The 33rd Conference on Computational Linguistics and Speech Processing (ROCLING 2021)
Taoyuan, Taiwan, October 15-16, 2021. The Association for Computational Linguistics and Chinese Language Processing

Unsupervised Multi-document Summarization for News Corpus with Key Synonyms and Contextual Embeddings

Yen-Hao Huang[1], Ratana Pornvattanavichai[2], Fernando Henrique Calderon Alvarado[3], Yi-Shin Chen[*]
Institute of Information Systems and Applications[12]
Department of Computer Science[*]
National Tsing Hua University, Hsinchu, Taiwan
Social Networks and Human-Centered Computing Program[3]
Institute of Information Sciences, Academia Sinica, Taipei, Taiwan
{yenhao0218[1], fhcalderon87[3], yishin[*]}@gmail.com, trp3110@hotmail.com[2]

Abstract

Information overload has been one of the challenges regarding information on the Internet. It is no longer a matter of information access, instead, the focus has shifted towards the quality of the retrieved data. Particularly in the news domain, multiple outlets report on the same news events but may differ in details. This work considers that different news outlets are more likely to differ in their writing styles and the choice of words, and proposes a method to extract sentences based on their key information by focusing on the shared synonyms in each sentence. Our method also attempts to reduce redundancy through hierarchical clustering and arrange selected sentences on the proposed orderBERT. The results show that the proposed unsupervised framework successfully improves the coverage and coherence, while also reducing the redundancy for a generated summary. Moreover, due to the process through which the dataset is obtained, a data refinement method is proposed to alleviate the problem of undesirable texts, which result from the process of automatic scraping.

1 Introduction

Text summarization is defined as the act of expressing the most important facts or ideas about something or someone in a short and clear form. The two most common types of summarizations are classified by their output types, known as extractive summarization and abstractive summarization. The first extracts sentences from the original document, then aggregates the extracted salient text units together to output a summary. Meanwhile, abstractive summarization is the act of paraphrasing to generate a summary that still maintains the main idea of the original document. Summarization can also be classified by the number of source documents they utilize, namely, single document and multi-document summarization. In this research, we are focusing on extractive summarization of multi-document news articles aiming to provide generic summaries.

Although single and multi-document summarization share common challenges which are coverage, the amount of main ideas are being covered in the summary, and coherence, the connection and consistency of the content in the extracted summary, there is an additional challenge that multi-document summarization has to address, redundancy. Redundancy occurs when a piece of information is being expressed more than once in the summary, especially for multi-document tasks. A good summary should not contain sentences that repeat the same ideas. Therefore, we propose a framework to address the problem of coverage and redundancy explicitly, then integrate them together to ensure coherence in the final step.

Our contributions to address the problems of coverage, redundancy, and coherence in multi-document summarization can be summarized as follows:

- We propose an unsupervised framework to construct a sentence-level graph with shared synonyms to address coverage.
- The redundancy level of the extracted summary is reduced by utilizing hierarchical clustering on BERT embeddings.
- Our experiment shows that the proposed orderBERT provides better coherence than original position ordering for

[*]The corresponding author.

news corpus.

2 Related Work

There are several well-known approaches for multi-document extractive summarization. Details are introduced in following sections.

2.1 Frequency-based Methods

One of the most well-known and explainable method in summarization is to utilize the term frequency of the content. Early researches on multi-document summarization focused on extracting words and their lexical characteristics to solve content selection. Some approaches to maximize coverage of the content utilized a classifier (Conroy et al., 2004; Ramanujam and Kaliappan, 2016; Hennig et al., 2008) or directly assigned a score to sentences (Schiffman et al., 2002; Lin and Hovy, 2002; Meena and Gopalani, 2014) to identify the importance of those sentences. Other works introduced "concept" (Schluter and Søgaard, 2015) or "event" (Filatova and Hatzivassiloglou, 2004) to represent the important text unit that best covers the main idea of the source documents. SumBasic (Nenkova and Vanderwende, 2005) was based on the relation of words frequency in a document cluster and human summaries. The study showed that the higher the frequency in the document cluster, the higher the probability of the word to appear in the human summary. Despite the different detailed approaches, one thing these methods have in common is utilizing the term frequency of the content. However, with the nature of multiple source documents, the limitation to word frequency is their lexical form. If only the lexical form is considered, we would be limited to capturing only some part of the information.

2.2 Greedy-algorithm Methods

A greedy-algorithm is an intuitive algorithm that is used in optimization problems. The process is to make the optimal choice at each step in order to find the overall optimal way to solve the whole problem. Some works integrated submodularity (Dasgupta et al., 2013) and minimum dominating set (Shen and Li, 2010) with the algorithm to solve the text summarization task. Other work such as the KLSum (Haghighi and Vanderwende, 2009) focused on minimizing the divergence between the true distribution and the approximating distribution. One of the most well-known method is the Maximal Marginal Relevance(MMR) (Carbonell and Goldstein, 1998) which tried to reduce redundancy while maintaining relevance in the retrieved text unit. The method performs well for the task of information retrieval where the task is to retrieve documents related to a user's query. However, for the task of multi-document summarization, there is no user's query labelled which means that further determination of the reference needs to be made. Moreover, Takamura and Okumura (2009) also stated that although relevance and redundancy are taken into consideration, no global viewpoint is given. We found that these methods do address redundancy in nature, but not explicitly.

2.3 Graph-based Methods

Sentence-level graph-based extractive summarization generally assigns each sentence to the nodes and determine the edges of them depending on their relationship. The centroid-based method (Radev et al., 2004b), MEAD (Radev et al., 2004a), TextRank (Mihalcea and Tarau, 2004), and LexRank (Erkan and Radev, 2004) are some of the common graph-based methods in previous works. However, the limitation to the nodes connections are that they either rely on whole sentence similarity which needs a defined threshold to determine their connection, or only considers the lexical form of the words that overlap between sentences.

3 Methodology

This work aims to overcome the three main challenges introduced in Section 1 and proposes a framework toward coverage, redundancy, and coherence as shown in Figure 1.

3.1 Data Refinement

In our work, we utilized the Multi-News dataset (Fabbri et al., 2019), a large multi-document news dataset consisting of 56,216 news clusters, obtained through automatic scraping. According to Kryściński et al. (2019), manual inspection of data is impractical and expensive and mostly limited to removing only markup structure and obvious noises. Despite the fact that the authors of

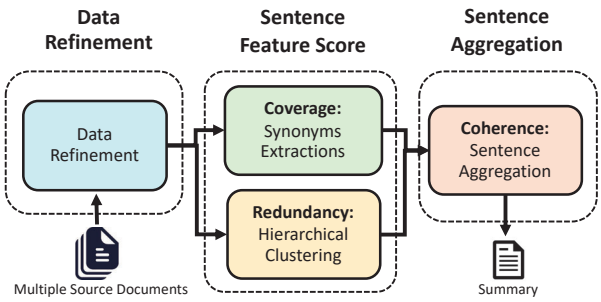

Figure 1: Overall Framework

Multi-News have provided an updated version of their dataset, we can still detect some unrelated source documents throughout the dataset. Therefore, in the process to refine data quality, we have divided noises into two categories: retrieval noises and content noises. **Retrieval noises** are error texts from the process of content retrieval. These texts are found to be duplicated within their own clusters and across different clusters. There are two major retrieval noises: (1) *Duplicated source documents within the same news clusters*, that are possibly result from a news service supplied articles to more than one outlets. As some news outlets have limited resources, they rely on information supplied by other news service instead. As a result, news articles from different outlets may be identical and ensure that the dataset does not unintentionally allow more weight to a specific document. (2) *Duplicated source documents across different news clusters*, which are generally the result of scraping error messages. We found that the same error messages appear throughout multiple documents. Therefore, regardless of the news clusters, there are risks that these error messages can appear as one of the source documents. With a list of scraping error messages, source documents in the list are removed at the end. **Content noises** refers to the source documents' lengths and the semantic similarity of source documents within their own clusters. (1) *Single sentence source document* are undesirable news articles. They are more likely to be error messages generated when scraping data from the website. Although they sometimes share the same words with other source documents within the cluster, but are entirely unrelated or can not provide enough information to be considered an independent source document. (2) *Discrepancy between source documents* of the same news cluster are also undesirable characteristics of source documents that needs elimination. In order to ensure that the remaining source documents are related to the reference summary but not totally identical, we compare the similarity of each source document by embedding them utilizing Sentence-BERT(Reimers and Gurevych, 2019) and compare the cosine similarity of all source documents within the same news clusters. We empirically set to filter unqualified news clusters that contain 2 source documents with cosine similarity less than 0.5 or equal to 1.0.

To comply with the task of multi-document summarization, after refining all the unrelated source documents, we finally eliminated news clusters with only one source document.

3.2 Sentence Feature Score

There are two components for sentence score features designed for coverage and redundancy factors, respectively.

3.2.1 Coverage Factor: Synonyms Extractions

This component aims to capture the overlapping content despite the lexical differences. Documents written by different authors are likely to be different in the writing styles and the word usage. Within the same news clusters, despite the different words, authors still have to deliver the same information. Therefore, this work considers that synonym is the key to identify the overlapping information between documents where different words are used. To extract synonyms of different source documents, a two-step approach is proposed and described as follow:

Wordnet Synonyms Extractions. WordNet (Miller, 1995) is a large lexical database links nouns, verbs, adjectives, and adverbs to sets of synonyms, known as synsets. Synsets are linked by their conceptual-semantic and lexical relations, resulting in a network of related words and concepts. To identify the overlapping content, we utilized the synonymy semantic relations from WordNet to capture the common word senses and their meaning between sentences.

First, the part-of-speech (POS) tagging was adopted to categorize words into their syntac-

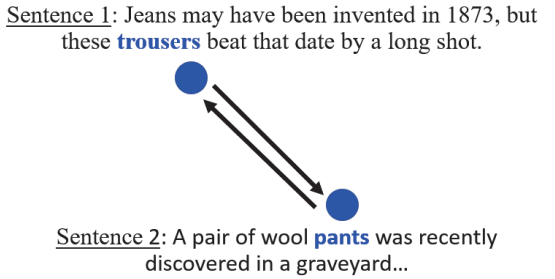

Figure 2: An Example of Sentences Connection.

tic category: nouns, verbs and adjectives. To determine which syntactic category to include, this work considers the journalistic questions when readers read news articles. In general, we assumed that what readers are most interested in an article is the "Who" did "What" in the article. According to the two questions, candidate answers usually result with nouns and a combination of other POS such as verbs and adjectives. Then, we performed synonym extractions by utilizing NLTK for WordNet.

Graph-based Sentence Scoring. With the list of synonyms for each word, a sentence graph was constructed where each individual sentence is the vertex and sentences were connected if there was any shared synonyms. The sentence graph is formally defined in Definition 1.

Definition 1 (Sentence Graph)**.** *Let D, S denote a set of source documents and sentences in documents D, respectively, such that $D = \{d_i\}$, $S = \{s_j\}$. The directed graph for each news cluster is denoted as:*

$$G_D = (S, E) \tag{1}$$

where S represents all the sentence vertices in D, E denotes the edges between two sentences that shared common synonyms.

To construct a sentence graph G_D, originally, one edge was constructed for each synonym; however, each undirected edge was further converted to two directed edges toward each sentence. Figure 2 shows an example of the connection between sentences.

After constructing a graph for each news cluster, PageRank (Page et al., 1999) algorithm was adopted to calculate the score of each vertex as in Equation 2.

$$PR(s_i) = (1 - \delta) + \delta * \sum_{s_j \in In(s_i)} \frac{PR(s_j)}{|Out(s_j)|} \tag{2}$$

where $PR(s_i)$ is the score of $(s_i) \in S$, δ is a parameter usually set to 0.85 by default, $In(s_i)$, $Out(s_i)$ are the inbound link and outbound link of sentence vertex s_i, respectively. The PageRank score $PR(s)$ was treated as the final sentence scores, which determines the level of coverage for each sentence.

3.2.2 Redundancy Factor: Hierarchical Clustering

To minimize redundancy, sentences S were first embed by Sentence-BERT (Reimers and Gurevych, 2019) to retrieve the contextual embeddings $X \in R^{|S| \times dim}$ where dim is the hidden size of Sentence-BERT. The agglomerative hierarchical clustering was adopted at sentence level for each news cluster with sentence embeddings X. The metrics of hierarchical clustering were set as the group average similarity, which takes into consideration all sentence features within the cluster as in Definition 2.

Definition 2 (Sentence Group Similarity)**.** *Let C_1, C_2 denote two sentence group, their corresponding features are presented as X_{C_1}, X_{C_2}. The sentence group similarity could be calculated by Equation 3.*

$$sim(C_1, C_2) = \sum sim(X_i, X_j)/|X_{C_1}| * |X_{C_2}| \tag{3}$$

where X_i and X_j are sentence features, s_i belongs to group C_1 and s_j belongs to group C_2.

In the process of clustering, we evaluated the quality of the clustering utilizing two cluster validity indices, Silhouette Score and DB Index or Davies—Bouldin Index as shown below.

The calculation of Silhouette Score is defined in Equation 4.

$$silh(s) = \frac{dis_{inter}(s) - dis_{intra}(s)}{max\{dis_{intra}(s), dis_{inter}(s)\}} \tag{4}$$

where $silh(s)$ is the Silhouette Score of the sentence s, $intra_d(s)$ is the average Euclidean distance on sentence feature $x \in X$ between

sentence s and all the other sentences in the cluster $C, s \in C$, $inter_d(s)$ is the minimum average distance from sentence s to all clusters $\{\check{C}|s \notin \check{C}\}$. The score ranges from -1 to 1. The higher values indicate that objects within the cluster are more similar to their own clusters and less similar to other clusters.

For DB index DB, it was calculated as follows:

$$DB = \frac{1}{k}\sum_{m=1}^{k}\max_{m\neq n} R_{m,n} \tag{5}$$

where $R_{m,n}$ is the within-to-between cluster distance ratio for the ith and jth clusters

$$R_{m,n} = \frac{dis_m + dis_n}{dis_{m,n}} \tag{6}$$

dis_m is the average Euclidean distance between each sentence in the mth cluster and the centroid of the mth cluster. dis_n is the average distance between each point in the nth cluster and the centroid of the nth cluster. $dis_{m,n}$ is the distance between the centroids of the mth and nth clusters. The minimum value of DB Index is 0. The lower value indicate that objects are less dispersed.

With the combination of the two indices, the optimal threshold for clustering is then able to obtained by selecting a threshold that can get the highest Silhouette score and the lowest value for DB index. Finally, sentence cluster label l_s was retrieved for each sentence s, which determines which sentences are semantically similar.

3.3 Sentence Aggregation

Sentence aggregation takes two input, namely, the sentence score $PR(s)$ and the sentence cluster label l_s. Both of the values were used as the criteria to select and rearrange sentences in this phase.

3.3.1 Sentence Selection

The goal of this step was to select top-N representative sentences, where $N = 9$ is the average number of sentences in the reference summary. To reduce redundancy, we first grouped the sentences by l_s. For each cluster, a candidate sentence was selected which had the highest $PR(v)$. By utilizing both the coverage indicating value (sentence score), and the redundancy grouping (sentence cluster label), we considered that the selected sentences were the top salient and least redundant sentences.

In cases that there were more than N clusters within the news cluster, the top-N sentences with the least position value in their original documents were selected.

3.3.2 Sentence Ordering

News content are known for their lead sentences bias where the main content and the flow are based on the first few sentences of the articles. However, for multi-document summarization, the common method leveraging original position of the sentences might not give the best fluent order for a summary.

Hence, we proposed to fine-tuned BERT for a modified next sentence prediction task (Devlin et al., 2019) with the extracted top-N sentences, namely orderBERT. Specifically, inverse-order sentences within source article are added as false samples. In the original paper, BERT was trained to predict whether the observed sentences come from the same or distinct documents, but not to manage the orders. Meanwhile, the orderBERT was trained to predict whether the second sentence is next order to the first sentence. The higher the *continuation value* (CV) output from orderBERT, the more continuity the given 2 sentences are.

For reordering, an *anchor sentence* was initialized as the least original position value sentence among top-N sentences. If there were many sentences at first position, the one with the highest $PR(s)$ was selected. The *anchor sentence* was further paired with all the other top-N sentences for orderBERT to obtain the corresponding CVs. The sentence with the highest CV was assigned as next anchor sentence and continue throughout the remaining top-N sentences. This algorithm generally take $O(N^2)$ time complexity; however, as the N is small, it was not too time consuming.

4 Experiment

4.1 Dataset

This work experiments on a multi-document news corpus, namely Multi-News (Fabbri et al., 2019). The dataset contains the reference summary obtained from Newser website [1] and multiple source documents of the

[1] www.newser.com

same news story. The total number of source documents per news story ranges from 2 to 10 documents per reference summary. The data refinement process is conducted as mention in Section 3.1. The statistics of the original and refined dataset are shown in Table 1.

Source #	Original	Refined
2	3049/3072/23894	2843/2854/22298
3	1565/1574/12707	1521/1531/12367
4	608/624/5022	583/586/4764
5	223/206/1873	198/188/1656
6	113/82/763	86/64/657
7	38/41/382	34/32/330
8	15/14/209	12/11/163
9	10/7/89	7/7/61
10	1/2/33	1/0/19
Total	**56,216**	**52,873**

Table 1: Data Statistics (test/validation/train)

4.2 Experimental Setup

To evaluate the performance, as our methods are unsupervised, we have experimented on our refined dataset with five other unsupervised baselines including *common extractive summarization baseline*: (1) **Lead-3** sentences, which takes first 3 sentences of each source documents to aggregate as the extracted summary; *frequency-based*: (2) **SumBasic** (Haghighi and Vanderwende, 2009) computes the probability distribution over the input words. For each input sentences, a weight equal to the average probability of the words are assigned as the sentence score. With top score sentences selected, the words probability are updated for additional sentence selection until a designated summary length is reached; *greedy algorithm*: (3) **KLSum** (Nenkova and Vanderwende, 2005) selects sentences by minimizing the divergence between the true distribution in the original document and the approximating distribution in the summary; *graph-based*: (4) **LexRank** (Erkan and Radev, 2004) is sentence-level graph algorithm where edges between the nodes are assigned when the node pair exceeds a cosine similarity threshold, in our work 0.1 according to the best performance LexRank experiment. When calculating the weight of the edges, their idf value is also taken into consideration; and (5) **TextRank** (Mihalcea and Tarau, 2004), for the task of extracting salient sentences, is sentence-level graph algorithm. The weighted edges are calculated by dividing the overlapping words of two sentences with the length of each sentence. For LexRank and TextRank, Equation 2 is applied to obtain the final sentence score. Finally, all the algorithms were limited to select Top-$N = 9$ sentences as generated summary, which is the average sentence number in the reference summary.

The settings to implement the proposed method are illustrated below. As shown in Figure 3, the highest value for Silhouette score and the lowest value for DB index equation indicate that the best cosine similarity threshold was found at 0.9 and set as the ultimate parameter. The final number of selected sentence is 9 as baselines. To finetune the orderBERT proposed in Section 3.3.2, a pretrained base version of BERT is selected to optimize the next sentence prediction objective with batch size set as 32 for 5 epochs.

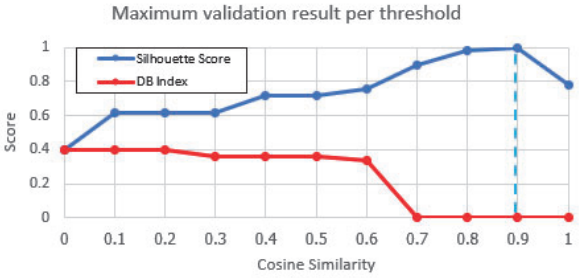

Figure 3: Optimal threshold from Silhouette Score and DB Index values

5 Results and Analysis

For the experimental results, this work carefully evaluates our main focuses separately in the following sections, which are coverage, redundancy, and coherence.

5.1 Coverage

To evaluate coverage, ROUGE score was selected by comparing 4 combinations of the different POS settings with our method as shown in Table 2. There are 3 different ROUGE scores adopted, which are ROUGE-1 (R-1), ROUGE-2 (R-2), ROUGE-SU (R-SU) for evaluating uni-gram, bi-gram overlaps and skip/uni-gram co-occurrence, respectively.

Method	Refined All			Refined Testing			Original Testing		
	R-1	R-2	R-SU	R-1	R-2	R-SU	R-1	R-2	R-SU
Lead-3	0.4114	0.1215	0.1545	0.4098	0.1210	0.1536	0.3964	0.1071	0.1406
LexRank	0.4174	0.1247	0.1620	0.4179	0.1253	0.1624	0.4154	0.1247	0.1611
TextRank	0.3963	0.1272	0.1446	0.3954	0.1271	0.1438	0.4016	0.1186	0.1494
SumBasic	0.3749	0.1020	0.1218	0.3769	0.1028	0.1231	0.3775	0.1041	0.1242
KLSum	0.3665	0.1012	0.1200	0.3672	0.1018	0.1203	0.3669	0.1027	0.1207
N. Syn.	**0.4272**	**0.1304**	**0.1680**	0.4268	**0.1315**	0.1681	**0.4218**	**0.1321**	**0.1663**
N.+V. Syn.	0.4266	0.1300	0.1674	0.4262	0.1310	0.1675	0.4209	0.1318	0.1655
N.+Adj. Syn.	0.4246	0.1291	0.1664	**0.4278**	0.1312	**0.1684**	0.4213	0.1319	0.1659
N.+V.+Adj. Syn.	0.4265	0.1299	0.1673	0.4275	0.1310	0.1681	0.4207	0.1317	0.1653
PG-ORIGINAL	Supervised Methods			*			0.4185	0.1291	0.1646
PG-BRNN							0.4280	0.1419	0.1675
CopyTransformer							0.4357	0.1403	0.1737
Hi-MAP							0.4347	0.1489	0.1741

Table 2: ROUGE Evaluation. Note that "Refined All" denotes the results from entire Multi-News dataset with refinement for all unsupervised methods; whereas, "Refined Testing" reports the results from only testing set with refinement, and "Original Testing" shows the results from testing set without refinement.

5.1.1 Performance

As observed from the results, our proposed methods that adapts synonyms to find sentence connections can help improve ROUGE score and outperform all unsupervised baselines. This implies that there are connections that are added when considering synonyms. We also found that within the each document itself, the author might utilize synonyms when talking about the same subject. Therefore, utilizing synonyms not only contribute to capturing connections between documents but within each document itself.

In addition to the unsupervised baselines, we also compare to supervised approaches, which are PG-ORIGINAL (Lebanoff et al., 2018), PG-BRNN (Gehrmann et al., 2018), CopyTransformer (Gehrmann et al., 2018), and Hi-MAP (Fabbri et al., 2019). It is worth mentioning that their results are obtained from Hi-MAP's paper for a fair comparison. Although our method can not outperform the supervised baselines, with data refinement, we can achieve comparable results to PG-BRNN. The possibles reasons are: first, our method is fully unsupervised; second, they summarize in an abstractive manner, which is suitable for multi-document tasks. These strengths will be further considered in our future work.

5.1.2 POS Connections Analysis

For different combinations of POS, we observe that utilizing only synonyms that are nouns performed the best. However, in the case of verbs and adjectives, we might need to consider the subject or object that is doing the action or being described. In order to improve the usage of verbs and adjectives, further improvements could focus on the noun that is directly associated with them.

5.1.3 Necessity of Data Refinement

Since we have proposed data refinement as part of our methodology to emphasize the importance of refined data, we analyse the results in comparison to other methods with the original dataset by the author. We compare our method before and after the refinement. Noted that we found some source documents without refinement to be about the same length or even shorter than the reference (golden) summary. For the mentioned instances, we utilize all sentences in the source document as the extracted summary for comparison. The results are shown in Table 2. We can observe that the proposed refinement could successfully improve most of the results on ROUGE including all the proposed methods. Only TextRank and SumBasic slightly decrease on their partial scores.

5.2 Redundancy

To evaluate performance of our hierarchical clustering method which is designed to address redundancy issues, we first test the redundancy reducing performance and study the ablation effects for the clustering on the coverage. Detailed analyses are discussed in below.

Method	Average Word #	
	max.	mean.
Lead-3	12.6	1.37
LexRank	19.75	1.50
TextRank	25.32	1.47
SumBasic	19.96	1.29
KLSum	14.06	1.42
N. Syn.	**9.54**	1.36
N. + V. Syn.	11.20	1.36
N. + Adj. Syn.	19.69	1.38
N. + V. + Adj. Syn.	11.20	**1.02**

Table 3: Redundancy Analysis

Method	ROUGE-1	
	w. Cluster.	w/o. Cluster.
N. Syn.	**0.4272**	0.4173
N.+ V. Syn.	**0.4266**	0.4187
N.+ Adj. Syn.	**0.4246**	0.4187
N.+ V.+ Adj. Syn.	**0.4265**	0.4167

Table 4: Ablation Study for Clustering on ROUGE

Technique	ROUGE-1	Average Word #
Hierarchical	0.42684	**1.48**
K-Means	0.42108	1.49

Table 5: Clustering Techniques Comparison

5.2.1 Hierarchical Clustering Influence on Reducing Redundancy

The maximum and mean value of the average occurrences for distinct words are calculated for generated summaries from each method with stopwords removed. The lower the redundancy, the fewer the word occurs in the summary. The results are shown in Table 3.

As observed from the table, the proposed methods (named with Syn.) generally have lower word occurrence for each distinct word, especially for our method with N.+V.+Adj. Syn. which has a lowest 1.02 on average. Although the other combinations of our methods did not have such big gaps as the N.+V.+Adj. Syn. approach, their occurrences are generally lower than the other baselines. This result implies that with the hierarchical clustering step, the extracted sentences for summaries contains less words that are redundant.

5.2.2 Hierarchical Clustering Influence on Boosting Coverage

Besides word occurrences, an ablation study of our methodology with/without hierarchical clustering was conducted with its evaluation based on ROUGE. As shown in Table 4, we found that hierarchical clustering not only helps reduce redundancy but also helps increase coverage. Results without hierarchical clustering are lower than with clustering. Without the clustering, the selected sentences may have high score but cover duplicated topics and contain redundant information. Therefore, reducing redundancy also contributes to the coverage of summaries.

5.2.3 Different Hierarchical Clustering Technique and their Performance

In addition to agglomerative hierarchical clustering, we also experimented with another common clustering technique, K-means, with the other settings remains the same. The performance of the 2 techniques in terms of coverage and redundancy is as shown in Table 5.

According to the result, we found that utilizing hierarchical clustering performs better than utilizing K-means in both coverage and redundancy performance. We found that the lower performance of K-means is most probably due to the pre-defined number of clusters. The limited number of clusters influence the degree in which sentences can be clustered. As for the agglomerative hierarchical clustering, the number of clusters are jointly decided by Silhouette Score and DB Index.

5.3 Coherence

For the coherence evaluation of our proposed orderBERT reordering, we conduct human evaluation with 14 participants to compare to other 2 ordering methods. The extracted sentences are obtained from our noun synonym method. Given an original article and three summaries generated by different reordering mechanisms, our questionnaire asks to respondents to answer two questions: (1) rate the fluency for each summary individually; and (2) rank the fluency from all summaries.

In Table 6, original position method has the most vote for score 5, which is 8% more than ours. However, considering score 4 and 5, the proposed orderBERT reordering method have totally 64% of vote that over original position's 43%. For average rating and ranking score in Table 7, the orderBERT reordering achieve top score over the other two methods. The above results show that the proposed method outperforms the common methods which reference to sentences' original position. Over-

Method	Rating Score				
	1	2	3	4	5
orderBERT	0%	7%	29%	**43**%	**21**%
Original Position	7%	21%	29%	14%	**29**%
Random	7%	29%	29%	**29**%	7%

Table 6: Coherence Rating Evaluation Result (1 is the least coherent, 5 is the most coherent rating)

Method	Avg. Score	
	Rate	Rank
orderBERT	**3.86**	**2.43**
Original Position	3.43	1.86
Random	3.07	1.71

Table 7: Coherence Rating and Ranking Score

all, orderBERT predicts the next sentence regarding the content of the reference sentence so that the flow of the whole content is consistent. The method takes into consideration the connectivity that the anchor sentence can transfer to the next sentence.

6 Conclusion and Future Work

In our research, we assume different authors write who write news articles for different outlets, are very likely to utilize a variety of words. With this intuition, the synonyms are adapted to connected sentences among multi-documents. Results showed that identifying synonyms shared between sentences can successfully help to capture the content both within and across documents. In addition to coverage, we were also able to reduce redundancy through hierarchical clustering and improve coherence of the final summary using the proposed orderBERT. Moreover, although our entire framework are fully unsupervised, we are able to achieve comparable result than the supervised methods. In future work, we would like to combined supervised objective in our algorithm and focus on the compression rate which is also another challenging task for multi-document summarization.

Acknowledgments

We would like to thank members of the IDEA Lab and the anonymous reviewers for the helpful feedback. This research is supported in part by Ministry of Science and Technology in Taiwan (Program No. MOST 110-2221-E-007-085-MY3, MOST 108-2221-E-007-064-MY3).

References

Jaime Carbonell and Jade Goldstein. 1998. The use of mmr, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 335–336.

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

Anirban Dasgupta, Ravi Kumar, and Sujith Ravi. 2013. Summarization through submodularity and dispersion. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1014–1022.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1*, pages 4171–4186.

Günes Erkan and Dragomir R Radev. 2004. Lexrank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Alexander R Fabbri, Irene Li, Tianwei She, Suyi Li, and Dragomir R Radev. 2019. Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model. *arXiv preprint arXiv:1906.01749*.

Elena Filatova and Vasileios Hatzivassiloglou. 2004. Event-based extractive summarization.

Sebastian Gehrmann, Yuntian Deng, and Alexander M Rush. 2018. Bottom-up abstractive summarization. *arXiv preprint arXiv:1808.10792*.

Aria Haghighi and Lucy Vanderwende. 2009. Exploring content models for multi-document summarization. In *Proceedings of human language technologies: The 2009 annual conference of the North American Chapter of the Association for Computational Linguistics*, pages 362–370.

Leonhard Hennig, Winfried Umbrath, and Robert Wetzker. 2008. An ontology-based approach to text summarization. In *2008 IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*, volume 3, pages 291–294. IEEE.

Wojciech Kryściński, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. *arXiv preprint arXiv:1908.08960*.

Logan Lebanoff, Kaiqiang Song, and Fei Liu. 2018. Adapting the neural encoder-decoder framework from single to multi-document summarization. *arXiv preprint arXiv:1808.06218*.

Chin-Yew Lin and Eduard Hovy. 2002. Automated multi-document summarization in neats. In *Proceedings of the Human Language Technology Conference (HLT2002)*, pages 23–27. San Diego, CA, USA.

Yogesh Kumar Meena and Dinesh Gopalani. 2014. Analysis of sentence scoring methods for extractive automatic text summarization. In *Proceedings of the 2014 international conference on information and communication technology for competitive strategies*, pages 1–6.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 404–411.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Ani Nenkova and Lucy Vanderwende. 2005. The impact of frequency on summarization. *Microsoft Research, Redmond, Washington, Tech. Rep. MSR-TR-2005*, 101.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Dragomir R Radev, Timothy Allison, Sasha Blair-Goldensohn, John Blitzer, Arda Celebi, Stanko Dimitrov, Elliott Drabek, Ali Hakim, Wai Lam, Danyu Liu, et al. 2004a. Mead-a platform for multidocument multilingual text summarization.

Dragomir R Radev, Hongyan Jing, Małgorzata Styś, and Daniel Tam. 2004b. Centroid-based summarization of multiple documents. *Information Processing & Management*, 40(6):919–938.

Nedunchelian Ramanujam and Manivannan Kaliappan. 2016. An automatic multidocument text summarization approach based on naive bayesian classifier using timestamp strategy. *The Scientific World Journal*, 2016.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Barry Schiffman, Ani Nenkova, and Kathleen McKeown. 2002. Experiments in multidocument summarization.

Natalie Schluter and Anders Søgaard. 2015. Unsupervised extractive summarization via coverage maximization with syntactic and semantic concepts. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 840–844.

Chao Shen and Tao Li. 2010. Multi-document summarization via the minimum dominating set. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 984–992.

Hiroya Takamura and Manabu Okumura. 2009. Text summarization model based on maximum coverage problem and its variant. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 781–789.