DOI: 10.17323/2587-814X.2024.1.79.88

Product information recognition in the retail domain as an MRC problem

Tho Chi Luong [a]
E-mail: tholc@vnu.edu.vn

Oanh Thi Tran [b] *
E-mail: oanhtt@gmail.com

[a] Institut Francophone International, Vietnam National University, Hanoi
Address: E5, 144 Xuan Thuy St., Cau Giay Dist., Hanoi, Vietnam

[b] International School, Vietnam National University, Hanoi
Address: G7, 144 Xuan Thuy St., Cau Giay Dist., Hanoi, Vietnam

Abstract

This paper presents the task of recognizing product information (PI) (i.e., product names, prices, materials, etc.) mentioned in customer statements. This is one of the key components in developing artificial intelligence products to enable businesses to listen to their customers, adapt to market dynamics, continuously improve their products and services, and improve customer engagement by enhancing effectiveness of a chatbot. To this end, natural language processing (NLP) tools are commonly used to formulate the task as a traditional sequence labeling problem. However, in this paper, we bring the power of machine reading comprehension (MRC) tasks to propose another, alternative approach. In this setting, determining product information types is the same as asking "Which PI types are referenced in the statement?" For example, extracting product names (which corresponds to the label PRO_NAME) is cast as retrieving answer spans to the question "Which instances of product names are mentioned here?" We perform extensive experiments on a Vietnamese public dataset. The experimental results show the robustness of the proposed alternative method. It boosts the performance of the recognition model over the two robust baselines, giving a significant improvement. We achieved 92.87% in the F1 score on recognizing product descriptions at Level 1. At Level 2, the model yielded 93.34% in the F1 score on recognizing each product information type.

* Corresponding Author

Keywords: product information recognition, MRC framework, retail domain, large-language models, viBERT, vELECTRA

Citation: Luong T.C., Tran O.T. (2024) Product information recognition in the retail domain as an MRC problem. *Business Informatics*, vol. 18, no. 1, pp. 79–88. DOI: 10.17323/2587-814X.2024.1.79.88

Introduction

Product Information (PI) is all the data about the products that a company sells. It includes a product's technical specifications, size, materials, prices, photos, schematics, etc. E-commerce requires companies to collect clear basic PI that consumers can actually understand and place orders. Without PI[1], the product could not be found and sold online at all.

Recognizing PI is crucial for widespread applications. For example, in the e-commerce field, it is vital to integrate this component to develop AI products like chatbots [1] to enhance customers' experiences. Chatbots significantly help reduce customer support costs, while increasing customer satisfaction with an AI chatbot that can recognize customer intents, instantly provide information, on any channel, and never take a day off. Identifying PI also helps to better analyze the sentiments [2] in comments/reviews of their customers. With PI, we can associate specific sentiments with different aspects of a product to analyze customer sentiments and opinions from reviews. This would improve the product and help make better marketing campaigns.

Conventionally, the task of PI recognition is formulated as a sequence labeling problem. It is a supervised learning problem that involves predicting an output sequence for a given input sequence. Most research in this field has proposed different machine learning approaches using handcrafted features or neural network approaches [3, 4] without using handcrafted features.

In this paper, we bring the power of machine reading comprehension (MRC) to this task. This idea is significantly inspired by a recent trend of transforming natural language processing (NLP) tasks to answering MRC questions. Specifically, Levy et al. [5] formulated the relation extraction task as a QA task. McCann et al. [6] transformed the tasks of summarization or sentiment analysis into question answering. For example, the task of summarization can be formalized as answering the question *"What is the summary?"* Li et al. [7] formalized the task of entity-relation extraction as a multi-turn question-answering problem.

So far, most current work has focused on high-resource languages. Therefore, to narrow the gap between low and high-resource languages, this paper also targets the Vietnamese language. This paper proposed an alternative way to extract PI by modeling it as a MRC problem. We conduct many extensive experiments on a public dataset by Tran et al. [1] and the results demonstrate that this approach introduces a significant performance boost over robust existing systems. The main contribution of this paper can be highlighted as follows:

- We proposed an alternative method to recognize PI by tailoring the MRC framework to suit the specific requirement of the task.
- We have conducted extensive experiments to verify the effectiveness of the proposed approach on a public Vietnamese benchmark dataset[2].

The remainder of this paper is organized as follows. Related work is presented in Section 1. Section 2 shows how to formulate the task as an MRC problem and then describes the method for generating questions, as

[1] In this paper, we consider seven types of PI types including *categories, attributes, extra-attributes, brands, packsizes, numbers,* and *unit-of-measurements (uoms)*.

[2] https://github.com/oanhtt84/PI_dataset/tree/main

well as the model architecture. Section 3 describes the experimental setups, experimental results, and some discussions. Finally, we conclude the paper and figure out some future lines of work.

1. Related work

This section first presents the work on PI identification, and then describes related work about the machine reading comprehension (MRC) tasks.

1.1. Work on PI recognition

Information retrieval chatbots are widely applied as assistants, to support customers formulate their requirements about the products they want when placing an order online. In order to develop such chatbots, most current systems use information retrieval techniques [8, 9] or a concept-based knowledge model [10] to identify product information details mentioned by their customers. Towards building task-oriented chatbots, Yan et al. [11] presented a general solution for online shopping. To extract PI asked by customers, the system matched the question to basic PI using the DSSM model. Unfortunately, these studies do not support customers who are performing orders online, and some external data resources exploited in their research are intractable in many actual applications.

Most work has been done for rich-resource languages such as English and Chinese; work for poor-resource languages is much rarer. In Vietnam, there is only one work focusing on recognizing PI types in the retail domain. Specifically, Tran et al. [1] introduced a study on understanding what the users say in chatbot systems. They concentrated on recognizing PI types implied in users' statements. In that work, they modeled the task as a sequence labelling problem and then explored different deep neural networks such as CNNs and LSTMs to solve the task.

1.2. Work on MRC

MRC refers to the ability of a machine learning model to understand and extract relevant information from written texts. It is similar to how a human reader would do this and accurately answer questions related to the content of the texts. The power of the MRC model is evaluated by the ability to extract the correct answer to the user question.

Many published novel datasets inspired a large number of new neural MRC models. In the past several years, we have witnessed many neural network models created such as BERT [12, 16] RoBERTa [13] and XLNet [10]. Many large language models utilize transformers [14] to pre-train representations by considering both the left and right context across all layers. Due to their remarkable success, this approach has progressively evolved into a mainstream method, involving pre-training large language models on extensive corpora and subsequently fine-tuning them on datasets specific to the target domain. Deep learning neural networks, particularly those based on transfer learning, are widely employed to address diverse challenges in natural language processing (NLP). Transfer learning methods emphasize the retention of data and knowledge acquired during the exploration of one problem, then applying this acquired knowledge to address different yet related questions. The effectiveness of these cutting-edge neural network models is noteworthy. For example, Lithe SOTA neural network models by Therasa et al. [12] has already exceeded human performance over many related MRC benchmark datasets.

In this paper, we borrow the idea of MRC to propose another alternative approach to this task. To prove the effectiveness of the approach, we conduct extensive experiments on a public Vietnamese dataset released by Tran et al. [1]. The results showed a new SOTA result over the traditional existing techniques.

2. Recognizing PI as an MRC problem

In this section, we first formulate the task of recognizing PI as an MRC problem. Then, we show the method to generate questions/queries for finding the answers (which could be the product information instances) appearing in the users' input utterances. Finally, the model architecture is presented and explained in more details.

2.1. Problem formulation

Given a users' statement x including n syllables $\{x_1, x_2, ..., x_n\}$, we need to build a model to identify every product information mentioned in x. For each instance of a product information type found in x we assign a label y to it. Here, y belongs to one of the pre-defined PI list including *product names*, *product size*, *product unit-of-measurements*, *product attribute*, *product brand*, *product number*, and *product extra attribute*.

To exploit the MRC approach, it is necessary to recast the task as an MRC problem. To this end, we construct triples of questions, answers, and contexts $\{q_{pi}, x_{start:end}, ..., x\}$ for each label pi mentioned in x as follows:

- x: the user's statement
- $x_{start:end}$: the product information mentioned in x. It is a sequence of syllables within x identified by the specified start and end indexes $\{x_{start}$, ... $x_{end}\}$, where the condition start <= end holds true. Expert knowledge is required to annotate this data.
- q_{pi}: the question to ask the model to find $x_{start:end}$ corresponding to the label pi. This is a natural question consisting of m syllables $\{q_1, q_2, ..., q_m\}$. Various approaches will be investigated in order to generate such questions.

This exactly establishes the triple (*Question, Answer, Context*) to be exploited in the proposed framework. And now, the task can be recast as an MRC problem as follows: Given a collection of k training examples $\{q^i, x^i_{start:end}, x^i\}$ (where $i = 1..k$). The purpose is to train a predictor which receives the statement x and the corresponding question q, and outputs the answer $x_{start:end}$. It is formulated as the following formula:

$$x^i_{start:end} = f(q^i, x^i).$$

2.2. Question generation

Each PI is associated with a specific question generated by combining the predefined templates and its training example values. It is a natural language question. In order to provide more prior knowledge about the label, we add some examples to the questions so that the model can recognize answers easier. These examples are randomly withdrawn from the training data set. Some typical generated questions for product information types are shown in *Table 1.*

Here we just provide some examples of each product information type to help the model find all of its instances appearing in the input statement.

Table 1.

Some questions generated for each PI types using templates

No.	Product information types	Generated questions
1	Product names	Which product names are mentioned in the text such as *smoothies* and *cakes*?
2	Product sizes	Which product sizes are mentioned in the text such as *big* and *small*?
3	Product colors	Which product colors are mentioned in the text such as *green* and *blue*?
4	Product uoms	Which product uoms are mentioned in the text such as *cup* and *cm*?
5	Product attributes	Which product attributes are mentioned in the text such as *extra ice* and *little sugar*?
6	Product extra attributes	Which product extra attributes are mentioned in the text such as *strawberry flavor* and *orange flavor*?
7	Product brand	Which brands are mentioned in the text such as *Samsung* and *Toyota*?

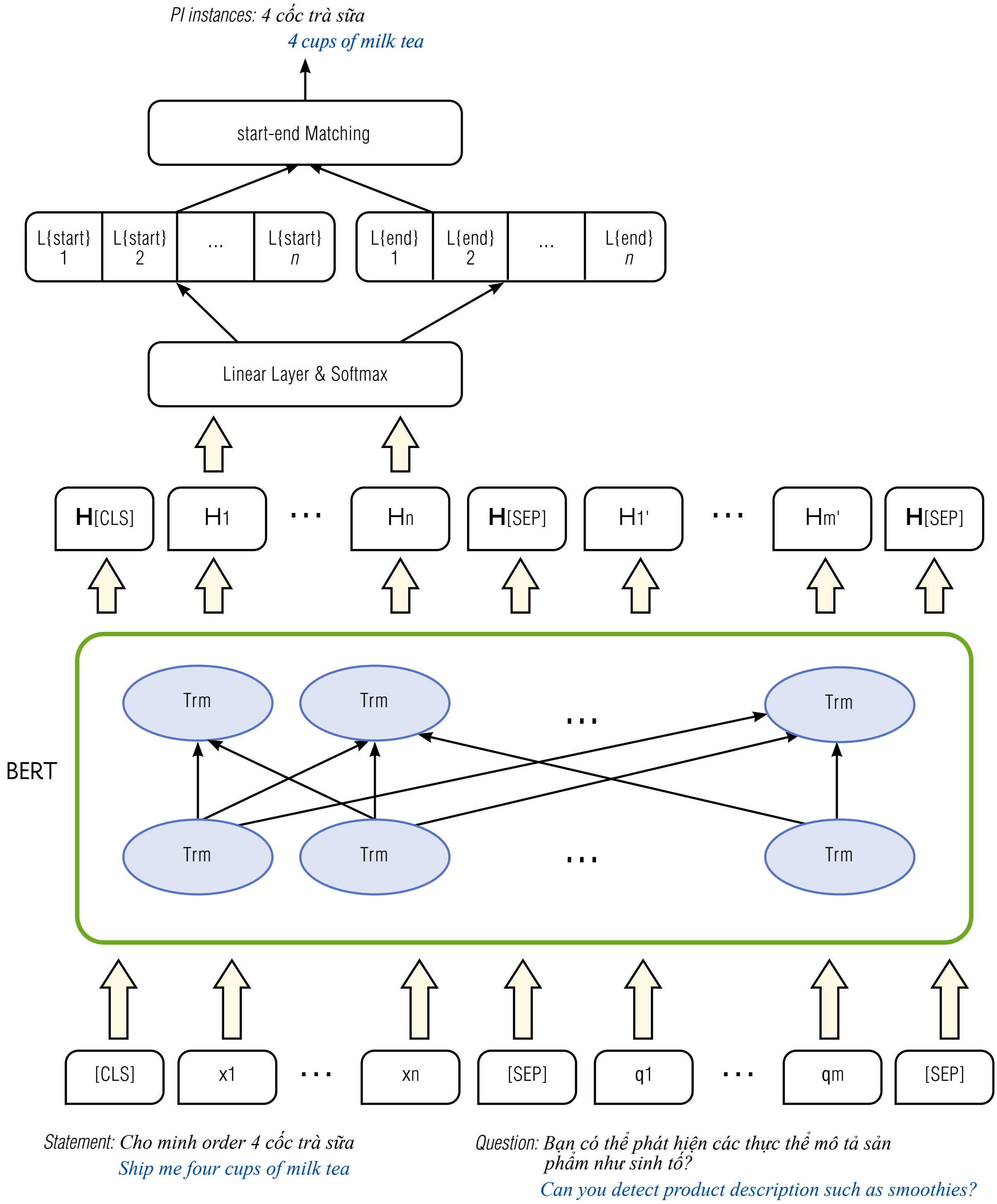
PI instances: 4 cốc trà sữa
4 cups of milk tea
start-end Matching
L{start} 1
L{start} 2
...
L{start} n
L{end} 1
L{end} 2
...
L{end} n
Linear Layer & Softmax
H[CLS]
H1
...
Hn
H[SEP]
H1'
...
Hm'
H[SEP]
BERT
Trm
...
[CLS]
x1
xn
[SEP]
q1
qm
Statement: Cho minh order 4 cốc trà sữa
Ship me four cups of milk tea
Question: Bạn có thể phát hiện các thực thể mô tả sản phẩm như sinh tố?
Can you detect product description such as smoothies?

Fig. 1. An architecture using BERT to solve the PI recognition task as an MRC problem.
(English translation is given right below the Vietnamese texts).

2.3. Model architecture

Figure 1 shows the general architecture which includes several main components. The model in this framework is built with a pre-trained large language model (i.e., BERT encoder) and a network designed to produce candidate for *start* and *end* indexes, along with their associated confidence scores indicating the likelihood of being product information.

Given the question q_{pi}, the purpose is to find the text span $x_{start:end}$$ categorized as the product information type *pi*. In the first step, q_{pi} and x are concatenated to establish the string $\{[CLS], q_1, q_2, ..., q_m; [SEP], [x_1, x_2, ..., x_n\}$,where $[CLS]$ and $[SEP]$ are special tokens employed in the conventional pre-trained LLMs. Then, the string is inputted into BERT to generate a contextual representation matrix $E \in R^{n \times d}$, (here d indicates the vector dimension of the final layer). Here, we do not make any prediction for the question, so its final vector representation is ignored.

2.4. Producing the indexes of start:end

To this end, we follow the method proposed by Li et al. [7] to build two corresponding binary classifiers. These two classifiers estimate the probability of each token to be a start or an end index using a softmax function. Specifically, p^{start}, $p^{end} \in R^n$ indicate the vectors that show the likelihood in probability of each token being the start index and end index, respectively:

$$[p^{start}, p^{end}] = softmax\ (EW + B),$$

where both W and $B \in R^{n^2}$ are trainable parameters.

Then, a ranked list of potential Product Information (PI) along with corresponding confidence scores is produce by the model. These scores are computed as the sum of the probabilities associated with their *start* and *end* tokens.

In training, the overall objective is to minimize the global loss of three types which are losses for start index, end index and start-end index matching. These losses are simultaneously trained in an end-to-end framework. We use [15] to optimize the loss.

3. Experiments

This section first shows the general information about the public benchmark dataset used for experiments. Then, it tells us about the setups of experiments. Finally, the experimental results and discussion are shown.

3.1. Dataset

In this paper, we used the dataset released by Tran and Luong [1] to perform comparative experiments. This data was collected from a history log of a retail restaurant, some forums and social websites. It was annotated with seven main types of PI which are *product category, product attribute, product extra-attribute, product brand, product packsizes, product number,* and *product uoms*. Two levels of annotation were provided. At the first level, descriptions of products (Level 1) are extracted. Then, these product descriptions are further decomposed into some detailed PI types (Level 2). An example is given in *Table 2.*

3.2. Experimental setups

The models are evaluated using popular metrics such as precision, recall, and F1 scores [17]. The best parameters were fine-tuned on development sets. The best values for parameters and hyper-parameters are listed as follows:

- Train sequence length: 768
- Number of epochs: 300
- Batch size: 8
- Learning rate: 3e-5
- Adam epsilon: 1e-8
- Max gradient norm: 1.0
- Bert embedding: 768 dimensions.

Table 2.

One example of a user's statement annotated at two levels.

(English translation is provided right after the Vietnamese statement at the first row)

Utterances	*Cho em đặt*	1	*hộp*	*bánh kem*	*vị xoài*	*Cỡ lớn*
	Let me order	1	*pack*	*cream cake*	*mango flavor*	*big size*
Level 1	other	Product description				
Level 2	other	number	uom	category	attribute	size

We adapted the MRC framework[3] to this task and exploited the viBERT[4] and vELECTRA[5], a pre-trained large language model optimized for Vietnamese, to build the PI recognition model. In case the pre-trained models are not available to optimize for a specific language, it is also feasible to use a multilingual pre-trained model, such as mBERT (a.k.a multi-lingual BERT) in order to get the vector representations for its sentences. We trained the model on the GPU Tesla V100 SXM2 32GB.

3.3. Experimental results

Tables 3 and *4* show the experimental results of the proposed model in comparison to the two baselines which are BiLSTM-CRF and CNN-CRF [1].

At Level 1, we can see that the MRC approach boosted the performance by a large margin on all evaluation metrics. In comparison to the best baseline CNN-CRF, it enhanced F1 score by nearly 2% in the case of using viBERT and 2.4% in the case of using vELECTRA. This suggested that the MRC approach is very promising and yields a better performance than other traditional approaches.

Table 3.

Experimental results of the models at Level 1 – Product descriptions

	Precision	**Recall**	**F1-scores**
biLSTM-CRF	89.71	91.35	90.52
CNN-CRF	90.6	91.24	90.91
MRC-viBERT	94.1	91.68	92.87
MRC-vELECTRA	**94.5**	**92.18**	**93.33**

At Level 2, in comparison to biLSTM-CRF, it significantly outperformed this baseline in all product information types. The MRC-viBERT approach also slightly increased the F1 score by about 0.3% in comparison the best baseline of CNN-CRF method. Among seven PI types, it achieved a significant improvement over the two baselines by a large margin on three PI types (i.e. *product branch, product category*, and *product extra_attribute*). For the type of attribute, the proposed approach got the competitive results.

[3] https://github.com/CongSun-dlut/BioBERT-MRC

[4] https://github.com/fpt-corp/viBERT

[5] https://github.com/fpt-corp/vELECTRA

Table 4.

Experimental results of the models at Level 2 – Product Information Types

	biLSTM-CRF			CNN-CRF			MRC-viBERT			MRC-vELECTRA		
PI types	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**	**Pre**	**Rec**	**F1**
attribute	93.69	95.63	94.63	95.9	97.24	95.8	95.82	95.25	95.53	96.02	95.71	**95.86**
brand	82.44	83.24	82.77	89.38	88.64	88.98	92.04	89.90	90.90	92.79	90.65	**91.71**
category	86.24	88.45	87.32	91.44	91.90	91.67	93.57	93.88	93.72	94.17	93.98	**94.07**
extra attribute	87.89	86.76	87.26	88.83	86.24	87.39	94.03	88.76	91.31	95.01	89.04	**91.93**
packsize	85.03	86.82	85.84	91.62	93.14	**92.36**	92.23	88.77	90.41	93.04	89.21	91.08
sys number	95.24	95.35	95.28	95.88	95.92	**95.89**	95.29	92.04	93.62	96.12	92.57	94.31
uom	88.80	91.73	90.16	92.12	92.33	**92.19**	89.16	93.07	91.05	90.01	93.11	91.53
Total	89.39	90.86	90.11	92.95	93.21	93.08	93.69	93.01	93.34	94.11	93.76	**93.93**

It surpassed biLSTM-CRF, but could not overcome CNN-CRF on the remaining three PI types (i.e., *product packsize, product sys_number,* and *product uom*).

Among two types of MRC basing on viBERT and vELECTRA as backbone, we witnessed that MRC-vELECTRA performed slightly better than MRC-viBERT. It increased the performance on four PI types (i.e. *product attribute, product branch, product category*, and *product extra_attribute*). However, similar to MRC-viBERT, the MRC-vELECTRA also could not surpass CNN-CRF on the remaining three PI types. Overall, in comparison to the best baseline – CNN-CRF, the MRC-vELECTRA increased the F1 score by 0.85%. This result is quite promising.

3.4. Discussion

Looking at the results shown in *Table 3* and *Table 4*, we acknowledge that using the MRC approach yielded higher F1 scores at both levels. This is because the queries/questions generated provide more prior knowledge to guide the identification process of product information.

It can be also seen that the proposed method yielded better performance on recognizing long PI types (such as *product attributes, product description, product extra attribute*) in comparison to the best baseline – CNN-CRF. This can be explained as follows: the MRC approach captures the sequence information better than CNN. CNN only leverages the local contexts based on n-gram characters and word embeddings. So, it does not have the power of capturing long PI types as compared to the MRC approach. Among two types of word embeddings, the MRC-vELECTRA was slightly better than MRC-viBERT on both two PI levels.

This proposed approach can be generalized to any language. In case BERT is not available to a specific language, we can instead use the mBERT (multi-linguage BERT) as the backbone.

Conclusion

This paper described the task of identifying product information mentioned by customers' statements in a retail domain. This is a vital step in developing many artificial intelligence commercial products. In contrast to many previous studies, we did not formulate the task as a conventional sequence labeling problem. Instead, we make use of the robustness of MRC tasks to propose an alternative approach. The proposed MRC architecture also leverages the knowledge gained during pre-training a large language model and then applies it to a new, related task – MRC. We performed experiments on a Vietnamese public benchmark dataset to verify the effectiveness of the proposed method. We achieved a new SOTA result by boosting the recognition performance over the two strong baselines. Specifically, we achieved 93.33% in the F1 score on recognizing product descriptions at Level 1 (upgraded by 2.4%). At Level 2, the model slightly improved the performance and yielded 93.93% in the F1 score on recognizing each product information type by using MRC-vELECTRA. The results also suggested that this approach is more effective in predicting long PI types with high precision.

In the future, we will continue exploring different kinds of generating questions by providing more clues to help find the product information. Furthermore, we will explore alternative robust pre-trained language models to improve the predictive model. ■

Acknowledgements

This paper was funded by the International School, Vietnam National University Hanoi under the project CS.NNC/2021-07.

References

1. Tran O.T., Luong T.C. (2020) Understanding what the users say in chatbots: A case study for the Vietnamese language. *Engineering Applications of Artificial Intelligence*, vol. 87, 103322. https://doi.org/10.1016/j.engappai.2019.103322
2. Tran O.T., Bui V.T. (2020) A BERT-based hierarchical model for Vietnamese aspect based sentiment analysis. Proceedings of the *12th International Conference on Knowledge and System Engineering (KSE), Can Tho, Vietnam, 2–14 November 2020*, pp. 269–274. https://doi.org/10.1109/KSE50997.2020.9287650
3. Bui V.T., Tran O.T., Le H.P. (2020) Improving sequence tagging for Vietnamese text using transformer-based neural models. *arXiv:2006.15994.* https://doi.org/10.48550/arXiv.2006.15994
4. Lample G., Ballesteros M., Subramanian S., Kawakami K., Dyer C. (2016) Neural architectures for named entity recognition. Proceedings of the *2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego, California, June 2016*, pp. 260–270. https://doi.org/10.18653/v1/N16-1030
5. Levy O., Seo M., Choi E., Zettlemoyer L. (2017) Zero-shot relation extraction via reading comprehension. *arXiv:1706.04115.* https://doi.org/10.48550/arXiv.1706.04115
6. McCann B., Keskar N.S., Xiong C., Socher R. (2018) The natural language decathlon: Multitask learning as question answering. *arXiv:1806.08730.* https://doi.org/10.48550/arXiv.1806.08730
7. Li X., Yin F., Sun Z., Li X., Yuan A., Chai D., Zhou M., Li J. (2019) Entity-relation extraction as multi-turn question answering. Proceedings of the *57th Conference of the Association for Computational Linguistics, Florence, Italy, July 2019*, pp. 1340–1350. https://doi.org/10.18653/v1/P19-1129
8. Ji Z., Lu Z., Li H. (2014) An information retrieval approach to short text conversation. *arXiv:1408.6988.* https://doi.org/10.48550/arXiv.1408.6988
9. Qiu M., Li F., Wang S., Gao X., Chen Y., Zhao W., Chen H., Huang J., Chu W. (2017) AliMe chat: A sequence to sequence and rerank based chatbot engine. Proceedings of the *55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Vancouver, Canada, July 2017*, pp. 498–503. https://doi.org/10.18653/v1/P17-2079

10. Goncharova E., Ilvovsky D.I., Galitsky B. (2021) Concept-based chatbot for interactive query refinement in product search. Proceedings of the *9th International Workshop "What can FCA do for Artificial Intelligence?" (FCA4AI 2021)*, vol. 2972, CEUR-WS, pp. 51–58. Available at: http://ceur-ws.org/Vol-2972/paper5.pdf (accessed 15 February 2024).

11. Yan Z., Duan N., Chen P., Zhou M., Zhou J., Li Z. (2017) Building task-oriented dialogue systems for online shopping. Proceedings of the *AAAI Conference on Artificial Intelligence*, vol. 31, no. 1. https://doi.org/10.1609/aaai.v31i1.11182

12. Liu Y., Ott M., Goyal N., Du J., Joshi M., Chen D., Levy O., Lewis M., Zettlemoyer L., Stoyanov V. (2019) RoBERTa: A robustly optimized BERT pretraining approach. *arXiv:1907.11692.* https://doi.org/10.48550/arXiv.1907.11692

13. Yang Z., Dai Z., Yang Y., Carbonell J., Salakhutdinov R.R., Le Q.V. (2019) XLNet: Generalized autoregressive pretraining for language understanding. *33rd Conference on Neural Information Processing Systems (NeurIPS 2019), Vancouver, Canada, arXiv:1906.08237.* https://doi.org/10.48550/arXiv.1906.08237

14. Vaswani A., Shazeer N., Parmar N., Uszkoreit J., Jones L., Gomez A.N., Kaiser Ł., Polosukhin I. (2017) Attention is all you need. *31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA, arXiv:1706.03762.* https://doi.org/10.48550/arXiv.1706.03762

15. Kingma J., Ba J. (2015) Adam: A method for stochastic optimization. *arXiv:1412.6980.* https://doi.org/10.48550/arXiv.1412.6980

16. Devlin J., Chang M.W., Lee K., Toutanova K. (2018) Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv:1810.04805.* https://doi.org/10.48550/arXiv.1810.04805

17. Therasa M., Mathivanan G. (2022) Survey of machine reading comprehension models and its evaluation metrics. Proceedings of the *6th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India*, pp. 1006–1013. https://doi.org/10.1109/ICCMC53470.2022.9754070

About the authors

Tho Chi Luong

Researcher, Institut Francophone International, Vietnam National University, Hanoi, E5, 144 Xuan Thuy St., Cau Giay Dist., Hanoi, Vietnam;

E-mail: tholc@vnu.edu.vn

ORCID: 0000-0002-7664-705X

Oanh Thi Tran

Associate Professor, PhD;

Lecturer, International School, Vietnam National University, Hanoi, G7, 144 Xuan Thuy St., Cau Giay Dist., Hanoi, Vietnam;

E-mail: oanhtt@gmail.com

ORCID: 0000-0002-3286-3623