method would assign lower Non-redundant score to the sentences that are more similar to the sentences already selected in previous steps. Specifically, S^p is used to represent the collection of sentences already selected in previous steps, the Non-redundant score of each sentence s_k^i in N_t is calculated as the dissimilarity between s_k^i and its most similar sentence in S^p, which is described as follows:

$$\begin{aligned}&\text{score}^{\text{NR}}(s_k^i, N_t) = \\ &\quad 1-\max(\{\text{Similarity}(\boldsymbol{s_k^i}, \boldsymbol{s_p})\}), s_p \in S^p \quad (3)\end{aligned}$$

The value of score$^{\text{NR}}$ is bounded in [0, 1]. The sentences with higher score$^{\text{NR}}$ are considered to have lower redundancy with the sentences already selected in previous steps. If S^p is Null (i.e., selecting the first sentence from the root node), the score$^{\text{NR}}$ of each sentence in the node is 1.

Position Score: Sentence position is one of the most effective heuristics for selecting sentences to generate summaries, especially for news articles [14], [15]. The sentence position relevance metric (as Eq. (4)) introduced by [16] is adopted to calculate the Position score of each sentence in each document.

$$\text{score}^{\text{P}}(s_k^i) = \max(0.5, \exp(\frac{-\mathcal{P}(s_k^i)}{\sqrt[3]{|d_i|}})) \quad (4)$$

$\mathcal{P}(s_k^i)$ denotes the relative position of the k^{th} sentence s_k^i in the document d_i (starting by 1). The score$^{\text{P}}$ is bounded in [0.5, 1]. The first sentence in each document obtains the highest score$^{\text{P}}$. The score$^{\text{P}}$ of sentences decrease as their distances from the beginning of documents increase, and remain stable at a value of 0.5 after several sentences.

Combination of Three Scores: The final score of each sentence s_k^i in N_t can be defined as a linear combination of the three scores (as Eq. (5)).

$$\begin{aligned}\text{score}^{\text{final}}(s_k^i, N_t) &= \alpha \cdot \text{score}^{\text{CS}}(s_k^i, N_t) \\ &\quad + \beta \cdot \text{score}^{\text{NR}}(s_k^i, N_t) + \gamma \cdot \text{score}^{\text{P}}(s_k^i) \quad (5)\end{aligned}$$

where $\alpha + \beta + \gamma = 1$, and $\alpha, \beta, \gamma \in [0, 1]$. Different values of α, β, and γ indicate different emphases on different scoring metrics. Setting $\alpha = 1$ means that only the Commonality-Specificity score is used to score sentences.

The method selects the sentences that have the highest final score and have not been selected in previous steps from N_t. Only one sentence is selected from each node in each iteration because the method wants to traverse as many nodes as possible under the limitation of the pre-given summary length, so as to increase the diversity of the generated summaries.

3) SUMMARY GENERATION

After the process of selecting sentences from the class tree, the proposed method ranks the selected sentences to form a summary: (1) For the sentences selected from different nodes, the method ranks these sentences according to the traversal order of their corresponding nodes on the class tree (i.e., the two principles introduced above); (2) For multiple sentences selected from the same node (i.e., the first iteration of traversing the class tree did not select enough sentences), the method ranks the sentences according to the order in which they are selected.

The sentences selected from the root node express the commonality of all documents, and the sentences selected from each sub node express the specificity of the corresponding subclass. The above sentences ordering way forms a summary with a total-sub structure.

IV. EXPERIMENT

A. DATASETS AND EVALUATION METRICS

The proposed method is evaluated on the standard multi-document summarization datasets, including DUC'2002-2004 datasets[4] and the Multi-News dataset.[5] The DUC (Document Understanding Conference) datasets, developed by NIST (National Institute of Standards and Technology), are extensively utilized corpora for evaluating text summarization. The DUC'2002 dataset comprises 59 news sets, and the DUC'2003 dataset includes 30 news sets. Each news set is composed of approximately 10 English news articles obtained from TREC. The DUC'2004 Task 2 dataset contains 50 news sets, with each news set consisting of 10 documents sourced from the Associated Press and New

[3] sklearn.cosine_similarity.

[4] https://duc.nist.gov/

[5] https://github.com/Alex-Fabbri/Multi-News

York Times newswires. The Multi-News dataset is a large-scale multi-document summarization news dataset released by Fabbri et al. [7]. Each news set of Multi-News contains a different number of documents (from 1 to 10) on the same topic. Table 1 displays the details of the four datasets.

The Recall-Oriented Understudy for Gisting Evaluation (ROUGE) [17] is adopted to evaluate the performance of the proposed approach. ROUGE is a standard evaluation metric for automatic document summarization. It counts the overlapping units between the generated summaries and reference summaries. Four ROUGE metrics are used in this paper: ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4. ROUGE-N (N=1 or 2) calculates the overlapping N-gram between a generated summary and a set of reference summaries. ROUGE-1 and ROUGE-2 are two most used ROUGE-N measures, and they calculate the number of overlapping unigrams and bigrams respectively. ROUGE-N is calculated as follows:

$$\mathrm{ROUGE-N} = \frac{\sum\limits_{S\in\{\mathrm{ReferenceSummaries}\}}\ \sum\limits_{\mathrm{N-gram}\in S} \mathrm{Count}_{\mathrm{match}}(\mathrm{N-gram})}{\sum\limits_{S\in\{\mathrm{ReferenceSummaries}\}}\ \sum\limits_{\mathrm{N-gram}\in S} \mathrm{Count}(\mathrm{N-gram})} \tag{6}$$

where $\mathrm{Count}_{\mathrm{match}}(\mathrm{N-gram})$ is the maximum number of N-grams that occur in both generated summary and reference summary, and $\mathrm{Count}(\mathrm{N-gram})$ is the number of N-grams in reference summary. ROUGE-SU4 measures the overlap of skip-bigrams between a generated summary and a set of reference summaries with a maximum distance of four words. It is different from ROUGE-2 as it allows for maximum gap of four words between the overlapping two words. ROUGE-L is based on the Longest Common Subsequence (LCS), and it calculates the ratio between the longest common subsequence of the generated summary and the length of the reference summary. Let X represent a generated summary and Y be a reference summary consisting of n words. The calculation of ROUGE-L is as follows:

$$\mathrm{ROUGE-L} = \frac{\mathrm{LCS}(X, Y)}{n} \tag{7}$$

where $\mathrm{LCS}(X, Y)$ is the length of the longest subsequence of X and Y.

This paper uses the ROUGE toolkit (version 1.5.5), and adopts the same ROUGE settings[6] that are commonly used on the DUC datasets and Multi-News dataset for multi-document summarization. Guided by the state-of-the-art methods, this paper reports ROUGE recall on DUC datasets and ROUGE F1-score on the Multi-News dataset, respectively.

B. EXPERIMENTAL SETTINGS

1) SELECTION OF PRE-TRAINED MODEL

The centroid-based multi-document summarization method based on different pre-trained sentence embedding models has been studied in [1], which verifies the effectiveness of sentence embedding representations for multi-document summarization, and shows that using different sentence embedding models would affect the performance of summarization. Its results show that the USE-DAN model [18] is one of the best performing sentence embedding models for multi-document summarization.

In order to focus on evaluating the performance of the proposed multi-document summarization method and not be affected by different embedding models, this study uses the USE-DAN model[7] to encode sentences. In order to unify the representation of sentences and documents and preserve the relationship between documents and sentences, this study obtains the embedding vector of each document $d_i \in D$ by calculating the average of the sentence embedding vectors of all sentences contained in the document. Furthermore, the work in [19] has shown that sentence average is a strong approach to obtain document embedding.

2) DETERMINATION OF HYPERPARAMETERS K AND δ

Since different values of hyperparameters would affect the results of the multi-document summarization method, this study determines their values both theoretically and experimentally.

a: ESTIMATION OF THE HYPERPARAMETER K IN K-MEANS ALGORITHM

The method needs to select sentences not only from the root node of the class tree, but also from as many sub nodes as possible, so as to mine both commonality and specificity information from the input documents. Thus, when generating the sub nodes of the second layer of class tree, the value of k in k-means algorithm should not be set too large. Otherwise, under the limitation of the pre-given summary length, the sub nodes participating in sentences selection cannot cover all input documents, resulting in the generated summary being unable to contain the specificity information of some subclasses of the input documents.

The approximate number of sentences required to generate a summary can be estimated by (average length of target summaries) ÷ (average length of sentences in source documents), i.e., 4.65 for DUC'2004, 3.93 for DUC'2003, 4.37 for DUC'2002. Based on the estimation, when generating the sub nodes of the second layer of class tree, the hyperparameter k of the k-means algorithm should be set within the range of $[2, 4]$ (i.e., minimum number of sentences estimated for different datasets-1). For simplicity, when generating the sub nodes of the third layer and subsequent layers, this study sets k in the k-means algorithm to 2.

[6] ROUGE-1.5.5 with parameters "-n 2 -2 4 -u -m -r 1000 -f A -p 0.5" and "-l 100" for DUC'2002 and DUC'2003; "-b 665" for DUC'2004; "-l 264" for Multi-News.

[7] universal-sentence-encoder.

TABLE 1. Description of datasets, including DUC'2002, DUC'2003, DUC'2004, and Multi-News. For each dataset, the number of news sets it contains, the total number of news documents it contains, the number of reference summaries for each news set, and the average length of sentences in news documents are displayed.

Dataset	Number of News Sets	Number of Docs	Number of Reference Summary of each news set	Number of Words of each sentence in Docs
DUC'2002	59	567	2	22.86
DUC'2003	30	298	4	25.43
DUC'2004	50	500	4	25.38
Multi-News	5622	15326	1	22.24

b: ESTIMATION OF THE HYPERPARAMETER δ IN COMMONALITY-SPECIFICITY SCORE

The hyperparameter δ in Commonality-Specificity score illustrates the degree of attention paid to the relevance of each sentence to the documents in its own node. Therefore, the hyperparameter δ in score^{CS} cannot be set too small theoretically.

In order to determine the exact values of these two hyperparameters k and δ, this study employs a procedure similar to that used by Lamsiyah et al. [1] and Joshi et al. [16]. A small held-out set is built by randomly sampling 25 news sets from the validation set of the Multi-News dataset, which contains a total of 5622 news sets. Then, a grid search is performed for these two hyperparameters: $k \in [2, 4]$ with constant step of 1, $\delta \in [0, 1]$ with constant step of 0.1. It totally contains 33 feasible combinations. Next, the ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4 scores are calculated for each combination on the held-out set. And the results show that the combination of $k = 3$ and $\delta = 0.9$ provides the most optimal scores on ROUGE-1, ROUGE-2, ROUGE-L, and ROUGE-SU4 metrics. Thus, this study chooses them as final values of the two hyperparameters, which are consistent with the theoretical analysis of the two hyperparameters.

C. EVALUATIONS

1) ABLATION STUDY

Two groups of comparative experiments are carried out to verify the effectiveness of the proposed method: (1) verify the effectiveness of mining both commonality and specificity of documents for multi-document summarization; (2) verify the effectiveness of using documents hierarchical clustering for multi-document summarization other than using sentences hierarchical clustering. Due to the randomness of k-means, each experiment was run three times to get the intermediate results.

a: VERIFY THE EFFECTIVENESS OF MINING BOTH COMMONALITY AND SPECIFICITY OF DOCUMENTS

In order to avoid being affected by other factors and focus on evaluating the effect of considering both commonality and specificity information of documents for multi-document summarization, in this part, the method uses only the Commonality-Specificity score to score sentences (i.e., $\alpha = 1, \beta = 0, \gamma = 0$), and generates summaries for DUC datasets (denoted as Ours$_{\text{onlyCS}}$).

For fair comparisons, different variant methods are designed as follows:

- **Comp$_1$: only considering the commonality of all documents.** The variant method Comp$_1$ only focuses on the commonality of all documents. It does not use any clustering algorithms and only scores sentences by calculating the cosine similarity between sentence embeddings and the centroid embedding of all documents, and then it selects the higher scored sentences to form a summary.
- **Comp$_2$: only considering the specificity of subclasses of documents.** The variant method Comp$_2$ only focuses on the specificity of each subclass of documents. It uses the k-means algorithm to cluster documents, and then it uses the Commonality-Specificity score, which is defined in this paper, to score sentences in each subclass. Finally, it selects the highest scored sentence from each subclass to form a summary.
- **Comp$_3$: similar to Comp$_2$ but only considering the similarity between sentences in each subclass and the centroid of the subclass.** The variant method Comp$_3$ uses the k-means algorithm to cluster documents, and then it scores each sentence in each subclass by only calculating the cosine similarity between the sentence embedding and the centroid embedding of the subclass, and finally, it selects the highest scored sentence from each subclass to form a summary. i.e., for each sentence in each subclass, Comp$_3$ only focuses on the relevance of the sentence to the documents in the subclass and ignores the irrelevance to the documents not in it.

Table 2 displays the experimental results on three DUC datasets: DUC'2002, DUC'2003, and DUC'2004. The higher ROUGE scores indicates that the generated summaries are more similar to those written by experts.

The difference between Ours$_{\text{onlyCS}}$ and Comp$_1$ is that Comp$_1$ only selects the sentences expressing the commonality of all documents for generating summary while Ours$_{\text{onlyCS}}$ selects both the sentences expressing the commonality of all documents and the sentences expressing the specificity of some important subclasses of these documents for generating summary. The superiority of the Ours$_{\text{onlyCS}}$ method over Comp$_1$ is evident across all metrics for each dataset.

TABLE 2. Comparison results of different variant methods on DUC datasets, about whether or not the commonality and specificity of documents are considered. Ours$_{\text{onlyCS}}$ is the proposed method that use only the Commonality-Specificity score to score sentences.

	Method	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU4
DUC'2002	**Ours$_{onlyCS}$**	**34.81**	**7.32**	**30.11**	**11.48**
	Comp$_1$	33.23	6.95	28.78	10.99
	Comp$_2$	32.90	6.20	28.37	10.37
	Comp$_3$	32.86	6.12	28.46	10.31
DUC'2003	**Ours$_{onlyCS}$**	**37.89**	**8.27**	**32.37**	**12.85**
	Comp$_1$	36.77	8.22	31.29	12.7
	Comp$_2$	35.31	7.05	30.24	11.39
	Comp$_3$	35.03	7.02	30.07	11.29
DUC'2004	**Ours$_{onlyCS}$**	**38.36**	**8.66**	**33.94**	**13.30**
	Comp$_1$	36.44	8.03	32.08	12.61
	Comp$_2$	36.38	7.91	31.78	12.28
	Comp$_3$	36.85	8.18	32.25	12.44

Furthermore, a significant average improvement of 1.54 ROUGE-1 score and 1.42 ROUGE-L score was observed across the three DUC datasets. Thus, it is crucial to consider the specificity information of document subclasses in order to generate a diversity multi-document summary.

The key distinction between Ours$_{\text{onlyCS}}$ and Comp$_2$ is that Comp$_2$ exclusively selects sentences that express the specificity of each subclass for generating the summary, without considering sentences that convey the commonality of all documents. Ours$_{\text{onlyCS}}$ consistently outperforms Comp$_2$ across all metrics for each dataset, demonstrating its superior performance. Moreover, the average improvement across the three DUC datasets was as high as 2.16 ROUGE-1 score, 2.01 ROUGE-L score, 1.03 ROUGE-2 score, and 1.2 ROUGE-SU4 score. Hence, it is important to take into account the commonality information of all documents to enhance the comprehensiveness of the generated multi-document summary.

Comp$_3$ differs from Ours$_{\text{onlyCS}}$ in that it selects sentences from each subclass by only considering the commonality of the documents within the subclass without considering the differences with the documents outside the subclass, and disregards sentences that express the commonality of all documents. Ours$_{\text{onlyCS}}$ also exhibits significant superiority over Comp$_3$ across all metrics on three DUC datasets.

The method Ours$_{\text{onlyCS}}$ first selects sentence based on the commonality of all documents, and then selects sentences based on the specificity of different subclasses, which is in line with the way of humans summarizing multiple documents. Therefore, this method can generate summaries that are more similar to those written by experts. Additionally, this method outperforms the three variant methods on all three datasets, which demonstrates its strong robustness.

b: VERIFY THE EFFECTIVENESS OF USING DOCUMENTS HIERARCHICAL CLUSTERING OTHER THAN SENTENCES HIERARCHICAL CLUSTERING

In order not to be affected by other factors, in this part, the proposed method also uses only the Commonality-Specificity score to score sentences (Ours$_{\text{onlyCS}}$), and generates summaries on three DUC datasets. For a fair comparison, the variant method is designed as follows:

- **Comp$_4$: similar to Ours$_{\text{onlyCS}}$ but hierarchically clustering all sentences and constructing the class tree of sentences.** The variant method Comp$_4$ first converts the document collection into a sentence collection, and then it uses the same hierarchical clustering algorithm introduced in this paper to cluster all sentences to construct a class tree of sentences, where each node is a group of sentences. Next, it uses the Commonality-Specificity score to score sentences in each node. Finally, it uses the same sentences selection method proposed in this paper to select sentences from the class tree to form a summary.

Table 3 displays the comparison results between Ours$_{\text{onlyCS}}$ and Comp$_4$ on three DUC datasets. The difference between Ours$_{\text{onlyCS}}$ and Comp$_4$ is that Ours$_{\text{onlyCS}}$ selects the sentences expressing the commonality of all documents and the sentences expressing the specificity of some important subclasses of these documents for generating summary, while Comp$_4$ selects the sentences expressing the commonality of all sentences in all documents and the sentences expressing the specificity of some important subclasses of all sentences for generating summary. Ours$_{\text{onlyCS}}$ consistently exhibits superior performance compared to Comp$_4$ across all metrics for each dataset. Additionally, a substantial average enhancement of 1.25 ROUGE-1 score, 0.97 ROUGE-L score, 0.75 ROUGE-2 score, and 0.75 ROUGE-SU4 score was observed across the three DUC datasets. Due to its operation of dividing documents into individual sentences, Comp$_4$ lacks the ability to establish comparisons between documents, thus limiting its ability to discover the related information between documents, which play a crucial role in multi-document summarization. Moreover, Ours$_{\text{onlyCS}}$ (the proposed method based on document hierarchical clustering) outperforms Comp$_4$ (the variant method based on sentence hierarchical clustering) on all three datasets, demonstrating its strong robustness.

2) COMPARISONS WITH STATE-OF-THE-ART METHODS

The Commonality-Specificity score can be combined with the Non-redundant score and the Position score together to score and select sentences. This section compares the

TABLE 3. Comparison results of using documents hierarchical clustering in the proposed method and using sentences hierarchical clustering in the method on DUC datasets. Both Ours$_{\mathrm{onlyCS}}$ and Comp$_4$ use only the Commonality-Specificity score to score sentences.

	Method	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU4
DUC'2002	**Ours**$_{onlyCS}$	**34.81**	**7.32**	**30.11**	**11.48**
	Comp$_4$	33.50	6.48	29.09	10.63
DUC'2003	**Ours**$_{onlyCS}$	**37.89**	**8.27**	**32.37**	**12.85**
	Comp$_4$	36.63	7.88	31.48	12.31
DUC'2004	**Ours**$_{onlyCS}$	**38.36**	**8.66**	**33.94**	**13.30**
	Comp$_4$	37.17	7.65	32.93	12.45

TABLE 4. ROUGE scores of different methods on DUC'2004 dataset. The best performing method for each metric is indicated by *.

Method	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU4
Unsupervised methods				
Lead	32.37	6.38	28.68	10.29
LexRank [3]	37.32	7.84	33.18	12.53
Centroid$_{BOW}$ [4]	37.03	8.19	32.48	12.68
GreedyKL [20]	37.99	8.54	33.03	13.02
OCCAMS_V [21]	37.51	9.42*	33.69	13.12
Ranking-clustering [2]	37.87	9.35	-	13.25
SummPip [22]	36.30	8.47	-	11.55
Centroid$^{Run1}_{embedding}$ [1]	36.92	8.20	32.53	12.72
Centroid$^{Run4}_{embedding}$ [1]	38.12	9.07	34.15	13.44
Supervised methods				
PG-MMR [23]	36.42	9.36	-	13.23
CopyTransformer [24]	28.54	6.38	-	7.22
Hi-MAP [7]	35.78	8.90	-	11.43
BART-Long-Graph [25]	34.72	7.97	-	11.04
Primera [26]	35.1	7.2	17.9	-
Ours$_{onlyCS}$	**38.36**	**8.66**	**33.94**	**13.30**
Ours$_{Final}$	**39.28***	**9.31**	**35.02***	**13.75***

proposed method with existing competitive unsupervised and supervised multi-document summarization methods, and lists the results of the proposed method using only Commonality-Specificity score (i.e., score$^{\mathrm{CS}}$) and using the combination of three scores (i.e., score$^{\mathrm{final}}$), respectively.

a: DETERMINATION OF HYPERPARAMETERS α, β, AND γ

The hyperparameters α, β, and γ in score$^{\mathrm{final}}$ illustrate different degrees of attention paid to the Commonality-Specificity score, the Non-redundant score, and the Position score, respectively. Theoretically, α cannot be set too small because the proposed method focuses on mining both commonality and specificity information of documents for multi-document summarization.

To determine the exact values of the three hyperparameters in score$^{\mathrm{final}}$, this study also builds a small held-out set by randomly sampling 25 news sets from the validation set of the Multi-News dataset, and sets the value of hyperparameters k and δ as 3 and 0.9 respectively. Next, a grid search is performed for the three hyperparameters: $\alpha, \beta, \gamma \in [0, 1]$ with constant step of 0.1 under the condition $\alpha + \beta + \gamma = 1$. The obtained values of the hyperparameters are 0.8, 0.1, 0.1 for α, β, and γ respectively, which are consistent with the theoretical analysis of the three hyperparameters.

b: RESULTS ON THE DUC'2004 DATASET

Table 4 compares the performance of the proposed method with both unsupervised methods and supervised deep learning-based methods on DUC'2004 dataset. **Ours**$_{\mathrm{onlyCS}}$ corresponds to the proposed method that uses only the Commonality-Specificity score to score sentences, and **Ours**$_{\mathrm{Final}}$ corresponds to the proposed method that scores sentences using score$^{\mathrm{final}}$, i.e., the combination of the three scores: Commonality-Specificity score, Non-redundant score, and Position score.

The unsupervised methods listed are some competitive baselines or state-of-the-art methods for extractive multi-document summarization. This study reproduces Centroid$_{\mathrm{embedding}}$ [1] using the USE-DAN sentence embedding model and lists its results on DUC'2004 dataset to compare with the proposed method, because both Centroid$_{\mathrm{embedding}}$ and the proposed method use the same sentence embedding model but Centroid$_{\mathrm{embedding}}$ is a centroid based method. As described in [1], Centroid$^{\mathrm{Run1}}_{\mathrm{embedding}}$ only uses the similarity between sentence embedding and the centroid embedding to score sentences, and Centroid$^{\mathrm{Run4}}_{\mathrm{embedding}}$ uses the combination of three scores introduced in [1] to score sentences. The results of other methods are directly taken from their original articles [20] or published materials.[8]

The listed supervised methods, including PG-MMR, CopyTransformer, Hi-MAP, BART-Long-Graph, and Primera, are first trained on large datasets, such as CNN, DailyMail, and Multi-News, and then tested on DUC'2004 dataset. The results are directly taken from their original articles.

[8] github/duc2004-results.

TABLE 5. ROUGE scores of different methods on Multi-News dataset. The best performing method for each metric is indicated by *.

Method	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-SU4
Unsupervised methods				
Lead	39.41	11.77	-	14.51
MMR	38.77	11.98	-	12.91
LexRank [3]	38.27	12.70	-	13.20
SummPip [22]	42.32	13.28	-	16.20
Spectral-BERT [27]	40.9	13.6	-	16.7
BART$_{fine-tuned}$ [28]	40.58	15.50	21.73	-
Centroid$^{Run4}_{embedding}$ [1]	42.93	14.04	27.7	17.27
Supervised methods				
PG-MMR [23]	40.55	12.36	-	15.87
CopyTransformer [24]	43.57	14.03	-	17.37
Hi-MAP [7]	43.47	14.89	-	17.41
DynE [29]	43.9	15.8*	22.2	-
MGSum [30]	44.75*	15.75	-	19.30*
Ours$_{Final}$	**44.04**	**14.15**	**39.74***	**18.19**

As shown in Table 4, for ROUGE-1 measure, the proposed method, both Ours$_{\text{onlyCS}}$ and Ours$_{\text{Final}}$, significantly outperforms all listed unsupervised and supervised methods. And for ROUGE-L and ROUGE-SU4 measures, the method Ours$_{\text{Final}}$ significantly outperforms all listed methods. For ROUGE-2 measure, the proposed method achieves comparable result with the state-of-the-art methods. The supervised methods yield worse results on DUC'2004 dataset than most unsupervised methods because these deep learning-based methods are trained on other datasets and tested directly on DUC'2004 dataset. The comparison results with Centroid$_{\text{embedding}}$ further illustrate the effectiveness of mining both commonality and specificity of documents for multi-document summarization.

c: RESULTS ON THE MULTI-NEWS DATASET

This study also compares the proposed method with some competitive or state-of-the-art unsupervised and supervised methods on Multi-News dataset. The results of these methods are directly taken from their original articles.

As shown in Table 5, the proposed method significantly outperforms all unsupervised methods on ROUGE-1, ROUGE-L, and ROUGE-SU4 metrics. By comparing with the supervised deep learning-based methods, which are both trained and tested on Multi-News dataset, the proposed method still achieves significantly better ROUGE-1, ROUGE-L, and ROUGE-SU4 scores than PG-MMR, CopyTransformer, Hi-MAP, and DynE methods. For ROUGE-L measure, the proposed method achieves the best result than all listed methods.

Overall, as an unsupervised and easy-to-implement method, the proposed method achieves considerable results. Moreover, the comparison experiments with different variant methods prove the effectiveness of mining both the commonality and specificity of documents for multi-document summarization.

V. CONCLUSION

This paper proposes a multi-document summarization method based on hierarchical clustering of documents, which makes use of the constructed class tree of documents to mine both the commonality information of all documents and the specificity information of some subclasses of documents for generating a summary. The experiments show that the proposed method significantly outperforms the variant methods that mine only commonality information or only specificity information, and outperforms the variant method based on sentences hierarchical clustering. Furthermore, as an easy-to-implement unsupervised method, the proposed method is superior to many competitive supervised and unsupervised multi-document summarization methods, and yields considerable results.

This paper has proven that utilizing the class tree constructed by documents hierarchical clustering is effective for multi-document summarization. In future work, we plan to explore other effective hierarchical clustering approaches for multi-document summarization task. Additionally, we will explore suitable document embedding representation methods for documents hierarchical clustering and multi-document summarization task.

REFERENCES

[1] S. Lamsiyah, A. El Mahdaouy, B. Espinasse, and S. E. A. Ouatik, "An unsupervised method for extractive multi-document summarization based on centroid approach and sentence embeddings," *Exp. Syst. Appl.*, vol. 167, Apr. 2021, Art. no. 114152.

[2] L. Yang, X. Cai, Y. Zhang, and P. Shi, "Enhancing sentence-level clustering with ranking-based clustering framework for theme-based summarization," *Inf. Sci.*, vol. 260, pp. 37–50, Mar. 2014.

[3] G. Erkan and D. R. Radev, "LexRank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Intell. Res.*, vol. 22, pp. 457–479, Dec. 2004.

[4] D. R. Radev, H. Jing, M. Stys, and D. Tam, "Centroid-based summarization of multiple documents," *Inf. Process. Manag.*, vol. 40, no. 6, pp. 919–938, Nov. 2004.

[5] G. Rossiello, P. Basile, and G. Semeraro, "Centroid-based text summarization through compositionality of word embeddings," in *Proc. MultiLing Workshop Summarization Summary Eval. Across Source Types Genres*, Valencia, Spain, 2017, pp. 12–21.

[6] K. Sarkar, "Sentence clustering-based summarization of multiple text documents," *TECHNIA-Int. J. Comput. Sci. Commun. Technol.*, vol. 2, no. 1, pp. 325–335, 2009.

[7] 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.

[8] M. Gambhir and V. Gupta, "Recent automatic text summarization techniques: A survey," *Artif. Intell. Rev.*, vol. 47, no. 1, pp. 1–66, Jan. 2017.

[9] D. Wang, T. Li, S. Zhu, and C. Ding, "Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization," in *Proc. 31st Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2008, pp. 307–314.

[10] M. Mohd, R. Jan, and M. Shah, "Text document summarization using word embedding," *Exp. Syst. Appl.*, vol. 143, Apr. 2020, Art. no. 112958.

[11] O. Rouane, H. Belhadef, and M. Bouakkaz, "Combine clustering and frequent itemsets mining to enhance biomedical text summarization," *Exp. Syst. Appl.*, vol. 135, pp. 362–373, Nov. 2019.

[12] Y. Ko and J. Seo, "An effective sentence-extraction technique using contextual information and statistical approaches for text summarization," *Pattern Recognit. Lett.*, vol. 29, no. 9, pp. 1366–1371, Jul. 2008.

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

[14] H. P. Edmundson, "New methods in automatic extracting," *J. ACM*, vol. 16, no. 2, pp. 264–285, Apr. 1969.

[15] Y. Ouyang, W. Li, Q. Lu, and R. Zhang, "A study on position information in document summarization," in *Proc. Coling*, Beijing, China, Aug. 2010, pp. 919–927.

[16] A. Joshi, E. Fidalgo, E. Alegre, and L. Fernández-Robles, "SummCoder: An unsupervised framework for extractive text summarization based on deep auto-encoders," *Exp. Syst. Appl.*, vol. 129, pp. 200–215, Sep. 2019.

[17] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. Text Summarization Branches Out*, Barcelona, Spain, Jul. 2004, pp. 74–81.

[18] D. Cer, Y. Yang, S.-Y. Kong, N. Hua, N. Limtiaco, R. S. John, N. Constant, M. Guajardo-Cespedes, S. Yuan, C. Tar, B. Strope, and R. Kurzweil, "Universal sentence encoder for English," in *Proc. Conf. Empirical Methods Natural Lang. Process., Syst. Demonstrations*, 2018, pp. 169–174.

[19] S. Sannigrahi, J. V. Genabith, and C. Espana-Bonet, "Are the best multilingual document embeddings simply based on sentence embeddings?" in *Proc. Findings Assoc. Comput. Linguistics*, Dubrovnik, Croatia, May 2023, pp. 2306–2316.

[20] K. Hong, J. Conroy, B. Favre, A. Kulesza, H. Lin, and A. Nenkova, "A repository of state of the art and competitive baseline summaries for generic news summarization," in *Proc. 9th Int. Conf. Lang. Resour. Eval.*, 2014, pp. 1608–1616.

[21] S. T. Davis, J. M. Conroy, and J. D. Schlesinger, "OCCAMS—An optimal combinatorial covering algorithm for multi-document summarization," in *Proc. IEEE 12th Int. Conf. Data Mining Workshops*, Dec. 2012, pp. 454–463.

[22] J. Zhao, M. Liu, L. Gao, Y. Jin, L. Du, H. Zhao, H. Zhang, and G. Haffari, "SummPip: Unsupervised multi-document summarization with sentence graph compression," in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 1949–1952.

[23] L. Lebanoff, K. Song, and F. Liu, "Adapting the neural encoder–decoder framework from single to multi-document summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process.*, Brussels, Belgium, 2018, pp. 4131–4141.

[24] 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.

[25] R. Pasunuru, M. Liu, M. Bansal, S. Ravi, and M. Dreyer, "Efficiently summarizing text and graph encodings of multi-document clusters," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Human Lang. Technol.*, 2021, pp. 4768–4779.

[26] W. Xiao, I. Beltagy, G. Carenini, and A. Cohan, "PRIMERA: Pyramid-based masked sentence pre-training for multi-document summarization," in *Proc. 60th Annu. Meeting Assoc. Comput. Linguistics*, Dublin, Ireland, 2022, pp. 5245–5263.

[27] K. Wang, B. Chang, and Z. Sui, "A spectral method for unsupervised multi-document summarization," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 435–445.

[28] T. Johner, A. Jana, and C. Biemann, "Error analysis of using bart for multi-document summarization: A study for English and German language," in *Proc. 23rd Nordic Conf. Comput. Linguistics*, 2021, pp. 391–397.

[29] C. Hokamp, D. G. Ghalandari, N. T. Pham, and J. Glover, "DynE: Dynamic ensemble decoding for multi-document summarization," 2020, *arXiv:2006.08748*.

[30] H. Jin, T. Wang, and X. Wan, "Multi-granularity interaction network for extractive and abstractive multi-document summarization," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 6244–6254.

BING MA is currently pursuing the Ph.D. degree with the Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, and the University of Chinese Academy of Sciences. She has published two papers in top journals and high-level conferences, such as *Expert Systems with Applications* in SCI Q1 journal. Her current research interests include natural language processing and resource space model.