VNU Journal of Science:
Computer Science and Communication Engineering
Journal homepage: http://www.jcsce.vnu.edu.vn/index.php/jcsce

Original Article

Enhancing Comparative Opinion Mining in Vietnamese Product Reviews: A Hybrid Generative Model Approach with Knowledge Base Integration

Thu-Trang Pham, Huu-Dong Nguyen, Dieu-Quynh Nguyen, Thi-Hai-Yen Vuong, Hoang-Quynh Le*

VNU University of Engineering and Technology, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam

Received 17 April 2024;
Revised 08 July 2024; Accepted 10 December 2024

Abstract: Opinion mining holds numerous practical applications, especially as user-generated sentiment data become increasingly prevalent in the digital age. Comparative Opinion Mining (ComOM), a specific sub-field of opinion mining, focuses on extracting the elements involved in the comparison in a text and retrieving the corresponding tuples expressing comparative opinions. This research describes an improved version of our system that participated in the VLSP 2023 ComOM shared task. The baseline system's performance was ranked second among 22 participating systems in the shared task through a combination of generative model and classification-based approaches, along with knowledge-based techniques with a 0.2300 F1-score. More experiments have been conducted to improve the performance of the system, reaching a final F1-macro score of 0.2391. This demonstrates the superiority of our proposed method compared to existing approaches in the task of comparative opinion mining.

Keywords: Comparative opinion mining, Comparative quintuples extraction, Graph Convolutional Networks, Generative models

1. Introduction

Reviews of products hold significant importance for both customers and businesses. These reviews can influence customers' purchasing decisions and businesses' reputations. With the exponential growth of social media and e-commerce platforms, the volume of product reviews has increased rapidly, emphasizing the need for efficient methods to extract valuable insights from this abundance of information. One of the most persuasive methods for evaluating

*Corresponding author.
E-mail address: lhquynh@vnu.edu.vn

https://doi.org/10.25073/2588-1086/vnucsce.2582

Table 1. Example of Quintuples extracted from the product review

Product Review Sentence
iPhone 14 được nâng cấp bộ nhớ lên đến 6GB RAM cao hơn iPhone 13 đến 2GB RAM, cho khả năng đa nhiệm tốt hơn.
English: *iPhone 14 has upgraded memory to 6GB of RAM, 2GB of RAM higher than iPhone 13, for better multitasking capabilities.*
Quintuple Constituents
Subject: iPhone 14
Object: iPhone 13
Aspect: {bộ nhớ, khả năng đa nhiệm} *(**English:** {memory, multitasking capabilities})*
Predicate: {cao hơn, tốt hơn} *(**English:** {higher, better})*
Comparison type: COM+
Results
{iPhone 14, iPhone 13, bộ nhớ, cao hơn, COM+}
{iPhone 14, iPhone 13, khả năng đa nhiệm, tốt hơn, COM+}
English:
{iPhone 14, iPhone 13, memory, higher, COM+}
{iPhone 14, iPhone 13, multitasking capabilities, better, COM+}

products involves direct comparisons with similar offerings or competitors. Extracting these comparisons from a lengthy product review can significantly save readers time. Consequently, the task of comparative opinion mining (ComOM) was first introduced in 2006 [1], which helps extract comparative relations among products within a product review. Developing an effective method for comparative opinion mining aims to benefit businesses and consumers alike in navigating the increasingly vast number of product reviews.

Opinion mining involves various subproblems such as classification, information extraction, and opinion summarization. This work focuses on identifying sentences for comparison and extracting five key elements from product reviews. The specific definition of the comparative opinion quintuple extraction problem is provided in the VLSP 2023 shared task [2]. Quintuples, which include subject, object, aspect, predicate, and comparison type, are extracted from review sentences. The subject represents the entity initiating the comparison, the object is the entity being compared, and the aspect is the feature or attribute under comparison. Each comparative sentence includes the user's viewpoint, referred to as the predicate, with the comparison type determined by this predicate. Table 1 provides an example of a comparative sentence and the quintuples that can be extracted from it. In this example, two quintuples can be derived from just one comparative sentence. During the development of this model, data posed a significant challenge. The quantity of data is limited because, in practice for the ComOM task, only a small portion of sentences in a review are comparative. Additionally, sentences often contain multiple tuples of comparative opinions, such as the example of Table 1, making it challenging to accurately identify and extract each comparison. There is also an imbalance among comparative labels. Certain types of comparison appear in much fewer samples compared to others, as in reality, consumers tend to post more positive reviews [3]. Another hurdle involves understanding the context surrounding comparisons since nuances in language and tone can greatly affect the interpretation of comparative statements. Moreover, the dynamic nature of language and the wide range of products and domains further complicate the task of developing robust and generalizable models for comparative opinion mining. Overall, addressing these challenges requires innovative approaches that can effectively handle sparse data, capture complex linguistic structures, and comprehend the contextual nuances inherent in product reviews.

To address these challenges, the data was pre-processed and augmented before the model development phase. Our main contributions are the hybrid generative model to extract comparative quintuples and the knowledge-based techniques for pre-processing and post-processing. Our baseline model is applied to the dataset from the VLSP 2023 ComOM shared task. The baseline achieved an F1-score of 0.1599 in the public test set and second place in the private leaderboard of the competition with a score of 0.2300. After going through further experiments, our improved model achieves an F1-score of 0.2391.

The rest of the paper is organized as follows. Previous works related to this study are presented in Section 2. Section 3 describes the proposed method while Section 4 describes the dataset and evaluation metrics that are used in the experiments. Section 5 provides the experimental results and discussion. The last section gives conclusions about the work and provides suggestions for future work.

2. Related Works

As a significant sub-field of Opinion Mining, ComOM has captured the interest of numerous researchers in the Natural Language Processing (NLP) field due to its high applicability [4]. With the explosion of user-generated reviews on the Internet and E-commerce platforms, the automated extraction of valuable information from abundant texts becomes increasingly significant. Various techniques have been employed to address this task, with a common approach involving the subdivision of this task into three distinct sub-tasks: 1) the recognition of comparative sentences, 2) the extraction of comparative constituents from the aforementioned sentences, and 3) the classification of comparative reviews into different polarity classes.

Comparative Sentence Identification (CSI). Since the first introduction of the ComOM task in [1], the authors proposed an approach combining class sequential rules (CSR) and Naïve Bayesian (NB) classification to perform this task. Their experiments show that CSR is useful in support of the final classification. After conducting experiments on the Naïve Bayesian model and Support Vector Machine (SVM) as classifiers using the CSR as features, the results show that the Naïve Bayesian model outperformed SVM with an overall precision of 79% and an overall recall of 81%. To capture the context-aware representation of a sentence, Can et al. in [5] combine the shortest dependency path (SDP) and attention model. Their proposed methods improve the competitive baselines on the SemEval-2010. More recently, Liu et al [6] introduced a multi-task learning BERT-based framework for the simultaneous identification of comparative sentences and extraction of comparative elements. Bon et al. [7] used a combination of rule-based, feature-based classification and neural classifier from BERT-variants to identify comparative questions. They ensemble the three classifiers with a cascading method. This method achieved 71% recall and 100% precision with their dataset. These recent studies show how superior BERT-based models are in recognizing comparative sentences task.

Comparative Elements Extraction (CEE). Arora et al. [8] tackled this task with a neural network approach, where they employed LSTMs to capture the relations of comparative subject, object, aspect, and predicate. Whereas in [9], instead of aiming to extract the subject and object of the comparison, the authors focused on extracting the Aspect-Category-Opinion-Sentiment Quadruple in a comparative sentence. They propsed four different systems, integrating various neural methods.

Polarity classification and comparative tuples extraction. Polarity classification aims to categorize comparative reviews into different polarity classes, typically positive, negative, or neutral. As the last element of a quintuple contains the type and polarity of a comparison, polarity classification techniques are necessary to tackle the comparative opinion quintuple extraction problem. Previously, Liu et al [6] proposed a multi-stage deep learning model approach and the results surpassed the baseline systems from previous methods in comparative opinion mining significantly, whereas Xu et al [10] proposed a BERT-based end-to-end neural model and developed a Graph Convolutional Network (GCN) to enhance that model. Their experimental results on three benchmarks show that the GCN-based model has a notable improvement in most instances when compared to the BERT-based pipeline baseline. The authors in [11] developed a unified generative model with a set-matching strategy to solve the COQE task. In 2015, Bach et al. [12] introduced a new corpus for the task in Vietnamese. They also proposed a general framework to tackle two sub-tasks: (1) identifying comparative sentences and (2) recognizing comparative relations. They achieved high results in each subtask with limited types of comparisons.

As observed in the work of Xu et al. [10] and Yang et al. [11], GCN-based and T5-based generative models demonstrated a sufficient performance at extracting comparative elements and polarity classes. However, these systems primarily target English and Chinese languages, limiting their applicability to Vietnamese data. To address this gap, this hybrid system that ensembles both approaches with knowledge base integration, employing fine-tuned models for Vietnamese to enhance its effectiveness in extracting opinions from Vietnamese reviews, is introduced in this study.

3. Methods

As introduced in [6], the COQE task can be formally stated as follows. Given a product review X, the goal of the task is to classify whether X is comparative and extract all corresponding comparative quintuples within it:

$$\begin{aligned} S_X &= \{tup_1, tup_2, \ldots, tup_k\} \\ &= \{(sub_1, obj_1, asp_1, pre_1, label_1), \ldots, \\ &\quad (sub_k, obj_k, asp_k, pre_k, label_k)\} \end{aligned}$$

where k is the number of comparative quintuples extracted from sentence X. Each quintuple $tup = (sub, obj, asp, pre, label)$ has five elements which are respectively equivalent to the subject and object being compared, the aspect or feature/attribute in comparison, the opinion of the writer and the comparison type label. In the VLSP shared-task, the *label* can be one of the following categories: ranked comparison (e.g., “better”, “worse”), superlative comparison (e.g., “best”, “worst”), equal comparison (e.g., “same as,” “as good as”), and non-gradable comparison (e.g., “different from,” “unlike”). These categories decompose into a total of eight types of comparison, denoting as {COM+, COM-, COM, SUP+, SUP-, SUP, EQL, DIF}. The first four elements of the quintuple can be

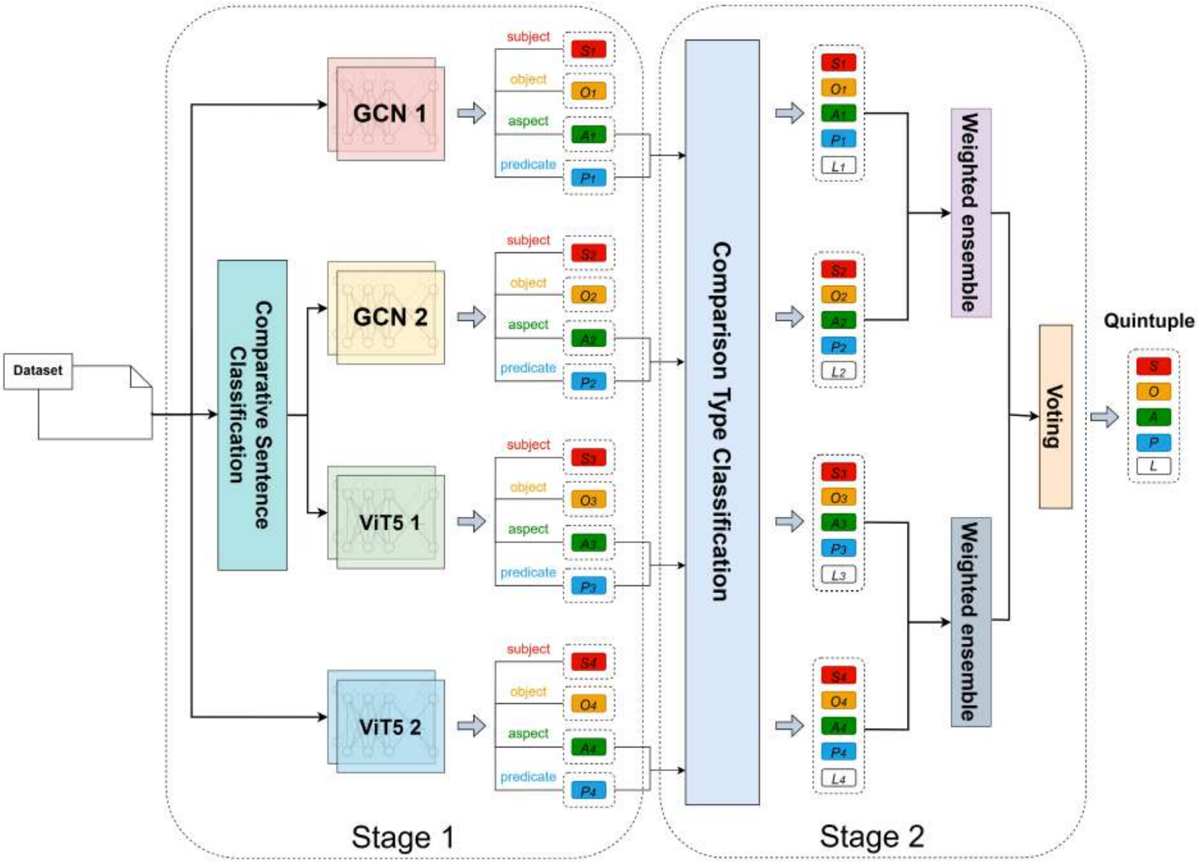


Figure 1. Overview of the end-to-end multistage COQE model.

extracted from the comparative sentence, whereas the *label* needs to be determined from the previous elements.

3.1. Overall Architecture

In this paper, a proposed method of ensembling classification models, an extraction model, and a generation model is explored. The following section specifies the details of this proposed method. Figure 1 demonstrates the multi-stage end-to-end architecture of the proposed method. There are 2 stages in this model.

In the First Stage, the comparative sentence classification is employed to identify all comparative sentences in the dataset. These sentences are then fed into the GCN-based extractors and ViT5 generation model to produce the quadruple of each sentence. The last element of the quintuple is obtained from the comparison type classification model in Stage 2.

In the Second Stage, which is the ensemble phase, the outputs of the GCN-based extractors and ViT5 generation are weighted-ensembled to get the best results of the two models. Voting-ensemble is then applied to the results, producing the final quintuples of the input sentence.

3.2. Stage 1: Quadruples Extraction

The objective of this stage is to determine whether the sentence contains comparisons with the comparative sentence classification. If the classification indicates a comparative nature, the next step involves generating a quadruple.

3.2.1. *Comparative Sentence Classification*

According to our data analysis, the number of comparative sentences makes up less than 20% of the corpus. Therefore, at the beginning of the first stage, a sentence classifier is employed to extract all comparative sentences, aiming to improve the performance of the generation model and the extraction model. Figure 2 visualizes how the classifier works and the next part describes the model in detail.

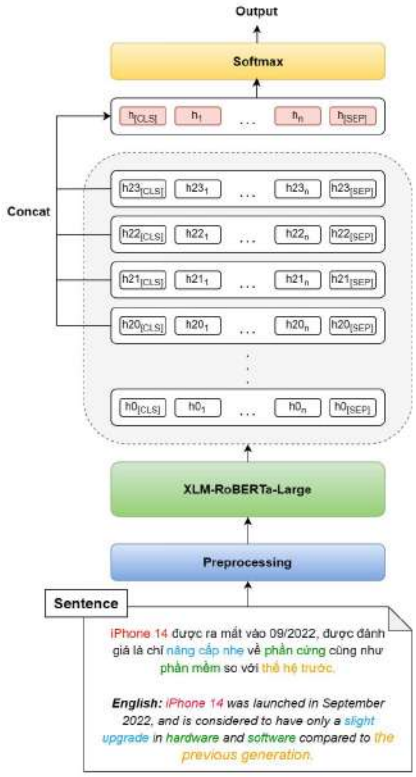

Output
Softmax
h[CLS] h1 ... hn h[SEP]
Concat
h23[CLS] h231 ... h23n h23[SEP]
h22[CLS] h221 ... h22n h22[SEP]
h21[CLS] h211 ... h21n h21[SEP]
h20[CLS] h201 ... h20n h20[SEP]
h0[CLS] h01 ... h0n h0[SEP]
XLM-RoBERTa-Large
Preprocessing
Sentence
iPhone 14 được ra mắt vào 09/2022, được đánh giá là chỉ nâng cấp nhẹ về phần cứng cũng như phần mềm so với thế hệ trước.
English: iPhone 14 was launched in September 2022, and is considered to have only a slight upgrade in hardware and software compared to the previous generation.

Figure 2. Architecture of classification model.

Given a sentence X, after being preprocessed, X is fed into a BERT-based model. In this work, XML-RoBERTa-Large [13] is utilized to produce the last four hidden representations.

$$h_i = [h_i^{[CLS]}, h_i^1, \dots, h_i^n, h_i^{[SEP]}], i \in \{20, 21, 22, 23\}$$

The last 4 hidden representations of the [CLS] token are then concatenated and put through a softmax layer to predict which class is the highest and whether the sentence is comparative.

$$h = h_{20}^{[CLS]} + h_{21}^{[CLS]} + h_{22}^{[CLS]} + h_{23}^{[CLS]} \quad (1)$$

$$y^c = Softmax(W^c h + b^c) \quad (2)$$

where W^c and b^c are weight matrices to learn, and $y^c \in \{0, 1\}$.

3.2.2. *Graph Convolutional Network (GCN)-based Extraction Model*

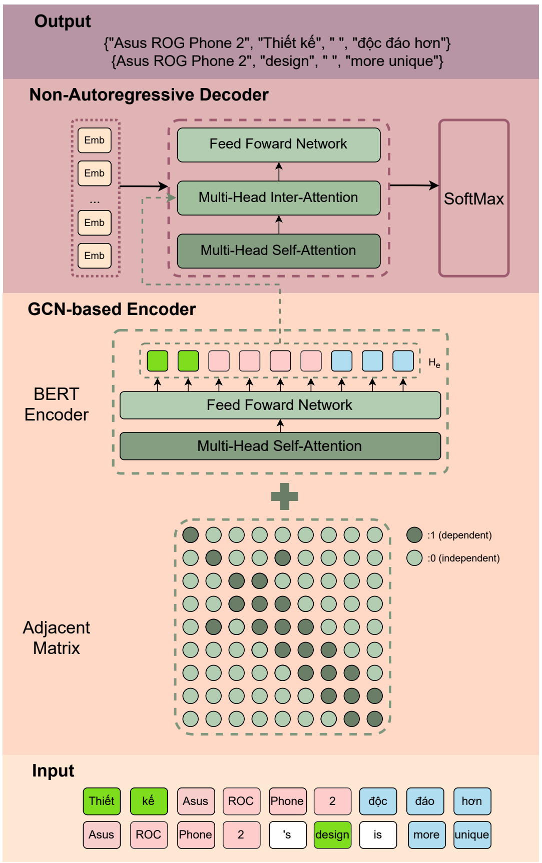

Output
{"Asus ROG Phone 2", "Thiết kế", " ", "độc đáo hơn"}
{Asus ROG Phone 2", "design", " ", "more unique"}
Non-Autoregressive Decoder
Emb
Emb
...
Emb
Emb
Feed Foward Network
Multi-Head Inter-Attention
Multi-Head Self-Attention
SoftMax
GCN-based Encoder
He
BERT Encoder
Feed Foward Network
Multi-Head Self-Attention
:1 (dependent)
:0 (independent)
Adjacent Matrix
Input
Thiết kế Asus ROG Phone 2 độc đáo hơn
Asus ROG Phone 2 's design is more unique

Figure 3. Architecture of GCN-based extraction model.

Graph Convolutional Network is a type of neural network architecture designed for processing graph-structured data [14]. The reason GCN is utilized in the proposed method is because it can enhance the encoder's ability to perceive adhesion among the quadruple components. Figure 3 shows the complete architecture of the GCN-based extraction model. Given a comparative sentence $X = \{x_1, x_2, \ldots, x_n\}$ consisting of N tokens, the context-aware attention-weighted representation for each token is obtained by a pre-trained BERT model [15].

$$H = \{h_1, h_2, \ldots, h_i, \ldots, h_N\} \tag{3}$$

Eq. 3 shows the hidden state H after the BERT encoding layer. The self-augmented representation of each token x_i in the encoding stage is denoted as h_i, where $h_i \in \mathbb{R}^d$.

In terms of the adjacency matrix, a $N \times N$ adjacency matrix A is built based on the parsed dependency relations among tokens of the input sentence, which is produced by employing the Underthesea toolkits. The value 1 in the adjacency matrix indicates connected tokens and 0 indicates otherwise. Since the graph is undirected, the produced adjacency matrix is symmetrical. The addition of A to the forward propagation step allows the GCN to learn the features better as it reflects the connectivity of the nodes. In this study, a single-layer GCN is utilized with the GCN presented below.

$$H^{(l)} = \sigma(AH^{(l-1)}W^{(l-1)}) \tag{4}$$

Eq. 4 presents the updated representation $H^{(l)}$ at layer l. A in the formula is the aforementioned adjacency matrix. $W^{(l-1)}$ and $H^{(l-1)}$ respectively refer to the trainable weight matrix and representation at the current GCN layer while σ denotes the nonlinear activation function (eg., ReLU).

There are two different approaches in the design of sequence-to-sequence models: Autoregressive and Non-autoregressive. These terms refer to how the output sequence is generated over time. In an autoregressive decoder, the generation of each output element is dependent on the previously generated elements. A non-autoregressive decoder, on the other hand, generates all output elements simultaneously or in parallel, without waiting for the generation of previous elements. Since the comparative constituents are sequentially disordered, a non-autoregressive decoder is employed in the proposed method to speed up the decoding process.

The representations of different sets of quintuple constituents, which consist of N 768-dimensional embeddings, are first randomly initialized. The number of embeddings is determined by getting the largest number of quintuples in all sentences in the training data. These embeddings are denoted as Q. Q and H are then fed into the transformer decoder layer where self-attention will be computed for Q, and further inter-attention is calculated between the self-attentive Q and H. In each decoder layer, the representation of Q is updated as follows:

$$Q^{(L)} = Decoder(H, Q^{(L-1)}) \tag{5}$$

The probabilities of classes of constituents are estimated using a softmax function.

$$p_i^e = Softmax(V_i^T tanh(W_e q_i^L + W_h H)) \tag{6}$$

In Eq. 6, W_e, W_h and V are all trainable parameters for the classes of elements p_i^e. q_i^L is the i-th embedding output by the final decoder layer. During the generation process, a series of special tokens can be produced, such as {start, end} which indicates the beginning and ending of an element.

3.2.3. T5-based Generative Model

Figure 4 demonstrates the overall structure of the generation model in the study and examples of the input and output of the model.

In the generative paradigm, k golden quadruples are concatenated with “ > ” as the target sequence of the model, padding missing comparison elements with "*unknown*".

For the input sentence X, during the training

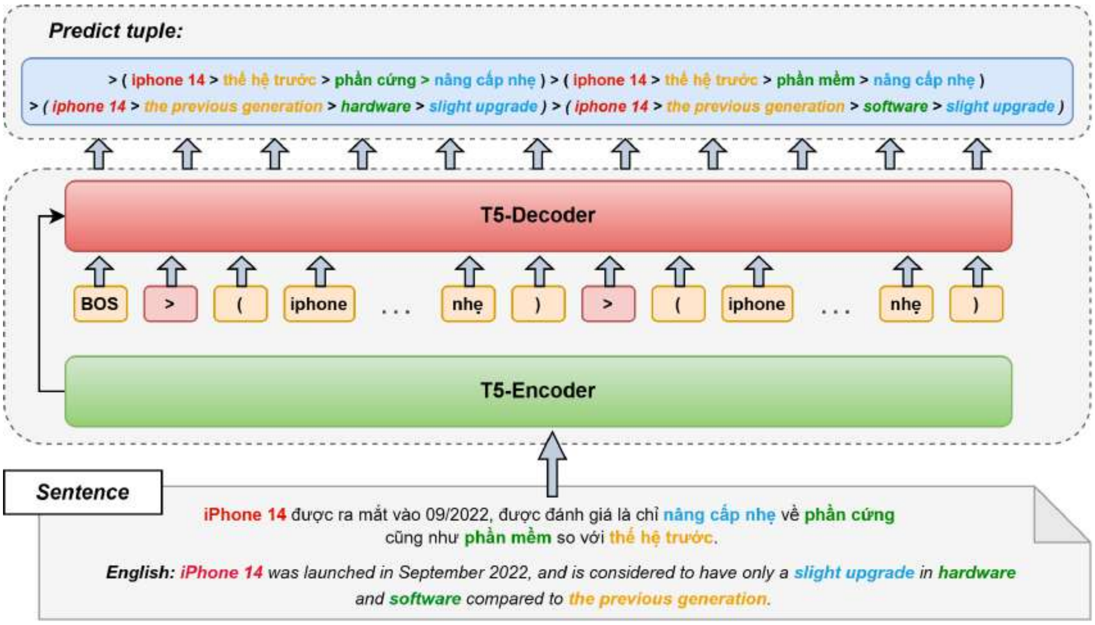

Predict tuple:
> (iphone 14 > thế hệ trước > phần cứng > nâng cấp nhẹ) > (iphone 14 > thế hệ trước > phần mềm > nâng cấp nhẹ)
> (iphone 14 > the previous generation > hardware > slight upgrade) > (iphone 14 > the previous generation > software > slight upgrade)
T5-Decoder
BOS
>
(
iphone
. . .
nhẹ
)
>
(
iphone
. . .
nhẹ
)
T5-Encoder
Sentence
iPhone 14 được ra mắt vào 09/2022, được đánh giá là chỉ nâng cấp nhẹ về phần cứng cũng như phần mềm so với thế hệ trước.
English: iPhone 14 was launched in September 2022, and is considered to have only a slight upgrade in hardware and software compared to the previous generation.

Figure 4. Architecture of generation model.

phase, the gradient backpropagation of the model is temporarily turned off. X is then fed into the T5-encoder to get the latent representation h_c^{dec} of the sentence.

$$h_c^{dec} = Encoder(X) \tag{7}$$

A T5-decoder is utilized to predict all the comparative quintuples autoregressively. At the $c_t h$ moment of the decoder, h^{enc} and the previous output tokens $t_{1:c-1}$ are formed as the input of the decoder:

$$h_c^{dec} = Decoder(h^{enc}, t_{1:c-1}) \tag{8}$$

The conditional probability of token t_c is defined as follows:

$$P(t_c|t_{1:c-1}, X) = Softmax(W_c h^{dec} + b_c) \tag{9}$$

where $W \in R^{d_h \times |v|}, b \in R^{|v|}$. v here refers to the vocabulary size of ViT5. Then the final predicted sequence of tuples is:

$$T_{pred} = t_{1:m} = \{t_1, \ldots, t_m\} \tag{10}$$

where m is the length of the predicted sequence. T_{pred} is split with the greater than symbol " > " to get the set of comparative quadruples predicted by the model

$$Q_{pred} = \{quad_1^{pred}, \ldots, quad_n^{pred}\} \tag{11}$$

with quadruple $quad_i = (sub_i, obj_i, asp_i, pre_i)$.

3.3. Stage 2: Quintuples Extraction

Following the generation of quadruples, the next step involves the classification of comparative labels. Subsequently, the models are ensemble together for each label based on the scores obtained from the GCN and ViT5 models to produce the final result.

3.3.1. Comparative Label Classification

A separate BERT-based classifier is proposed to specialize in identifying comparison type labels. Even though the extraction model and the generation model could extract and generate all elements of the comparative opinion quintuple, the classification model is proved to be superior because it is fine-tuned specifically for this task. This classification model has the same architecture as the Comparative sentence classification model, but instead of a full sentence, the input of this model is the aspect and predicate produced by the extraction and generation models. First, the aspect and predicate are combined into one sentence. Then a special prefix "hơn +, hơn -, hơn, nhất +, nhất -, nhất, bằng, khác", which can be translated as "com +, com -, com, sup +, sup -, sup, eql, dif", is added to the beginning of the sentence to form a complete input. The input is put through the model to determine which comparison type label has the highest probability being the label of the quintuple.

3.3.2. *Ensemble*

The ensemble phase begins with ensembling two GCN models. The thresholds θ are set based on the validation set. Quintuple constituents with score $score \geq \theta$ are retained. Then, we still apply a similar ensemble approach to two ViT5-large models. Finally, these outputs are ensembled via voting. It can be noticed that rare labels suffer from the lack of data, making it difficult for models to learn them. To address this issue, we employed a method of duplicating the quintuples of non-rare labels and then changing the comparison type component within it.

4. Data and Evaluation Metrics

This section provides information about the dataset and evaluation metrics used in this study.

4.1. *VLSP 2023 Challenge on ComOM from Vietnamese Product Reviews Dataset*

The primary benchmark dataset used in this study is the VCOM corpus provided by the VLSP 2023 challenge on Comparative Opinion Mining [2]. A labeled dataset consisting of product reviews in Vietnamese is split into training, development, and test sets.

Based on our data analysis on the training and development sets, approximately only 20% of sentences are comparative. Specifically, there are 4171 sentences in the training set, of which 812 sentences are comparative. Regards to the development set, there are 1733 sentences with a total of 349 comparative sentences.

In terms of sentence length distribution, the majority of sentences are less than 30 words long. The shortest is a one-word sentence and the longest sentence in the corpus has 82 words. The average sentence lengths in the training and development sets are 18.29 words and 19.01 words respectively. Especially, based on our analysis, we identified that sentences with less than 3 words are not comparative sentences and can therefore be eliminated.

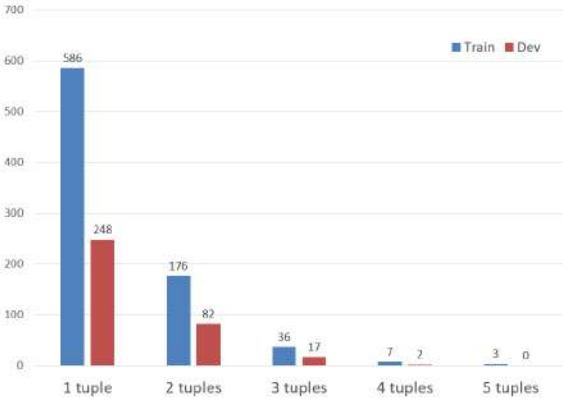


Figure 5. Number of quintuples of each sentence.

The number of quintuples for each sentence is imbalanced and varied between the training and development sets. Most comparative sentences have one tuple, and no sentence has more than five tuples. The detailed numbers of tuples are demonstrated in Figure 5.

The datasets show an imbalance in comparative labels as well, the distribution is consistent as shown in Figure 6. Quintuples with COM+ label account for 42% and 46% while SUP and SUP- only appear in 1% and 0.21% of sentences in the training and development sets respectively. This poses a challenge in handling

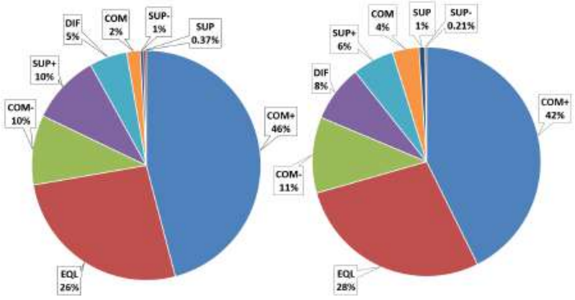


Figure 6. Distribution of labels.

biases against minority labels.

4.2. *Data Enrichment*

As the number of comparative sentences provided in the competition is limited, a method of paraphrasing is employed to enrich the data for our experiments. First, a pretrained MT5 [1] on Huggingface is used to produce paraphrases of all comparative sentences in the training set. The paraphrased sentences are then postprocessed using knowledge-based techniques. The final augmented set consists of 205 new sentences.

4.3. *Evaluation Metrics*

In this study, the performance of the proposed method is evaluated with different measures for extracting quadruples and quintuples. For the Comparative Element Extraction, Precision, Recall, and F1 score for each individual element (subject, object, aspect, predicate, and comparison type label), as well as their Micro-average F1 score will be used. For the quintuples evaluation, the evaluation metrics are Precision, Recall, and F1 score for the entire quintuple.

$$Precision = \frac{\#correct}{\#predict}$$

$$Recall = \frac{\#correct}{\#gold}$$

The number of predictions by the model is presented as *#predict* while *#gold* indicates the number of comparative elements for CEE and quintuples for COQE in the dataset. The *#correct* signifies the number of correct predictions.

There are three matching strategies to measure correct predictions: Exact Match (E) where the entire extracted quintuple component must match exactly with the ground truth, Proportional Match (P) where the proportion of matched words in the extracted component with respect to the ground truth will be considered, and Binary Match (B) where at least one word in the extracted component must overlap with the ground truth.

The main focus of our experiment is on E-T4-F1, E-T5-MACRO-F1 and E-T5-MICRO-F1. E-T4-F1 presents the F_1 score with Exact matching for quadruples and E-T5-MACRO-F1, and E-T5-MICRO-F1 indicate the macro and micro F_1 scores with Exact matching for quintuples. The formulae are shown as follows.

$$E - T4 - F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

$$E - T5 - F1 - macro = avg \sum E - T5 - F1 - c$$

with c presenting the F_1 *score* of each class, $c \in$ {COM+, COM-, COM, SUP+, SUP-, SUP, EQL, DIF}.

$$E - T5 - F1 - micro = \frac{TP}{TP + \frac{1}{2} \times (FP + NP)}$$

with the sum of true positive, false positive, true negative, and false negative values across all classes.

5. Results and Discussion

This section presents a comprehensive overview of the model's performance on the public and private test sets. The section is divided into four subsections. Subsection 5.1 shows the results of the baseline and improved models, highlighting the improvement after

[1] huggingface.co/chieunq/vietnamese-sentence-paraphase

Table 2. Result of Private test

	E-T5-MACRO-F1	E-T5-MACRO-P	E-T5-MACRO-R	E-T5-MICRO-F1	E-T5-MICRO-P	E-T5-MICRO-R
Best run of other teams						
thindang	0.2373	**0.2862**	0.2216	**0.2952**	0.2880	0.3029
thanhlt998	0.2131	0.2093	0.2199	0.2941	**0.2941**	0.2941
duyvu1110	0.1119	0.0964	0.1375	0.2092	0.1709	0.2698
ComOM_RTX5000	0.0997	0.0968	0.1065	0.1778	0.1675	0.1894
Our runs						
pthutrang513_baseline	0.2300	0.2021	0.2718	0.2684	0.2234	0.3359
pthutrang513_postcomp	**0.2391**	0.2078	**0.2894**	0.2791	0.2240	**0.3700**

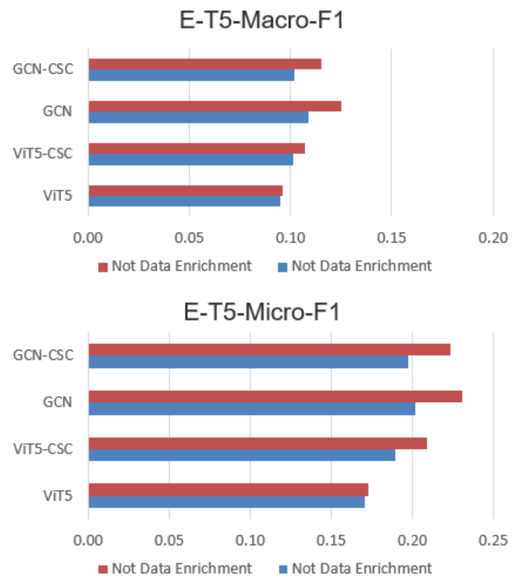


Figure 7. Results of data enrichment by single models.

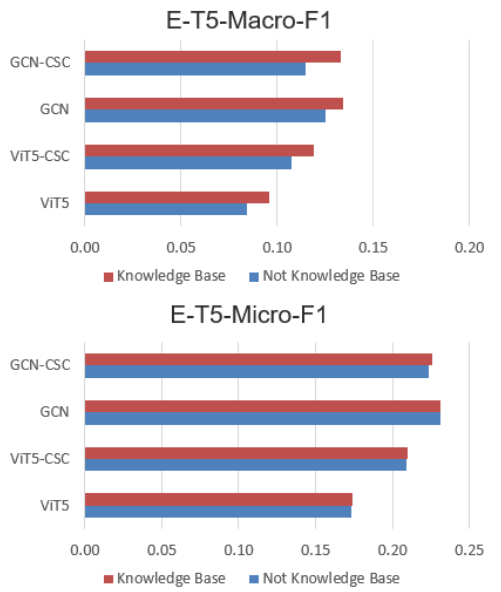


Figure 8. Results of knowledge base by single models.

the competition. The following subsection demonstrates the contribution of each component in the architecture via an ablation test. The third subsection gives detailed results of each stage and the last subsection opens a discussion of the overall result and analysis of errors.

5.1. Results of Shared Task

Table 2 describes the results on the private test set, showcasing the performance of our baseline ensemble model submitted for the VLSP 2023 competition. Initially, two GCN models were ensembled, followed by further ensembling of this output with the ViT5 model, integrating comparative sentence classification for input enhancement. Our baseline model achieved a 0.23 E-T5-F1-macro score, finishing second place in the task as announced by the VLSP organizers. In addition, ensembling models helped achieve the highest Recall scores in both macro and micro.

Post-competition, architectural enhancements were made to the model, incorporating the results of the T5 model without comparative sentence classification and integrating data augmentation during training, leading to an increase in both F1-macro and F1-micro scores, approximately 0.01 higher than the baseline. Consequently, the F1-macro score surpassed the first-place team's score with 0.2373. Furthermore, the recall score for the improved model exhibited significant improvements, increasing by 0.0341 in micro and

Table 3. Ablation for Public test

	Models	**E-T5-Macro-F1**	**E-T5-Macro-P**	**E-T5-Macro-R**	**E-T5-Micro-F1**	**E-T5-Micro-P**	**E-T5-Micro-R**
1	ViT5	0.1157	0.1380	0.1000	0.2005	0.2363	0.1741
	ViT5-CSC	0.1473	0.1697	0.1308	0.2548	0.2900	0.2272
	GCN	0.1359	0.1411	0.1357	0.2542	0.2516	0.2569
	GCN-CSC	0.1550	0.1548	0.1559	0.2621	0.2588	0.2654
2	Ensemble-GCN	0.1998	0.2074	0.2081	0.3004	0.2719	0.3355
	Ensemble-ViT5	0.1840	0.1971	0.1635	0.2914	**0.3061**	0.2781
3	Ensemble-w-o-ViT5	0.1895	0.1597	0.2409	0.2727	0.2161	0.3694
	Ensemble-w-o-ViT5-CSC	0.1890	0.1711	0.2186	0.2816	0.2345	0.3524
	Ensemble-w-o-GCN	0.2011	0.1906	0.2293	0.3023	0.2612	0.3588
	Ensemble-w-o-GCN-CSC	0.2024	0.1914	0.2314	0.2942	0.2547	0.3482
4	**Ensemble**	**0.2270**	**0.2024**	**0.2760**	**0.3081**	0.2453	**0.4140**

0.0176 in macro.

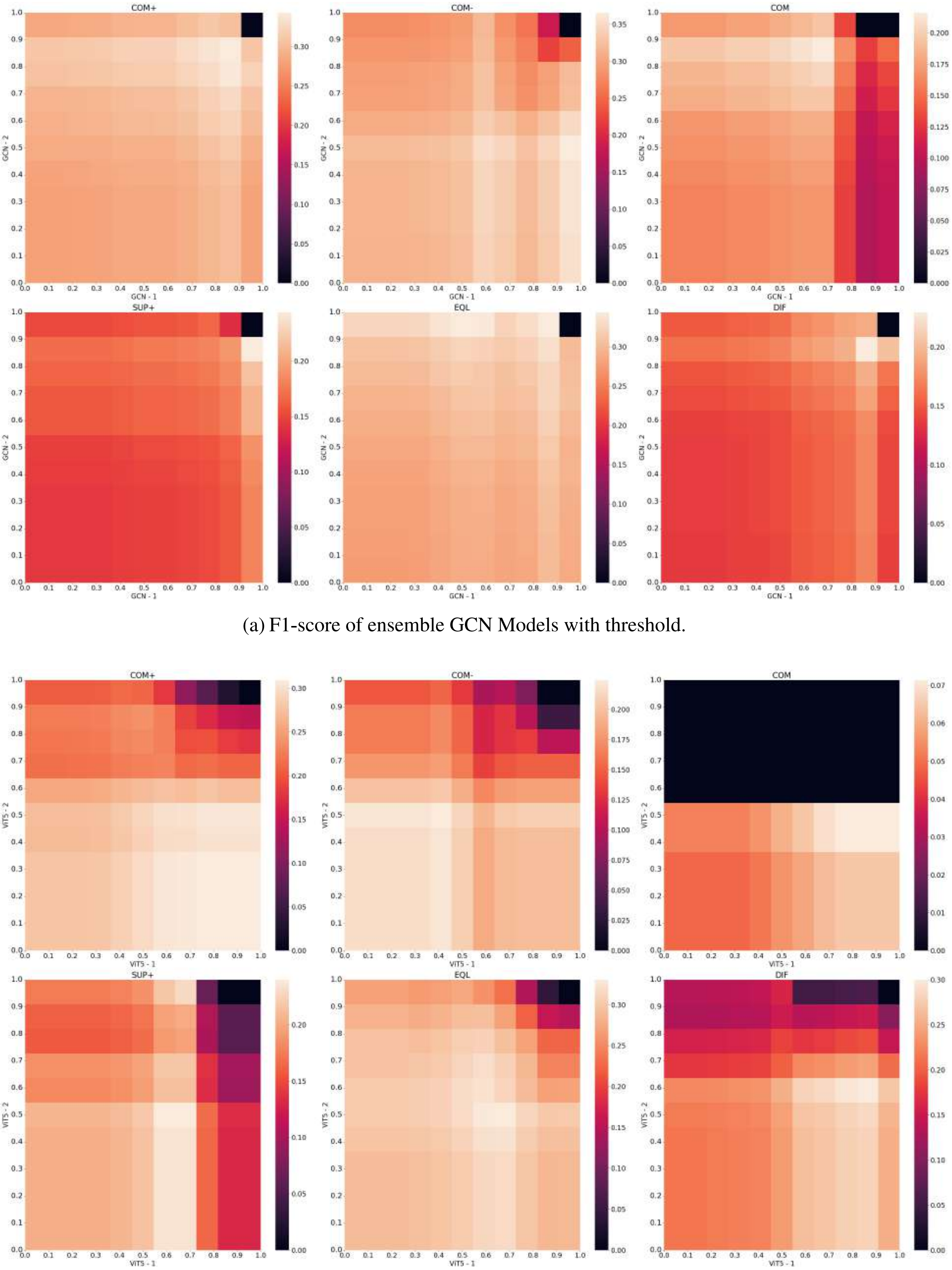
5.2. Ablation Test

Ablation experiments are conducted to verify the effect of data augmentation, knowledge base on the performance of single models and the effect of single model performance on ensemble model performance. Figure 7 shows the impact of data enrichment on single models. It is evident that the performance of all models improved when trained on the augmented dataset in both F1-macro and F1-micro. The performance of the knowledge base is presented in Figure 8. Since the knowledge base focused on addressing the label imbalance issue, it can be observed that the F1-macro score has increased, but the F1-micro score does not change much.

Table 3 shows the experimental results of single models and ensemble models on public test. When it comes to single models, the implementation of a comparative sentence classification model has improved the performance of ViT5 and GCN models, with F1 scores increasing from 0.01 to 0.02 for both macro-F1 and micro-F1. Ensemble models have been shown to outperform single models, especially in recall. The model that ensembles two or three models increases from 0.01 to 0.11 in recall compared to the single models. Nonetheless, they are still lower than the results of ensembling all 4 models with the highest recall of 0.4140 in micro and 0.3081 in macro. This indicates that each model captures different patterns in the data and their combination leads to improved performance.

Additionally, Figure 9a and Figure 9b illustrate the results of combining two GCN models and ensembling two ViT5 models with selected thresholds, respectively. The relatively balanced nature of the charts further highlights the significant contribution of each individual model. Furthermore, it is observed that popular labels generally have better overall results than the results for labels with less data. When ensembling the models together, in addition to the number of correctly predicted quintuples increasing significantly, the total number of all quintuples generated from the ensemble model also increases. That led to a decrease in precision and in the end the F1 score only increased by 0.072, 0.046 in Macro and Micro respectively.

As for the private test, the detailed results are displayed in Table 4. The results of the single models decrease from 0.004 to 0.04 compared to the public test set. This shows that the private test data set is slightly more complicated than the public one. The ensemble models results continue to increase significantly from 0.04 to 0.2 in F1 of Macro, and Micro compared to other single models. Among the ensemble models, the best result is still obtained by ensembling 4 models. However, the ensemble model still has the same trade-off problem between precision and recall as public testing.



(a) F1-score of ensemble GCN Models with threshold.

(b) F1-score of ensemble ViT5 Models with threshold.

Figure 9. F1-scores of ensemble Models with threshold.

Table 4. Ablation for Private test

	Models	E-T5-Macro-F1	E-T5-Macro-P	E-T5-Macro-R	E-T5-Micro-F1	E-T5-Micro-P	E-T5-Micro-R
1	ViT5	0.0959	0.1234	0.0792	0.1731	0.2162	0.1443
	ViT5-CSC*	0.1074	0.1275	0.0932	0.2091	**0.2481**	0.1806
	GCN	0.125	0.1284	0.1231	0.2308	0.2336	0.228
	GCN-CSC*	0.115	0.1141	0.1166	0.2238	0.2187	0.2291
2	Ensemble-ViT5	0.1349	0.1479	0.1266	0.2267	0.2428	0.2126
	Ensemble-GCN	0.1621	0.1427	0.1973	0.2629	0.2273	0.3117
3	Ensemble-w-o-ViT5	0.2300	0.2021	0.2718	0.2684	0.2234	0.335
	Ensemble-w-o-ViT5-CSC*	0.2105	0.1957	0.235	0.267	0.2245	0.3293
	Ensemble-w-o-GCN	0.2093	0.1991	0.2302	0.2671	0.2313	0.3161
	Ensemble-w-o-GCN-CSC*	0.2192	0.2072	0.2388	0.2726	0.2403	0.315
4	**Ensemble**	**0.2391**	**0.2078**	**0.2894**	**0.2791**	0.2240	**0.3700**

5.3. *Results of Classification Models, Extraction and Generation Models*

In this subsection, independent experimental results of each stage are presented, which include the result of classification models and quadruple (T4) extraction.

5.3.1. *Results of Classification Models*

Experiments are conducted with pre-trained models such as PhoBERT[2], XLM-RoBERTa-Large[3], XLM-RoBERTa-Base[4] with different parameters to determine the most suitable Comparative Sentence Classification and Comparative Label Classification models for the pipeline.

Table 5. Sentence Classification Models Experimental Results

Model	Precision	Recall	F1
PhoBERT-2layer	0.8031	0.8997	0.8486
PhoBERT-4layer	0.7635	0.9341	0.8402
PhoBERT-8layer	0.7761	0.8940	0.8309
XLM-RoBERTa-2layer	0.7914	0.8481	0.8188
XLM-RoBERTa-4layer	**0.8524**	**0.8768**	**0.8644**
XLM-RoBERTa-8layer	0.8343	0.8653	0.8495

Regarding the Comparative Sentence Classification model, we experimented with changing the number of final layers of the pre-trained models, which were taken and combined to pass through the final softmax layer. With PhoBERT, fewer layers result in better performance. Specifically, the PhoBERT-2layer model achieves the highest F1-score of 0.8486. When increasing the number of layers to 4 or 8, the F1-score slightly decreases. In contrast, with XLM-RoBERTa-Large, the model with 4 layers performs best with an F1-score of 0.8644, higher than the 2-layer and 8-layer models. The F1-score of XLM-RoBERTa-Large is only about 0.02 higher than PhoBERT-2layer, but XLM-RoBERTa-Large shows more stability with Precision and Recall scores of 0.8524 and 0.8768, respectively. The detailed results of each model are shown in Table 5.

We also considered the results of the models further when changing the threshold. As illustrated in Figure 10, XLM-RoBERTa-Large model with the last 4 hidden state embeddings of the sentence classification task achieved better results than the other models considered.

Table 6. Label Classification Models Results

Model	P-macro	R-macro	F1-macro	Accuracy
PhoBERT	0.6088	0.5788	0.5700	0.8910
XLM-RoBERTa-Base	0.4525	0.3622	0.3338	0.7224
XLM-RoBERTa-Large	**0.7650**	**0.7400**	**0.7563**	**0.8948**

Table 6 shows the performance of the Comparative Label Classification model, where XML-RoBERTa-Large achieves the highest results. Both XLM-RoBERTa-Large and PhoBERT are pretrained models that perform

[2] huggingface.co/vinai/phobert-base-v2
[3] huggingface.co/xlm-roberta-large
[4] huggingface.co/xlm-roberta-base

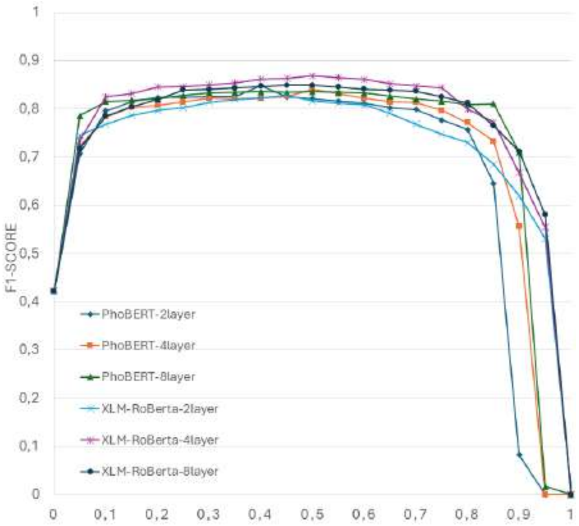


Figure 10. F1-score of Sentence Classification Models with threshold.

well for Vietnamese, and their accuracy results are nearly equal. However, the F1-macro score of XML-RoBERTa-Large is significantly higher than that of PhoBERT by about 0.19. This indicates that the XML-RoBERTa-Large model is more effective across all labels, even for those with less data. Therefore, XML-RoBERTa-Large is used for both sentence classification and label classification models.

5.3.2. Results of Extraction and Generation Models

Table 7 shows the outcomes of quadruple generation for each standalone model, considering whether data augmentation (DA) and comparative sentence classification (CSC) are employed or not. It is evident that when considering individual models on the public test set without the augmented data, the performance of GCN surpasses that of the ViT5 model by approximately 0.0017 to 0.0303. Data augmentation enhances the effectiveness of generating the first four elements in all models, particularly with the two GCN models, which increase by 0.0234 and 0.0414.

Table 7. Public test T4 result

Model	E-T4-F1	E-T4-P	E-T4-R
ViT5	0.2072	0.2515	0.1762
ViT5-CSC	0.2311	0.1975	0.2784
GCN	0.2524	0.2484	0.2566
GCN-CSC	0.2649	0.2797	0.2516
ViT5-DA	0.2103	0.2478	0.1826
ViT5-CSC-DA	0.2726	**0.3077**	0.2447
GCN-DA	**0.2938**	0.2895	**0.2981**
GCN-CSC-DA	0.2883	0.2826	0.2944

On the private test set, the results of quadruples from GCN models continue to be slightly better than ViT5. It can be confirmed that data augmentation helps to improve the models' performance, as shown in Table 8.

Table 8. Private test T4 result

Model	E-T4-F1	E-T4-P	E-T4-R
ViT5	0.1881	0.2386	0.1553
ViT5-CSC	0.2057	0.2429	0.1784
GCN	0.2206	0.2235	0.2178
GCN-CSC	0.2232	0.2315	0.2155
ViT5-DA	0.1902	0.2376	0.1586
ViT5-CSC-DA	0.2282	**0.2708**	0.1971
GCN-DA	**0.2519**	0.2539	0.2500
GCN-CSC-DA	0.2447	0.2369	**0.2530**

5.4. Error Analysis and Discussion

Figure 11 and 12 is the further analysis of the ensemble model's performance in extracting components of quadruples. In Figure 11, it can be observed that the scores for each element are relatively similar. The precision score is relatively low, which is a trade-off to achieve higher recall. The result of predicting the aspect element (A) is the lowest. Based on our data analysis, this happens because the aspect has a high empty rate, and it is a complex and ambiguous element, which can cause noise. For example, it can be easily seen that in table 9a, "độ phân giải" which is predicted by the model has the same meaning as "có độ phân giải" which is the gold label. Considering the Binary and Proportional metrics in Figure 12, the model determines the components in the quintuple relatively accurately,

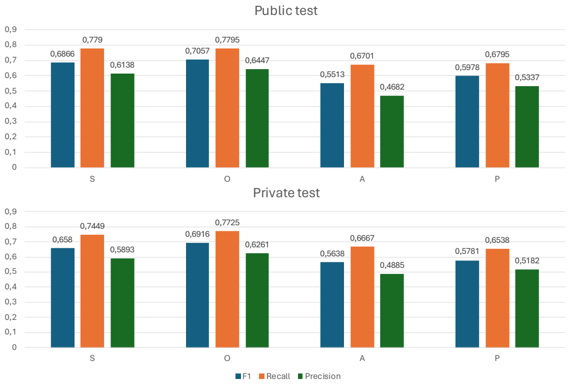


Figure 11. F1, Recall, Precision of each element in quadruples.

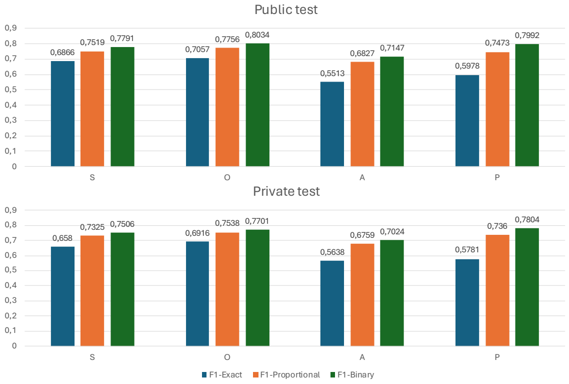


Figure 12. F1-E, F1-P, F1-B of each elements in quadruples.

resulting in a high F1-Binary score. The F1-Proportional score is approximately equivalent to the F1-Binary. This proves that when the model could determine the position of each component, it predicts that component relatively accurately. However, F1-Exact score is low. This is partly because the training data contains noise, and partly because the extracted components are relatively complex and easily confused, especially the element Aspect.

In addition, because of the imbalance of labels and the confusion between rare and non-rare labels, the model struggles to accurately predict rare labels. This is clearly shown in the second example of Table 9a.

On the other hand, our proposed model can address a few challenges. This is shown in Table 9b. Example 1 shows that the ensemble model can generate all quintuples in long and complex sentences. For errors as shown in examples 1 in Table 9a, our model partially solves them by combining all quintuples learned from different aspects by the individual models as in example 2.

6. Conclusion and Future work

In this study, a method for extracting opinion quintuples is introduced, combining GCN-based extraction models, generation models, and classification models. Comparative sentences are extracted using classification models, and subsequently, GCN, ViT5, and BERT-based classifiers are employed to extract the opinion quintuples. A data augmentation method is proposed to enhance the models' performance. Experimental results demonstrate the superiority of the method and augmented data over approaches presented in the VLSP 2023 competition. In future work, more sophisticated augmentation methods could be explored to enhance the diversity of data.

Acknowledgement

This work has been supported by VNU University of Engineering and Technology under project number CN24.15.

References

[1] N. Jindal, B. Liu, Mining Comparative Sentences and Relations, in: Proceedings of the 21st National Conference on Artificial Intelligence - Volume 2, AAAI'06, AAAI Press, 2006, p. 1331–1336.

[2] H.-Q. Le, D.-C. Can, K.-V. Nguyen, M.-V. Tran, Overview of the VLSP 2023 - ComOM Shared Task: A Data Challenge for Comparative Opinion Mining from Vietnamese Product Reviews, in: Proceedings of the 10th International Workshop on Vietnamese Language and Speech Processing, 2023, https://doi.org/10.48550/arXiv.2402.13613.

Table 9. Examples of Model Predictions

(a) Examples of Incorrect Predict by the Model

ID	Sentence	Predict	Gold label
1	Theo đánh giá cá nhân , tôi vẫn thích camera trên iPhone 11 hơn , ít nhất nó có độ phân giải cao hơn (12MP) . (**English:** In my personal opinion, I still prefer the camera on the iPhone 11; at least it has a higher resolution (12MP).)	{"nó", "", **"độ phân giải"**, "cao hơn", "COM+"} (**English:** {"it", "", **"resolution"**, "higher", "COM+"})	{"nó", "", **"có độ phân giải"**, "cao hơn", "COM+"} (**English:** { "it", "", **"has a resolution"**, "higher", "COM+"})
2	Với iPhone 12, bạn sẽ nhận được màn hình có độ phân giải thấp hơn , kích thước nhỏ hơn và độ sáng cũng như tốc độ làm mới thấp hơn Mi 11 . (**English:** With the iPhone 12, you get a lower-resolution, smaller display with lower brightness and refresh rate than the Mi 11.)	{"iPhone 12", "Mi 11", "kích thước", "nhỏ hơn", **"COM-"**} (**English:** {"iPhone 12", "Mi 11", "size", "smaller", **"COM-"**})	{"iPhone 12", "Mi 11", "kích thước", "nhỏ hơn", **"COM"**} (**English:** {"iPhone 12", "Mi 11", "size", "smaller", **"COM"**})

(b) Examples of Correct Predict by the Model

ID	Sentence	Predict	Gold label

1	Phần mềm không có khác biệt quá nhiều so với các điện thoại Samsung khác gần đây hay cả những điện thoại của 2019 .
	(**English:** The software is not significantly different from other recent Samsung phones or even phones from 2019.)
		{"", "các điện thoại Samsung khác gần đây", "Phần mềm", "không có khác biệt quá nhiều", "EQL"}
		{"", "những điện thoại của 2019", "Phần mềm", "không có khác biệt quá nhiều", "EQL"}
		(**English:** {"", "other recent Samsung phones", "The software", "not significantly different", "EQL"}
		{"", "phones from 2019", "The software", "not significantly different", "EQL" })
			{"", "các điện thoại Samsung khác gần đây", "Phần mềm", "không có khác biệt quá nhiều", "EQL"}
			{"", "những điện thoại của 2019", "Phần mềm", "không có khác biệt quá nhiều", "EQL"}
			(**English:** {"", "other recent Samsung phones", "The software", "not significantly different", "EQL"}
			{"", "phones from 2019", "The software", "not significantly different", "EQL"})

2	Khi nói đến chất lượng hoàn thiện , Xiaomi chắc chắn hoàn toàn ngang bằng với Galaxy S23 Ultra .
	(**English:** When it comes to build quality, Xiaomi is definitely completely equal to the Galaxy S23 Ultra.)
		{"Xiaomi", "Galaxy S23 Ultra", "chất lượng hoàn thiện", "hoàn toàn ngang bằng", "EQL" }
		{"Xiaomi", "Galaxy S23 Ultra", "chất lượng hoàn thiện", "ngang bằng", "EQL"}
		(**English:** {"Xiaomi", "Galaxy S23 Ultra", "build quality", "completely equal", "EQL" }
		{"Xiaomi", "Galaxy S23 Ultra", "build quality", "equal", "EQL" })
			{"Xiaomi", "Galaxy S23 Ultra", "chất lượng hoàn thiện", "hoàn toàn ngang bằng", "EQL" }
			(**English:** {"Xiaomi", "Galaxy S23 Ultra", "build quality", "completely equal", "EQL" })

[3] M. D. Rocklage, D. D. Rucker, L. F. Nordgren, Mass-scale Emotionality Reveals Human Behaviour and Marketplace Success, Nature human behaviour, Vol. 5, No. 10, 2021, pp. 1323–1329, https://doi.org/10.1038/s41562-021-01098-5.

[4] S. Sun, C. Luo, J. Chen, A Review of Natural Language Processing Techniques for Opinion Mining Systems, Information fusion, Vol. 36, 2017, pp. 10–25, https://doi.org/10.1016/j.inffus.2016.10.004.

[5] H. Q. Le, D. C. Can, Q. T. Ha, N. Collier, A Richer-but-Smarter Shortest Dependency Path with Attentive Augmentation for Relation Extraction, in: 2019 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Vol. 1, Association for Computational Linguistics, 2019, pp. 2902–2912, https://doi.org/10.18653/v1/N19-1298.

[6] Z. Liu, R. Xia, J. Yu, Comparative Opinion Quintuple Extraction from Product Reviews, in: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, 2021, pp. 3955–3965, https://doi.org/10.18653/v1/2021.emnlp-main.322.

[7] A. Bondarenko, Y. Ajjour, V. Dittmar, N. Homann, P. Braslavski, M. Hagen, Towards Understanding and Answering Comparative Questions, in: Proceedings of the Fifteenth ACM International Conference on Web Search and Data Mining, 2022, pp. 66–74, https://doi.org/10.1145/3488560.3498534.

[8] J. Arora, S. Agrawal, P. Goyal, S. Pathak, Extracting Entities of Interest from Comparative Product Reviews, in: Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, 2017, pp. 1975–1978, https://doi.org/10.1145/3132847.3133141.

[9] H. Cai, R. Xia, J. Yu, Aspect-Category-Opinion-Sentiment Quadruple Extraction with Implicit Aspects and Opinions, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 340–350, https://doi.org/10.18653/v1/2021.acl-long.29.

[10] Q. Xu, Y. Hong, F. Zhao, K. Song, J. Chen, Y. Kang, G. Zhou, GCN-based End-to-End Model for Comparative Opinion Quintuple Extraction, in: 2023 International Joint Conference on Neural Networks (IJCNN), IEEE, 2023, pp. 1–6, https://doi.org/10.1109/IJCNN54540.2023.10191436.

[11] Z. Yang, F. Xu, J. Yu, R. Xia, UniCOQE: Unified Comparative Opinion Quintuple Extraction as a Set, in: Findings of the Association for Computational Linguistics: ACL 2023, 2023, pp. 12229–12240, https://doi.org/10.18653/v1/2023.findings-acl.775.

[12] N. X. Bach, D. Van Pham, N. D. Tai, T. M. Phuong, Mining Vietnamese Comparative Sentences for Sentiment Analysis, in: 2015 Seventh International Conference on Knowledge and Systems Engineering (KSE), IEEE, 2015, pp. 162–167, https://doi.org/10.1109/KSE.2015.36.

[13] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, V. Stoyanov, Unsupervised Cross-lingual Representation Learning at Scale, in: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics, Online, 2020, pp. 8440–8451, https://doi.org/10.18653/v1/2020.acl-main.747.

[14] T. N. Kipf, M. Welling, Semi-Supervised Classification with Graph Convolutional Networks, in: International Conference on Learning Representations, 2017, https://doi.org/10.48550/arXiv.1609.02907.

[15] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, 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 (Long and Short Papers), Association for Computational Linguistics, Minneapolis, Minnesota, 2019, pp. 4171–4186, https://doi.org/10.18653/v1/N19-1423.