NOWJ Team's Approach in ComOM@VLSP 2023: A Multi-Stage End-to-End Architecture for Vietnamese Comparative Opinion Mining

Thu-Trang Pham[1], Huu-Dong Nguyen[1], Thach-Anh Nguyen[1], Hoang-Trung Nguyen[1], Tan-Minh Nguyen[1], Dieu-Quynh Nguyen[1], Ha-Thanh Nguyen[2], Thi-Hai-Yen Vuong[1]
[1]*VNU University of Engineering and Technology*, Hanoi, Vietnam
[2]*National Institute of Informatics*, Tokyo, Japan
[1]{21020248, 21020760, 21020166, 20020083, 20020081, 20021424, yenvth}@vnu.edu.vn
[2]nguyenhathanh@nii.ac.jp

Abstract

With the explosion of e-commerce and online shopping, Opinion Mining tasks have received great attention from both the academic and industry fields. As an important task in opinion mining, Comparative Opinion Quintuple Extraction (COQE) aims to extract comparative relations at the sentence level and obtain the corresponding comparative opinion tuples from product reviews. It is required to identify quintuple elements that construct a specific comparison including subject, object, aspect, predicate, and comparison type. In this paper, we propose a multi-stage end-to-end architecture for Comparative Opinion Mining from Vietnamese product reviews task. The first stage is to identify comparative sentences and extract quadruples of (subject, object, aspect, predicate) using a combination of GCN and generative models. The second stage is to further classify each comparative quadruple into one of eight comparison types. Our proposals achieved competitive results and finished at the second rank in the challenge on Comparative Opinion Mining from Vietnamese Product Reviews - VLSP 2023.

1 Introduction

Comparative opinion mining aims to identify comparative instances from product reviews, which can be used to support the customers' decisions and companies' strategies. This year, a challenge on comparative opinion mining from Vietnamese product reviews was first organized in the VLSP 2023. The goal of this challenge is to develop NLP models that can extract comparative information quintuples. The quintuple is defined as (subject, object, aspect, predicate, and comparison type label). The challenge comprises two sub-tasks including 1) identifying comparative sentences that contain comparisons between devides, and 2) extracting comparative quintuple from these sentences. However, more than one quintuple can be extracted from a single comparative sentence. This leads to multiple outputs given a review, as shown in 1).

1. *"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."*

Quintuple Constituents
Subject: iPhone
Object: thế hệ trước
Aspect: {phần cứng, phần mềm}
Predicate: {nâng cấp nhẹ}
Comparison type: COM+
Results
{iPhone, thế hệ trước, phần cứng, nâng cấp nhẹ, COM+}
{iPhone, thế hệ trước, phần mềm, nâng cấp nhẹ, COM+}

Table 1: Quintuples extracted from the example 1

To overcome the problem, we propose a multi-stage end-to-end architecture to extract comparative quintuples from Vietnam product reviews. The first stage is to identify comparative sentences and extract quadruples of (subject, object, aspect, predicate) using GCN and generative models. In the next stage, PhoBERT models are employed to classify quadruple into one of eight comparison types. The experiment results show that our approach achieved significant results and ranked second place in the challenge (Le et al., 2023).

2 Related Works

Comparative opinion statements comprise 10% the total text (Kessler and Kuhn, 2013), and these statements contain meaningful information to the consumers. Therefore, comparative opinion mining has been a topic of significant research interest (Varathan et al., 2017). It consists of recognizing comparative elements from the text and the comparison relation between entities. Comparative opinion mining covers both Comparative Sentence Identifying and Comparative Constituent Extraction tasks, starting from simple sentence classification to complex comparative element extraction.

2.1 Comparative Sentence Identifying

Jindal et al. (Jindal and Liu, 2006) proposed novel techniques based on label sequential rule (LSR) to classify comparative sentences from the texts and extract relations from the identified comparisons. Park and Blake (Park and Blake, 2012) introduced a set of semantic and syntactic features that are used to boost classifiers: Naive Bayes (NB), Support Vector Machine (SVM), and Bayesian network (BN). This approach was applied to identify comparative statements from full-text scientific articles. Wang et al. (Wang et al., 2015) proposed an SVM model to classify comparative and non-comparative statements using linguistic and statistical features. They applied their models to Chinese customer reviews. Liu et al (Liu et al., 2013) employed rule-based and Class Sequential Rules (CSR) methods to identify comparative sentences.

2.2 Comparative Constituent Extraction

Rule-based is one of the most traditional and common methods in the field of NLP. Ganapathibhotla et al. (Ganapathibhotla and Liu, 2008) considered the associations between comparative words and the sentiment of the reviews to identify the preferred entity in the comparative sentences. They experimented using comparative sentences from product reviews and forum posts to show that their approach is effective. Arora et al. (Arora et al., 2017) first presented a neural network approach to extract multi-entity extraction for product comparison. They employed LSTMs to capture the relations of comparative entity, aspect, and opinion. Recently, a new Comparative Opinion Quintuple Extraction (COQE) task (Liu et al., 2021) is introduced with the aim of extracting all the comparative quintuples from product reviews. Besides, they proposed a novel multi-stage deep learning approach for this task.

3 Methods

The COQE task can be formulated as follows: Given a product review $X = \{x_1, x_2, \ldots, x_n\}$ consisting of n tokens, the task aims to determine whether X is a comparative sentence and extract all comparative quintuples in it:

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

In each tuple, sub and obj respectively refer to the subject and object being compared whereas asp denotes the comparative aspect of the entities and pre denotes the comparative word or phrase expressing the comparison. The $label$ indicates the type of comparison. The first four elements of the quintuple need to be extracted from the sentence, while the $label$ needs to be classified from predefined categories.

To tackle this task, a method ensembling an extraction model, a generation model and classification models is introduced. The next section specifies the details of the proposed method.

3.1 Extraction Model

GCN-based Encoder. Given X, the context-aware attention-weighted representation for each token using the pre-trained BERT is obtained

$$H = BERT(X) \tag{1}$$

where H indicates the hidden state after the BERT encoding layer.

To improve the encoder's ability to perceive adhesion among the quintuple components, H is fed into a Graph Convolutional Network (GCN). Specifically, Underthesea toolkits are employed to parse dependency relations among tokens for Vietnamese sentences. The one-hot adjacency matrix is constructed based on the parsed arcs of a sentence, where 1 indicates that two connected tokens are related and 0 indicates that two tokens are not related. Note that undirected graphs are used to construct the adjacency matrix, meaning that the matrix is symmetrical. Then, the GCN is used to impose the effects of the adjacency matrix upon the initial embeddings H, in order to create an adhesion-sensitive representation:

$$H^{(l)} = f(AH^{(l-1)}W^{(l-1)}) \tag{2}$$

where $H^{(l)}$ denotes the updated representation at layer l, A is the adjacency matrix, W refers to the trainable weight matrix at the current GCN layer, and f is the ReLU activation function. In our model, a single-layer GCN is utilized.

Non-autoregressive Decoder. In this method, a non-autoregressive decoder is used to generate the quintuple constituents, instead of an autoregressive model. Firstly, we randomly initialize the representations of different sets of quintuple constituents, which consist of N 768-dimensional embeddings. The number of embeddings is larger than the largest number of quintuples in all sentences in the training data. We denote such embeddings as queries Q. Q and H are then fed into the transformer decoder layer where self-attention will be first computed for Q, and further interattention is calculated between the self-attentive Q and H. In each decoder layer, the representation of Q is updated as follows:

$$Q^{(L)} = Decoder(H, Q^{(L-1)}) \quad (3)$$

Given the output of the final decoder layer $Q^{(L)}$, the type of comparison is determined using a discriminative model. The probabilities of all types of comparison are estimated as follows:

$$p^t = softmax(W_t Q^{(L)} + b) \quad (4)$$

where Wt and b are trainable parameters. The other classes of constituents are generated as follows:

$$p_i^e = softmax(V_i^T tanh(W_e q_i^L + W_h H)) \quad (5)$$

where W_e, W_h and V are all trainable parameters. q_i^L is the i-th embedding output by the final decoder layer. Note that during the generation process, a series of special tokens can be produced, such as {start, end} which indicates the beginning and ending of a constituent.

3.2 Generation Model

A ViT5 generation template is designed for end-to-end extraction of quadruples. Below is an example of the input and target of the model:

Input: *Dù thép không gỉ có khối lượng nặng hơn nhưng về độ bền thì chúng ta phải công nhận rằng chúng có mức độ chóng ăn mòn , chống gỉ và cứng hơn , đảm bảo bảo vệ điện thoại trước những va đập tốt hơn .*

Target: *(thép không gỉ > unknown > khối lượng > nặng hơn) > (thép không gỉ > unknown > độ bền > cứng hơn) > (unknown > unknown > bảo vệ điện thoại trước những va đập > tốt hơn) >*

In the generative paradigm, k golden quintuples are concatenated with " > " as the target sequence of the model. If the comparison element does not exist, it is padded with the word "*unknown*".

For the input sentence X, during the training phase, we temporarily turn off the gradient backpropagation of the model and send X into the ViT5-encoder to get the latent representation of the sentence:

$$h_c^{dec} = \textbf{Encoder}(X) \quad (6)$$

We then use ViT5-decoder 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 utilized as the input into the decoder:

$$h_c^{dec} = \textbf{Decoder}(h^{enc}, t_{1:c-1}) \quad (7)$$

The conditional probability of token t_c is defined as follows:

$$P(t_c|t_{1:c-1}, X) = Softmax(h_c^{dec} W + b) \quad (8)$$

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

$$T_{pred} = t_{1:m} = \{t_1, \ldots, t_m\} \quad (9)$$

where m is he length of the predicted sequence. T_{pred} is split with the greater than symbol " > " to get the set of comparative quintuple predicted by the model: $Q_{pred} = \{tup_1^{pred}, \ldots, tup_l^{pred}\}$.

3.3 Comparative sentence and comparative label classification

To improve the performance of the generation model and the extraction model, we propose a comparative sentence classification model for the input and a comparative label classification model for the output.

Comparative sentence classification. The comparison sentence classification model is used to predict whether sentence X is a comparative sentence. First, we put the sentence X through BERT to get the 4 representative classes at the end.

$$h8 = [h8_{[CLS]}, h8_1, \ldots, h8_n, h8_{[SEP]}]$$
$$h9 = [h9_{[CLS]}, h9_1, \ldots, h9_n, h9_{[SEP]}]$$
$$h10 = [h10_{[CLS]}, h10_1, \ldots, h10_n, h10_{[SEP]}]$$
$$h11 = [h11_{[CLS]}, h11_1, \ldots, h11_n, h11_{[SEP]}]$$

Then we pass 4 $h_{[CLS]}$ layers through a softmax layer to be able to predict which class the score of that quintuples belongs to is the highest.

$$h = h8 + h9 + h10 + h11 \quad (10)$$

$$y^c = softmax(W^h_{[CLS]} + b^c) \quad (11)$$

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

Comparison type classification. Although the extraction model or the generation model can generate the full five elements of the comparative opinion quintuple, a separate label classifier is built to make comparison type classification more effective. Similar to *Comparative sentence classification*, first, we combine aspect and predicate into 1 sentence. Then we add the prefix "hơn +, hơn -, hơn, nhất +, nhất -, nhất, bằng, khác" to the beginning of the sentence. Then we put the sentence through the model which has the same architecture of *Comparison sentence classification model*

3.4 Ensemble

During the ensemble phase, we begin with ensembling two extraction models. One of these models has a comparison sentence classifier for the input, and both models have a comparison label classifier for the output. We set thresholds θ based on the validation set. Quintuple constituents with score $score \geq \theta$ are returned as output. Then, we still ensemble this output and generation model which is added comparative sentence classification for the input and comparison type classification for the output.

4 Experiments and Results

4.1 Expreiments setup

Extraction Model. We employ the basic BERT version of bert-base-multilingual-cased as the encoder. The models are available on the Hugging Face website. During training, we use the AdamW optimizer and the dropout rate is set to 0.1. The learning rates for the encoder and decoder are set to 1e-5 and 2e-5 respectively. The batch size is set to 2. The maximum length of the input sentences is set to 128 for all datasets. The number of training epochs is set to 50 for all datasets. The initial representation is set to 5 for all datasets.

Generation Model. We utilize T5 as the underlying model architecture and conduct experiments using the ViT5-large model provided by the HuggingFace library. The batch size is set to 1. The learning rate for the model is configured at 1.5e-5. The training process is conducted over 30 epochs.

4.2 Results

In the competition, the Precision, Recall and F_1 score are calculated as follows:

$$Precision = \frac{\#correct}{\#predict} \quad (12)$$

$$Recall = \frac{\#correct}{\#gold} \quad (13)$$

$$F_1 = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (14)$$

In the given context, $\#predict$ represents the count of predicted comparative elements (or quintuples for COQE) by the model while $\#gold$ indicates the number of comparative elements (or quintuples) in the dataset and $\#correct$ signifies the number of correctly predicted comparative quintuples (or quintuples).

In this task, E-T5-F1-macro score is used as the principal measure. The formula is noted below.

$$E - T5 - F1 - macro = avg \sum x_i \quad (15)$$

with $x_i \in \{E{-}T5{-}F1{-}COM{+}, E{-}T5{-}F1{-}COM{-}, E-T5-F1-COM, E-T5-F1-SUP{+}, E-T5-F1-SUP{-}, E-T5-F1-SUP, E{-}T5{-}F1{-}EQL, E{-}T5{-}F1{-}DIF\}$

The result running on the private test is shown in Table 2. The model we submitted was the ensemble model described above. Our system achieved a 0.23 E-T5-F1-macro score, finishing second place in the task as announced by the VLSP organizers.

Table 2: Final results on private test set

Top	Precision	Recall	F1-macro
1	0.2862	0.2216	0.2373
2	**0.2021**	**0.2718**	**0.2300**
3	0.2093	0.2199	0.2131

Conclusion

In this paper, we demonstrate our approach to tackling the challenge of Comparative Opinion Mining from Vietnamese Product Reviews at VLSP 2023. Our proposed end-to-end architecture involves two main stages. The first stage aims to identify comparative sentences and extract quadruples of (subject, object, aspect, predicate) based on GCN and generative models. The second stage is to further classify each comparative quadruple into one of eight comparison types. Our proposals achieved promising results and ranked second place in the challenge. In future work, we will consider more sophisticated approaches and knowledge integration for the challenge.

References

Jatin Arora, Sumit Agrawal, Pawan Goyal, and Sayan Pathak. 2017. Extracting entities of interest from comparative product reviews. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pages 1975–1978.

Murthy Ganapathibhotla and Bing Liu. 2008. Mining opinions in comparative sentences. In *Proceedings of the 22nd international conference on computational linguistics (Coling 2008)*, pages 241–248.

Nitin Jindal and Bing Liu. 2006. Mining comparative sentences and relations. In *Aaai*, volume 22, page 9.

Wiltrud Kessler and Jonas Kuhn. 2013. Detection of product comparisons-how far does an out-of-the-box semantic role labeling system take you? In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1892–1897.

Hoang-Quynh Le, Duy-Cat Can, Khanh-Vinh Nguyen, and Mai-Vu Tran. 2023. 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*.

Chengxiang Liu, Ruifeng Xu, Jie Liu, Peng Qu, He Wang, and Chengtian Zou. 2013. Comparative opinion sentences identification and elements extraction. In *2013 international conference on machine learning and cybernetics*, volume 4, pages 1886–1891. IEEE.

Ziheng Liu, Rui Xia, and Jianfei Yu. 2021. Comparative opinion quintuple extraction from product reviews. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3955–3965.

Dae Hoon Park and Catherine Blake. 2012. Identifying comparative claim sentences in full-text scientific articles. In *Proceedings of the workshop on detecting structure in scholarly discourse*, pages 1–9.

Kasturi Dewi Varathan, Anastasia Giachanou, and Fabio Crestani. 2017. Comparative opinion mining: a review. *Journal of the Association for Information Science and Technology*, 68(4):811–829.

Wei Wang, TieJun Zhao, GuoDong Xin, and YongDong Xu. 2015. Exploiting machine learning for comparative sentences extraction. *International Journal of Hybrid Information Technology*, 8(3):347–354.