Journal of Computer Science and Cybernetics, V.40, N.3 (2024), 203–217
DOI no. 10.15625/1813-9663/18353

GRAPH-BASED AND GENERATIVE APPROACHES TO MULTI-DOCUMENT SUMMARIZATION

TAM DOAN THANH[1,2], TAN MINH NGUYEN[2], THAI BINH NGUYEN[2], HOANG TRUNG NGUYEN[2], HAI LONG NGUYEN[2], MAI VU TRAN[2,*], QUANG THUY HA[2], HA THANH NGUYEN[3]

[1] *Viettel Group, Lane 7, Ton That Thuyet Street, Yen Hoa Ward, Cau Giay District, Ha Noi, Viet Nam*
[2] *VNU University of Engineering and Technology, E3 Building, 144 Xuan Thuy Street, Cau Giay District, Ha Noi, Viet Nam*
[3] *National Institute of Informatics, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo, Japan*

Abstract. Multi-document summarization is a challenging problem in the Natural Language Processing field that has drawn a lot of interest from the research community. In this paper, we propose a two-phase pipeline to tackle the Vietnamese abstractive multi-document summarization task. The initial phase of the pipeline involves an extractive summarization stage including two different systems. The first system employs a hybrid model based on the TextRank algorithm and a text correlation consideration mechanism. The second system is a modified version of SummPip - an unsupervised graph-based method for multi-document summarization. The second phase of the pipeline is abstractive summarization models. Particularly, generative models are applied to produce abstractive summaries from previous phase outputs. The proposed method achieves competitive results as we surpassed many strong research teams to finish the first rank in the AbMusu task - Vietnamese abstractive multi-document summarization, organized in the VLSP 2022 workshop.

Keywords. Multi-document summarization, abstractive summarization, NLP, graph-based, generative models.

1. INTRODUCTION

Text summarization plays a crucial role in processing and extracting valuable information from large volumes of data, particularly in the context of big data. It provides an efficient means for users, including non-expert individuals, to quickly grasp the key ideas and main points within a document or a collection of documents [30]. Summarization also supports other tasks such as information retrieval [9, 26, 36] and question answering [16, 35, 40]. The exponential growth of online content has created a challenge for users in locating relevant and valuable information amidst the vast amounts of available data. In the Vietnamese

*Corresponding author.
E-mail addresses: tamdt9@viettel.com.vn (T.D. Thanh); 20020081@vnu.edu.vn (T.M. Nguyen); 20020328@vnu.edu.vn (T.B. Nguyen); 20020083@vnu.edu.vn (H.T. Nguyen); long.nh@vnu.edu.vn (H.L. Nguyen); vutm@vnu.edu.vn (M.V. Tran); thuyhq@vnu.edu.vn (Q.T. Ha); nguyenhathanh@nii.ac.jp (H.T. Nguyen).

field, the shared task Vietnamese Abstractive Multi-document summarization (AbMusu) is the first time established in The 9^{th} International Workshop on Vietnamese Language and Speech Processing (VLSP 2022) [17].

Multi-document summarization is a complex task that involves aggregating information and producing a concise summary from a set of documents. The challenge lies in creating summaries that contain crucial information from the entire document collection while maintaining a coherent interpretation. More specifically, this paper tackles the text summarization task from a set of Vietnamese newspapers. Multi-document summarization has witnessed remarkable progress in recent years, driven by the strengths of deep learning models. Extractive models extract relevant sentences from input documents and combine them to form a summary, while abstractive models generate summary sentences from scratch. Although abstractive models exhibit superior performance compared to extractive models, they pose challenges in training due to the need for a comprehensive understanding of input document content and the generation of coherent summary sentences. Available pre-trained language models in Vietnamese [29, 38] allow us to apply these models to summarize Vietnamese text.

This paper introduces a two-phase pipeline for tackling the Vietnamese abstractive multi-document summarization. The first phase involves an extractive summarization stage including two different systems: a graph-based text correlation model and a modified version of SummPip, which involves four main steps: processing text, constructing sentence graphs, applying graph clustering, and generating a summary from the extracted sub-graphs. Since SummPip was originally designed for the English dataset, we proposed changes in the second and final steps to apply it to the Vietnamese dataset. The second phase fine-tunes pre-trained generation models to produce novel words and terms in summaries. The proposed methods achieved competitive results as we surpassed many strong research teams to rank first place in the Vietnamese abstractive multi-document summarization share task, organized in the VLSP 2022 workshop.

2. RELATED WORKS

There are two common approaches in text summarization: extractive summary and abstractive summary. Extractive summary aims to determine salient terms or phrases in the original text, using statistical and semantic features [21]. Meanwhile, abstractive summary requires advanced natural language models (e.g. Seq2Seq model [28], graph-based approach [37]) to understand and rewrite the input into a concise, fluent, human-like form [7].

2.1. Extractive summarization

Extractive summarization has been an active research area since its initial emergence in the 1950s [21]. Over the years, numerous techniques and approaches have been developed to address the challenges and improve the effectiveness of extractive summarization systems. Early methods in extractive summarization often relied on heuristics and rule-based approaches. These techniques involved identifying important sentences based on features such as sentence position, length, and keyword frequency [42]. Celikyilmaz et al. [5] proposed a two-step learning system that involves i) scoring sentences based on a hierarchical topic model and ii) training a regression model based on these scores to produce summaries for new documents. *GRAPHSUM* [1] proposed a novel summarizer based on a knowledge graph and association rules to define correlation among words and terms. With the explosion of deep learning, experiments are carried out to validate the performance of deep learning models in extractive summarization [10, 22].

Some works have considered this task as a sentence classification problem and handled it using pre-trained language models [14, 19]. Another approach to extractive summarization is GNNs, where these graphs can present complex semantic relations among terms and sentences [43, 44]. Wang et al. [41] proposed a *"heterogeneous graph-based neural network"* to improve the learning of cross-sentence relations among documents. *AREDSUM* [2] employed adaptive redundancy-aware to combine *salience* and *redundancy* for ranking sentences. Authors in [4] proposed an extractive summarization system based on various features and state-of-the-art techniques, combined with novel *PtN* strategies to enhance performance.

2.2. Abstractive summarization

Abstractive summarization generates the summarized information in a coherent form that is simple to read and grammatically correct. There are two main approaches for abstractive summarization: structure-based and semantic-based [20]. In the structure-based approach, key information from the document(s) is derived using features like templates, extraction rules, trees, ontology, lead and body, and graph-based structures. In the semantic-based technique, the Natural Language Generator (NLG) generates a summary of important information in the document.

Neural network sequence models offer a new approach to handling abstractive summarization. Rush et al. [33] presented the very first data-driven system based on a neural attention model for abstraction summarization. Summarizing long documents remains challenging as models are required to understand the topic and produce summaries with minimal amounts of duplication. See et al. [34] proposed a novel architecture based on a hybrid pointer generator to produce new words and *coverage* to reduce repetition. Another work [28] proposed a novel intra-attention for neural networks combined with a new training method based on reinforcement learning.

The success of transformer-based models in recent years has been remarkable, particularly in natural language processing. These models, which leverage attention mechanisms [39], have proven to be highly effective in capturing and incorporating more abstract information from documents [12, 31, 45].

Apart from these approaches, there are novel developed methods for abstraction summarization. The authors of [8] proposed a multi-granularity interaction network MGSUM based on the Transformer model for capturing the semantic relationships. *StructSum* proposed improved encoder-decoder models based on rich structure-aware document representations. The authors in [11] presented an effective technique to improve abstractive summaries by pre-training language models with the BRIO training diagram [15].

3. METHODOLOLY

The Vietnamese abstractive multi-document summarization (AbMusu) task is to generate a concise and abstractive summary S from a given set of topic-related documents $D = \{d_1, d_2, \ldots, d_n\}$. Summarization models are trained following the supervised learning approach, aiming to maximize the similarity between S and human-generated summary R. To tackle the AbMusu challenge, we consider both extractive and abstractive approaches involving graph-based and generative models. The extractive summarization phase aims to determine salient sentences or phrases from the original documents and form them into a summary. In the next phase, we employ generative models to rewrite inputs and produce concise, fluent, and human-like text.

3.1. Extractive summarization

3.1.1. Single document extraction

A document $d \in D$ is divided into three parts: title, anchor text $A = \{a_1, a_2, \dots, a_k\}$, and body text $B = \{b_1, b_2, \dots, b_l\}$. The anchor text serves as a summary to provide an overview to readers. Therefore, the correlation between anchor text A and body text B contributes to the summary's quality. This step utilizes these correlation features to determine the most important sentences that contain the overall idea of the document.

Particularly, to estimate an importance score of a sentence b, we use the Cosine similarity score as follows

$$Score(b) = \sum_{j=1}^{k} Sim(b, a_j), \tag{1}$$

$$Sim(b, a) = \frac{v_{b_i} \cdot v_a}{||v_b|| \cdot ||v_a||}, \tag{2}$$

where v_b and v_a represent the vector presentation of sentences b and a. Before applying similarity computation, we need a model that can derive semantically meaningful sentence embeddings. Therefore, we employ SentBERT [32] loaded with pre-trained parameters from PhoBERT [23] to produce sentence embeddings.

Sentences with the highest relevance score are selected to combine with the anchor text part to form a summary of the document d. This process is repeated for the other documents, we receive the first summary S_1 of the cluster D.

3.1.2. Multi-document extraction

After applying the initial step, Single document extraction, we obtain an extractive summary called S_1, which is generated by selecting and combining sentences from both the anchor text and the body text parts of documents. This process may introduce noise and duplicated information into the summary. To address this issue and improve the summary's quality, we employ TextRank [18], a graph-based sentence ranking method.

TextRank is an algorithm based on graph centrality that identifies important sentences in a document by considering their co-occurrence patterns. Let $\mathcal{G}(\mathcal{V}, \mathcal{E})$ be an undirected graph with the set of vertices $\mathcal{V}$, where each vertex is a sentence and a set of edges $\mathcal{E}$. Given a summary $S_1 = \{s_1, s_2, \dots, s_n\}$, the similarity between s_i and s_j is defined as their Cosine similarity score denoted in Equation 3. Edge weights are defined by the similarity score between two nodes it connects

$$w(i, j) = Sim(s_i, s_j). \tag{3}$$

The multi-document extraction step aims to refine the summary S_1 by reducing duplicated sentences and improving the summary's quality. The resulting output denoted as S_2, provides an overview content of the entire cluster D.

3.1.3. A modified version of SummPip

SummPip, proposed by Zhao et al. [46], is an unsupervised method for multi-document summarization that leverages sentence graph compression. The authors evaluate the performance of SummPip using ROUGE metrics on the Multi-News dataset [6] and the DUC-2004 dataset [27], demonstrating its effectiveness on both datasets.

The SummPip pipeline consists of four major steps. First, the text is processed. Then, a sentence graph is constructed, where each node represents a sentence from the processed text, and edges are defined based on both lexical and deep semantic relations between sentences. Next, graph clustering is applied to identify communities of nodes. Finally, a summary is generated based on the extracted sub-graphs. The input to SummPip is the raw dataset provided by the organizer, and the output is an extractive summary. Figure 1 illustrates the workflow of SummPip for unsupervised multi-document summarization.

To adapt SummPip for the Vietnamese dataset, we adjust the second and fourth steps. These modifications are necessary to accommodate the linguistic characteristics and unique features of the Vietnamese language while still leveraging the linguistic knowledge and deep neural representations utilized in the original method.

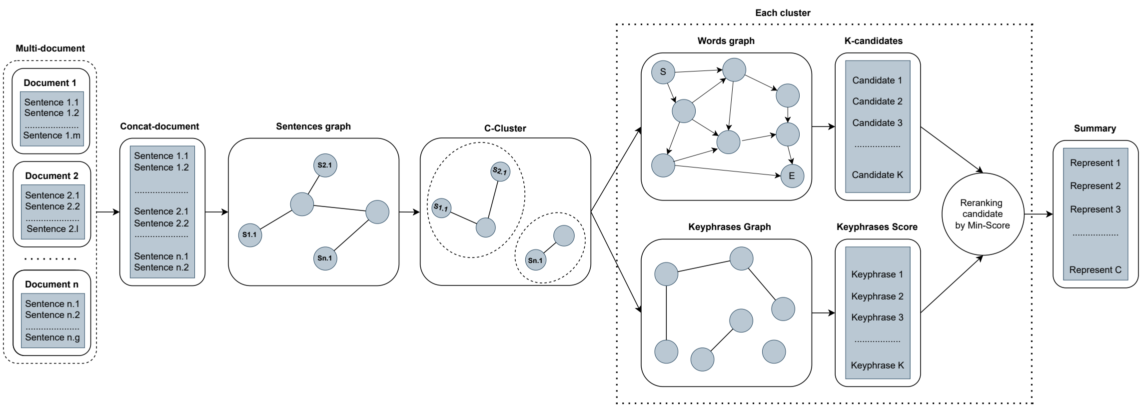

Figure 1: Automatic pipeline for unsupervised multi-document summarization

Text processing: Given a cluster of documents $D = \{d_1, d_2, \ldots, d_n\}$, we convert D into a list of sentences by concatenating all documents in D. Afterward, we perform word segmentation using RDRsegmenter [24] to form compound words. The output of this step is a list of segmented sentences, which serves as the input for the subsequent sentence graph construction phase.

Sentence graph construction: We build a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where each node $v_i \in \mathcal{V}$ represents a sentence. Compared to the original version, the linking conditions are changed based on Vietnamese linguistic knowledge and tools. Particularly, two nodes are connected if one of the following conditions is met:

- Discourse Markers: Two consecutive sentences v_i and v_{i+1} are connected if v_{i+1} contains discourse markers (e.g., tuy nhiên, vì vậy).
- Entity continuation: two sentences are connected if they contain the same proper nouns. We use the VnCoreNLP library to perform POS tagging [25] to determine word type.

- Sentence similarity: Instead of using word2vec as in the original version of SummPip, we utilized SentBERT with the initial parameters of PhoBERT to produce embedding vectors of Vietnamese sentences. These vectors can present contextual information from sentences. Finally, the Cosine score is applied to these vectors to measure the sentence similarity score.

Multi-sentence Compression for Summary: Mutli-sentence Compression is applied in the last step to generate summaries from the sentence clusters. SummPip uses an extended version of multi-sentence compression [3] by considering key phrases to adjust the compression process. The keyphrases extraction approach in Multi-Sentence Compression (MSC) can be divided into two steps: sentence compression and re-ranking paths using keyphrases.

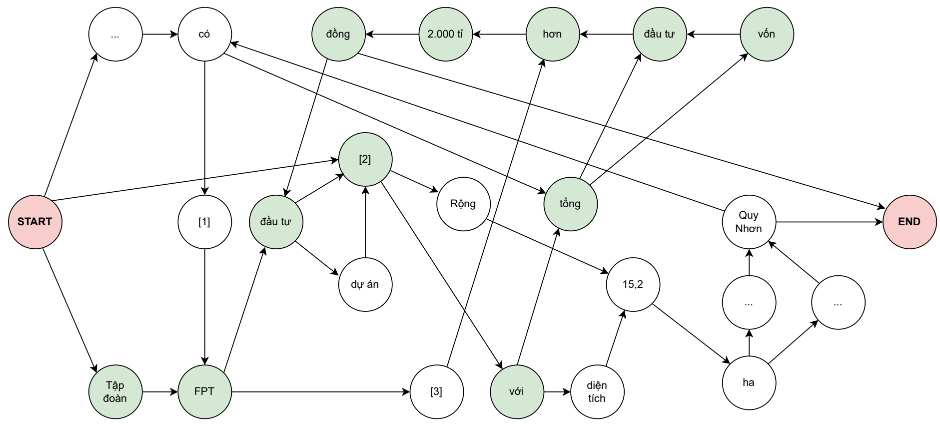

Figure 2: Example of word graph construction. A possible compressed path is also given

For the sentence compression step, a word graph is constructed to determine the shortest path as the new compressed sentence. Figure 2 depicts the word graph generated from the given sentences. For clarity, edge weights are omitted and some phrases in sentences are compressed in the graph due to space limitations. Particularly, node [1] stands for "quyết_định chấp_thuận cho Công_ty TNHH Phần_mềm FPT", node [2] is "Tổ_hợp Trung_tâm nghiên_cứu, sản_xuất và đào_tạo chuyên_gia công_nghệ FPT Software", and node [3] means "tiếp_tục đổ vốn".

- UBND tỉnh Bình_Định ngày 20-7 đã có quyết_định chấp_thuận cho Công_ty TNHH Phần_mềm FPT đầu_tư dự_án Tổ_hợp Trung_tâm nghiên_cứu, sản_xuất và đào_tạo chuyên_gia công_nghệ FPT Software với tổng đầu_tư hơn 2.000 tỉ đồng.

- Tập_đoàn FPT tiếp_tục đổ vốn trên 2.000 tỷ đồng đầu_tư Tổ_hợp Trung_tâm nghiên_cứu, sản_xuất và đào_tạo chuyên_gia công_nghệ FPT Software với diện_tích 15,2 ha trong khu quy_hoạch Khu đô_thị khoa_học và giáo_dục Quy_Hoà, TP Quy_Nhơn.

- Tổ_hợp Trung_tâm nghiên_cứu, sản_xuất và đào_tạo chuyên_gia công_nghệ FPT Software rộng 15,25 ha thuộc khu_vực 2, phường Ghềnh_Ráng, TP Quy_Nhơn, tỉnh Bình_Định có tổng vốn đầu_tư hơn 2.000 tỷ đồng.

The next step is to re-rank the retrieved paths using keyphrases extracted from the set of related sentences. We replace the original pattern for English with the regular expression $(\wedge(N|Np)*(A)*)$ to determine the keyphrase for Vietnamese. In this regular expression, $\wedge$ asserts position at the start of a line, $*$ means zero or more instances of the preceding regex token, A are adjectives, and Np is proper nouns, N are nouns. For example, "Bộ trưởng Bộ Giao thông Anh quốc Grant Shapps" would be considered as a keyphrase of a sentence.

3.2. Abstractive summarization

To address the issues of the incoherence of the extractive summary, we introduce an additional abstractive summarization phase utilizing the BARTpho and ViT5 models. While extractive summarization involves selecting and arranging topic-related sentences from the source documents, it may lack the ability to generate fluent and coherent summaries. Abstractive summarization, on the other hand, aims to generate summaries by paraphrasing and rewriting the content in a more human-like language.

BARTPho has a standard sequence-to-sequence transformer architecture and can be fine-tuned to tackle various downstream NLP tasks [38]. We fine-tune the BARTpho$_{word}$ version combined with word segmentation from the VnCoreNLP library to leverage Vietnamese linguistic features. The setting of min-len, max-len, and beam size are determined based on the validation set.

Introduced shortly after BARTPho, ViT5 [29] achieved state-of-the-art results on the Vietnamese summarization task. In this work, we fine-tune the ViT5-based model which contains 310M parameters on the training set made of extractive summaries and their corresponding gold labels. The settings of the text generation process are optimized using the validation set, which involves 100 clusters of documents.

3.3. Combining graph-based and generative models

One of the main challenges of multi-document summarization is the length of documents. An abstractive summarization model cannot aggregate information from a set of documents. Thus, we applied two-phase extractive summarization to compress data length while remaining topic-related information.

In particular, Figure 3 shows the overall pipeline's architecture. We first extract single documents to shorten the summary length. In the second phase, we concatenate the candidate sentences from single documents. Then we apply multi-document extraction to produce quickview summaries. The multi-document extraction has a great result because the gold summaries are quite similar to sentences in documents. Finally, we apply generative models for quickview summaries to produce abstractive summaries.

4. EXPERIMENTS AND RESULTS

4.1. Metrics

The official evaluation metrics used in the competition are the ROUGE-2 scores [13] and the ROUGE-2 F1 score. The ROUGE-2 F1 score is computed as follows

$$ROUGE - 2\ P = \frac{|Matched\ N - grams|}{|Predict\ summary\ N - grams|}, \tag{4}$$

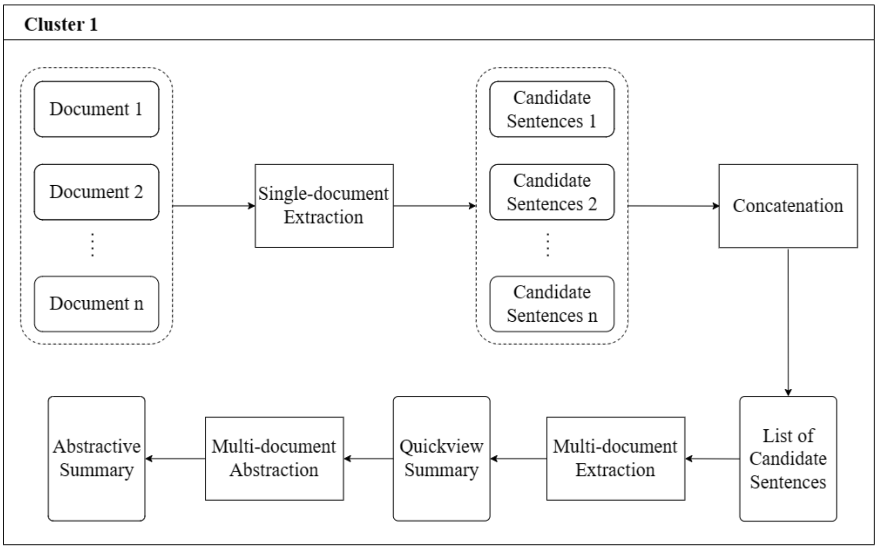

Figure 3: Hybrid architecture for abstractive multi-document summarization

$$ROUGE - 2\ R = \frac{|Matched\ N - grams|}{|Reference\ summary\ N - grams|}, \tag{5}$$

$$ROUGE - 2\ F1 = \frac{2 \times ROUGE - 2\ P \times ROUGE - 2\ R}{ROUGE - 2\ P + ROUGE - 2\ R}. \tag{6}$$

4.2. Data analysis

The provided dataset consists of Vietnamese news data covering various topics such as economy, society, culture, science, technology, etc. The data is divided into training, validation, and test sets. Each cluster in the dataset contains 3-5 documents that revolve around the same topic. Table 1 provides detailed statistics about the dataset. The document and summary lengths vary significantly, with the minimum and maximum document lengths being 73 and 4619 words, respectively. On average, the document length in the training and validation sets is approximately 700 words, while in the test set, it is around 667 words per document. The average document length is approximately three times greater than the average summary length. Figure 4 depicts the relationship between cluster length and summary length. The average cluster length is calculated by summing the lengths of all documents in a cluster and dividing by the number of documents. The figure suggests a positive correlation between cluster length and summary length. In Figure 5, the ratio between the cluster length and its corresponding summary length is 2.0 to 4.0. This feature determines the output length for each phase in the summarization process.

4.3. Hyper-parameters setup

Single document extraction. Given a cluster $D = \{d_1, d_2, \dots, d_n\}$ consists of the anchor text $A = \{a_1, a_2, \dots, a_k\}$, top 3 sentences with the highest score in each document are selected in the single document extraction phase. As a result, the output contains $3 \times n + k$ sentences.

Table 1: Statistics analysis of the dataset

	Training	Validation	Test
Number of clusters	200	100	300
Average number of documents	3.105	3.04	3.05
Maximum words per document	3474	4619	4291
Minimum words per document	148	132	73
Average words per document	715.32	701.65	667.13
Maximum words per summary	446	380	-
Minimum words per summary	63	83	-
Average words per summary	197.74	202.35	-

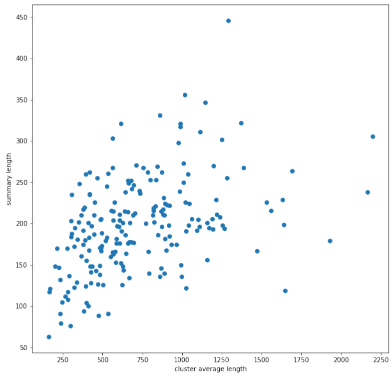

Figure 4: Relation between cluster and label length.

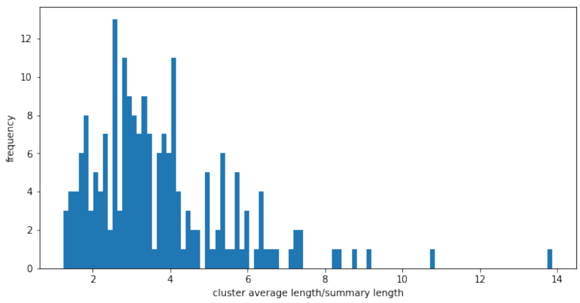

Figure 5: Ratio between cluster and summary length.

Multi-document extraction. After multiple experiments with different settings, we choose the top 5 candidate sentences from documents.

SummPip. We made some changes compared to the original SummPip. In particular, we use POS tags C and Cc from VnCoreNLP to define discourse markers, in which C is a subordinating conjunction and Cc denotes a coordinating conjunction. The threshold for sentence similarity is 0.83. The number of clusters in step three is equal to $3 \times N$, where N is the number of sentences of anchor texts.

BARTpho. The BARTpho model is fine-tuned using a dataset form of extractive summaries and their corresponding gold labels for 30 epochs with Adam optimizer. For the inference phase, the minimum and maximum output lengths are set as 0.7 and 1 of their inputs. As BARTpho is a generative model, generating a summary takes around 1-2 minutes for each input. Given the size of the testing dataset, it took approximately 6-7 hours to generate 300 summaries.

ViT5. The fine-tuning dataset of ViT5 contains pairs of extractive summaries and their corresponding gold labels. The fine-tuning process is carried out over 5 epochs using the Adam optimizer. During the inference process, we set the minimum and maximum target lengths for the abstractive summaries to be 0.7 and 1 of the size of their corresponding input texts, respectively. During each iteration, generating an abstractive summary takes approximately 2-3 minutes, and in total, it requires around 2 hours to produce 100 abstractive summaries for the validation dataset.

4.4. Results and discussion

The test data set has 100 document clusters. The public leaderboard is calculated with approximately 50% of the testing data. The private results are based on the other 50%. Table 2 shows our results on the private leaderboard. Multi-document extraction method achieves the best ROUGE-2 F1 and Recall scores, but the abstractive method using the BARTpho model has the highest Precision score because the extractive output is longer than the abstractive one. We used generative models to rewrite extractive summaries, so the final summaries are shorter, and more coherent than the extractive method. Table 4 shows an example of methods outputs. The abstractive summary has 50 words fewer than the multi-document extractive output. Table 3 shows the top 5 teams' final results. Our team achieves the first rank in the competition with the multi-document extractive method.

Table 2: Private result of the test set

	R2-F1	R2-Precision	R2-Recall
Extractive methods			
Multi-document extraction	**0.3035**	0.2298	**0.4969**
Single document extraction	0.2498	0.2093	0.3378
Modified SummPip	0.1993	0.1642	0.2849
Abstractive methods			
BARTpho-word-base	0.2512	**0.2429**	0.2848
ViT5-base	0.1693	0.1806	0.1556

Table 3: Private leaderboard in the competition

Team	R2-F1	R2-Precision	R2-Recall
minhnt_2709	**0.3035**	0.2298	**0.4969**
thecoach_team	0.2937	0.2284	0.4463
ngtiendong	0.2805	0.2629	0.3192
TheFinalYear	0.2785	0.2272	0.4040
nhanv	0.2689	**0.2773**	0.2829

As the team that has achieved top results in the task, it is crucial to discuss potential biases in the evaluation process. Identifying and understanding these biases can help drive improvements in the task and ensure fair and comprehensive evaluation in the future. Firstly, the ROUGE-2 scores and ROUGE-2 F1 are the main scores for ranking the models. Therefore, the team that has the highest ROUGE-2 score is the winner. However, ROUGE-2 is not a good metric to evaluate the quality of summaries. The main reason is that it cannot measure the coherence of summaries. For example, the predicted summary N-grams can match the reference summary N-grams very well, but the predicted summary can be completely incoherent. The ROUGE-2 scores will still be very high, but the content of the summary is meaningless. Secondly, this metric is based on the gold summary, but we cannot have gold summaries in the real world. People will have different opinions about what should be included in the summary.

Further analysis of the results reveals some challenges in our proposed method. Firstly, the method encounters difficulties in handling large document collections. This challenge

Table 4: Example of each module's output

	Single document extraction	Multi-document extraction	BARTpho	Modified SummPip
Inference Time	1 second	1 second	120 seconds	60 seconds
Words per summary	368	298	246	323
Generated summary	Trong khi đó, sàn giao dịch tiền mã hóa Coinbase cũng mất hơn 1,1 tỷ USD trong cùng khoảng thời gian, chủ yếu đến từ việc các đồng coin mất giá liên tiếp. Đồng thời, việc các công ty khai thác lớn chốt lãi những khoản vay là một điều đáng bận tâm. Song, Bitcoin vẫn mang lại lợi nhuận cho các hoạt động thương mại quy mô lớn, với mỗi khối được khai thác trị giá khoảng 6,25 BTC, hoặc khoảng 120.000 USD theo giá hiện nay. [...]	Theo một báo cáo từ Bloomberg, 3 công ty khai thác Bitcoin lớn đang niêm yết trên thị trường chứng khoán Mỹ gồm Core Scientific, Marathon Digital Holding và Riot Blockchain đã mất hơn 1 tỷ USD trong quý II/2022. Trong khi đó, sàn giao dịch tiền mã hóa Coinbase cũng mất hơn 1,1 tỷ USD trong cùng khoảng thời gian, chủ yếu đến từ việc các đồng coin mất giá liên tiếp. [...]	Giá tiền mã hoá hôm nay 1/9 : Giảm nhẹ trên các sàn giao dịch thế giới . Trong 24 giờ qua , giá Bitcoin giao dịch ở mức hơn 20.000 USD , giảm khoảng 1% trong 24 giờ qua . Ông Lionel Lim - CEO tại DBS Digital Exchange - bình luận rằng dự báo này là " khó xảy ra " [...]	nhà phân tích thị trường cao cấp của oanda , edward moya đã viết trong một email , lưu ý rằng nếu " đợt bán tháo cổ phiếu " tăng lên , bitcoin đã phải vật lộn để giữ ngưỡng quan trọng về mặt tâm lý này để níu chân các nhà đầu tư . trên sàn coindesk , lúc 17h15 ngày 318 , giá bitcoin giao dịch mức 20.188,47 usd , giảm 1,15% trong 24 giờ qua . " đây là một thành phần quan trọng của thị trường lao động sẽ giúp fed biện minh cho việc tăng lãi suất " , [...]
	Meanwhile, cryptocurrency exchange Coinbase also lost more than 1.1 billion USD in the same period, mainly due to the continuous devaluation of coins. At the same time, the fact that large mining companies are taking profits on loans is a concern. Yet Bitcoin is still profitable for large-scale commercial operations, with each mined block worth about 6.25 BTC, or about $120,000 at today's prices. [...]	According to a report from Bloomberg, three major Bitcoin mining companies listed on the US stock market, including Core Scientific, Marathon Digital Holding and Riot Blockchain, lost more than 1 billion USD in the second quarter of 2022. Meanwhile, cryptocurrency exchange Coinbase also lost more than 1.1 billion USD in the same period, mainly due to the continuous devaluation of coins. [...]	Cryptocurrency prices today September 1: Slight decrease on world exchanges. In the past 24 hours, Bitcoin price traded at over 20,000 USD, down about 1% in the past 24 hours. Mr. Lionel Lim - CEO at DBS Digital Exchange - commented that this forecast is "unlikely" [...]	Oanda senior market analyst Edward Moya wrote in an email, noting that if the "equity sell-off" intensified, bitcoin would have struggled to hold this psychologically important threshold to hold on. the investors . On coindesk, at 5:15 p.m. on day 318, bitcoin price was trading at 20,188.47 USD, down 1.15% in the past 24 hours. " this is an important component of the labor market that will help the fed justify raising interest rates " ,

arises due to the inherent slowness of generative models, making it impractical to apply them to extensive document sets efficiently. As indicated in Table 4, it takes nearly two minutes to generate a single abstractive summary. The second limitation pertains to the lack of consideration for sentence and document relationships in the proposed method. Since the primary goal in the competition is to achieve the highest ROUGE-2 score, the focus is primarily on generating summaries that match the gold labels. However, it is possible to incorporate sentence and document relationships by modifying the summary generation process. This aspect presents an interesting topic for further research, as it could improve the coherence and quality of the generated summaries.

5. CONCLUSION

In this paper, we propose a two-phase pipeline for the Vietnamese abstractive multi-document summarization task using both extractive and abstractive approaches. Particularly, the pipeline involves text correlation, TextRank algorithm, a modified version of SummPip, and generative models. We adjusted the SummPip model to apply it to the Vietnamese dataset. The second phase fine-tunes pre-trained generation models to produce novel words and terms in summaries. The proposed methods achieved competitive results as we surpassed many strong research teams to finish the first rank in the AbMusu share task - Vietnamese abstractive multi-document summarization, organized in the VLSP 2022 workshop. Future work would explore the latent relationship among sentences and documents to improve the quality of summaries.

ACKNOWLEDGMENT

Hai Long Nguyen was funded by the Master, PhD Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2023.ThS.075. Additionally, this research was partly supported by the AIP Challenge Funding in relation to JST, AIP Trilateral AI Research, under Grant Number JPMJCR20G4.

REFERENCES

[1] E. Baralis, L. Cagliero, N. Mahoto, and A. Fiori, "Graphsum: Discovering correlations among multiple terms for graph-based summarization," *Information Sciences*, vol. 249, pp. 96–109, 2013.

[2] K. Bi, R. Jha, W. B. Croft, and A. Celikyilmaz, "Aredsum: Adaptive redundancy-aware iterative sentence ranking for extractive document summarization," *arXiv preprint arXiv:2004.06176*, 2020.

[3] F. Boudin and E. Morin, "Keyphrase extraction for n-best reranking in multi-sentence compression," in *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2013.

[4] D. C. Can, Q. A. Nguyen, Q. H. Duong, M. Q. Nguyen, H. S. Nguyen, L. N. T. Ngoc, Q. T. Ha, and M. V. Tran, "Uetrice at mediqa 2021: A prosper-thy-neighbour extractive multi-document summarization model," in *Proceedings of The 20th Workshop on Biomedical Language Processing*, 2021, pp. 311–319.

[5] A. Celikyilmaz and D. Hakkani-Tur, "A hybrid hierarchical model for multi-document summarization," in *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, 2010, pp. 815–824.

[6] A. R. Fabbri, I. Li, T. She, S. Li, and D. R. Radev, "Multi-news: A large-scale multi-document summarization dataset and abstractive hierarchical model," *arXiv preprint arXiv:1906.01749*, 2019.

[7] S. Gupta and S. Gupta, "Abstractive summarization: An overview of the state of the art," *Expert Systems with Applications*, vol. 121, pp. 49–65, 2019.

[8] H. Jin, T. Wang, and X. Wan, "Multi-granularity interaction network for extractive and abstractive multi-document summarization," in *Proceedings of The 58th Annual Meeting of The Association for Computational Linguistics*, 2020, pp. 6244–6254.

[9] A. Kanapala, S. Jannu, and R. Pamula, "Passage-based text summarization for legal information retrieval," *Arabian Journal for Science and Engineering*, vol. 44, pp. 9159–9169, 2019.

[10] C. Kedzie, K. McKeown, and H. Daume III, "Content selection in deep learning models of summarization," *arXiv preprint arXiv:1810.12343*, 2018.

[11] K. N. Lam, T. G. Doan, K. T. Pham, and J. Kalita, "Abstractive text summarization using the brio training paradigm," *arXiv preprint arXiv:2305.13696*, 2023.

[12] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 7871–7880.

[13] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, 2004, pp. 74–81.

[14] Y. Liu and M. Lapata, "Text summarization with pretrained encoders," *arXiv preprint arXiv:1908.08345*, 2019.

[15] Y. Liu, P. Liu, D. Radev, and G. Neubig, "Brio: Bringing order to abstractive summarization," *arXiv preprint arXiv:2203.16804*, 2022.

[16] C. Lyu, L. Shang, Y. Graham, J. Foster, X. Jiang, and Q. Liu, "Improving unsupervised question answering via summarization-informed question generation," *arXiv preprint arXiv:2109.07954*, 2021.

[17] T. Mai Vu, L. Hoang Quynh, C. Duy Cat, and N. Quoc An, "VLSP 2022 – ABMUSU challenge: Vietnamese abstractive multi-document summarization," *Proceedings of The 9th International Workshop on Vietnamese Language and Speech Processing (VLSP 2022)*, 2022.

[18] R. Mihalcea and P. Tarau, "Textrank: Bringing order into text," in *Proceedings of The 2004 Conference on Empirical Methods in Natural Language Processing*, 2004, pp. 404–411.

[19] D. Miller, "Leveraging bert for extractive text summarization on lectures," *arXiv preprint arXiv:1906.04165*, 2019.

[20] N. Moratanch and S. Chitrakala, "A survey on abstractive text summarization," in *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*. IEEE, 2016, pp. 1–7.

[21] ——, "A survey on extractive text summarization," in *2017 International Conference on Computer, Communication and Signal Processing (ICCCSP)*. IEEE, 2017, pp. 1–6.

[22] R. Nallapati, F. Zhai, and B. Zhou, "Summarunner: A recurrent neural network based sequence model for extractive summarization of documents," in *Proceedings of The AAAI Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[23] D. Q. Nguyen and A. T. Nguyen, "PhoBERT: Pre-trained language models for Vietnamese," *arXiv preprint arXiv:2003.00744*, 2020.

[24] D. Q. Nguyen, D. Q. Nguyen, T. Vu, M. Dras, and M. Johnson, "A fast and accurate Vietnamese word segmenter," *arXiv preprint arXiv:1709.06307*, 2017.

[25] D. Q. Nguyen, T. Vu, D. Q. Nguyen, M. Dras, and M. Johnson, "From word segmentation to POS tagging for Vietnamese," in *Proceedings of the Australasian Language Technology Association Workshop 2017*, Brisbane, Australia, Dec. 2017, pp. 108–113. [Online]. Available: https://aclanthology.org/U17-1013

[26] H. T. Nguyen, M. P. Nguyen, T. H. Y. Vuong, M. Q. Bui, M. C. Nguyen, T. B. Dang, V. Tran, L. M. Nguyen, and K. Satoh, "Transformer-based approaches for legal text processing," *The Review of Socionetwork Strategies*, vol. 16, no. 1, pp. 135–155, 2022.

[27] P. Over and J. Yen, "An introduction to duc-2004," *National Institute of Standards and Technology*, 2004.

[28] R. Paulus, C. Xiong, and R. Socher, "A deep reinforced model for abstractive summarization," *arXiv preprint arXiv:1705.04304*, 2017.

[29] L. Phan, H. Tran, H. Nguyen, and T. H. Trinh, "ViT5: Pretrained text-to-text transformer for Vietnamese language generation," *arXiv preprint arXiv:2205.06457*, 2022.

[30] D. Radev, E. Hovy, and K. McKeown, "Introduction to the special issue on summarization," *Computational Linguistics*, vol. 28, no. 4, pp. 399–408, 2002.

[31] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, "Exploring the limits of transfer learning with a unified text-to-text transformer," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[32] N. Reimers and I. Gurevych, "Sentence-bert: Sentence embeddings using siamese bert-networks," *arXiv preprint arXiv:1908.10084*, 2019.

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

[34] A. See, P. J. Liu, and C. D. Manning, "Get to the point: Summarization with pointer-generator networks," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1073–1083.

[35] H. Song, Z. Ren, S. Liang, P. Li, J. Ma, and M. de Rijke, "Summarizing answers in non-factoid community question-answering," in *Proceedings of the Tenth ACM International Conference on Web Search and Data Mining*, ser. WSDM '17. New York, NY, USA: Association for Computing Machinery, 2017, p. 405–414. [Online]. Available: https://doi.org/10.1145/3018661.3018704

[36] T. Strzalkowski, J. Wang, and G. B. Wise, "Summarization-based query expansion in information retrieval," in *COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics*, 1998.

[37] J. Tan, X. Wan, and J. Xiao, "Abstractive document summarization with a graph-based attentional neural model," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, R. Barzilay and M.-Y. Kan, Eds. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 1171–1181. [Online]. Available: https://aclanthology.org/P17-1108

[38] N. L. Tran, D. M. Le, and D. Q. Nguyen, "Bartpho: Pre-trained sequence-to-sequence models for Vietnamese," *arXiv preprint arXiv:2109.09701*, 2021.

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[40] Y. T. H. Vuong, Q. M. Bui, H. T. Nguyen, T. T. T. Nguyen, V. Tran, X. H. Phan, K. Satoh, and L. M. Nguyen, "SM-BERT-CR: A deep learning approach for case law retrieval with supporting model," *Artificial Intelligence and Law*, pp. 1–28, 2022.

[41] D. Wang, P. Liu, Y. Zheng, X. Qiu, and X.-J. Huang, "Heterogeneous graph neural networks for extractive document summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 6209–6219.

[42] K. F. Wong, M. Wu, and W. Li, "Extractive summarization using supervised and semi-supervised learning," in *Proceedings of The 22nd International Conference on Computational Linguistics (Coling 2008)*, 2008, pp. 985–992.

[43] J. Xu, Z. Gan, Y. Cheng, and J. Liu, "Discourse-aware neural extractive text summarization," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, Eds. Online: Association for Computational Linguistics, Jul. 2020, pp. 5021–5031. [Online]. Available: https://aclanthology.org/2020.acl-main.451

[44] M. Yasunaga, R. Zhang, K. Meelu, A. Pareek, K. Srinivasan, and D. Radev, "Graph-based neural multi-document summarization," in *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, 2017, pp. 452–462.

[45] J. Zhang, Y. Zhao, M. Saleh, and P. Liu, "Pegasus: Pre-training with extracted gap-sentences for abstractive summarization," in *International Conference on Machine Learning.* PMLR, 2020, pp. 11 328–11 339.

[46] 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 *Proceedings of The 43rd International ACM Sigir Conference on Research and Development in Information Retrieval*, 2020, pp. 1949–1952.

Received on June 01, 2023
Accepted on April 01, 2024