Advanced Joint Model for Vietnamese Intent Detection and Slot Tagging

Nguyen Thi Thu Trang[1(✉)], Dang Trung Duc Anh[1], Vu Quoc Viet[1], and Park Woomyoung[2]

[1] School of Information and Communication Technology, Hanoi University of Science and Technology, Hanoi, Vietnam
trangntt@soict.hust.edu.vn
[2] Naver Corporation, Gyeonggi, South Korea
max.park@navercorp.com

Abstract. This paper aims to propose BiJoint-BERT-NLU, an advanced BERT-based joint model for Vietnamese intent detection and slot tagging, which extends the state-of-the-art JointBERT-CRF model. This model leverages the bi-directional relationships of these two tasks by: (i) adopting an intent-slot attention layer to explicitly incorporate the simple intent output (but with a temporary intent loss) into slot tagging (with a slot tagging loss) from the JointIDSF model, and (ii) introducing an advanced intent classification layer (with a final intent loss) that uses the slot tagging results to improve the accuracy of intent classification. The slot tagging outputs of all tokens, i.e. slot probability, will be summed up for each slot to build the final slot vector for the intent classifier. During the training phase, the coefficients of the three losses are optimized by grid search. The experiments have been done on the recently (and only) published PhoATIS dataset, the Vietnamese version of ATIS. The experimental results show that the proposed model using PhoBERT encoder on word-level on the syllable-level variant of the dataset gives a significant enhancement of Intent accuracy compared to state-of-the-art baseline models, i.e. JointBERT-CRF and JointIDSF. The Sentence accuracy has a considerable improvement on both syllable-level (using XLM-R encoder) and word-level variant.

Keywords: Vietnamese · NLU · BiJoint-BERT-NLU · Intent classification · Slot tagging

1 Introduction

Dialog systems, such as virtual assistant, chatbot systems, etc. have been increasingly become more and more popular. Natural language understanding (NLU) is one of the most important components in such systems. The two main tasks of this component are intent classification and slot tagging. The goal of intent classification is to understand what users want by classifying users' intents from a given utterance into a given set of intents. Whereas, slot tagging is a sequence

N.-S. Vo et al. (Eds.): INISCOM 2022, LNICST 444, pp. 125–135, 2022.
https://doi.org/10.1007/978-3-031-08878-0_9

labeling task, which aims to tag each token in utterance to correct a defined tag to extract semantic concepts.

Several previous works solved these two tasks independently using modern deep learning methods and achieved high accuracy. For text classification, in both [7] and [20], a Convolutional Neural Network (CNN) is used. Long Short-Term Memory (LSTM) [6] and recently Transformer [16] were proposed to classify the intent of the input sentence. For slot tagging task, in [19], the author used a regression model on top of an LSTM, whereas the work in [8] introduced one LSTM to extract contextual information and then use another LSTM for sequence tagging. These models achieve 95.08% and 95.47% slot F1-score on the ATIS dataset respectively.

Recent research has shown that jointly learning these two tasks helps to boost the performance since information used to solve one task can be useful for the other. Models which learn to solve both of these problems simultaneously are called joint models. [18] designs new Bi-model based Recurrent Neural Network (RNN) semantic frame parsing network structures to consider cross-impact between two tasks. Currently, this is the best performing model on the ATIS dataset with 98.99% intent accuracy and 96.89% slot F1-score. Moreover, some research use pre-trained language models that are trained on a large unlabeled corpus such as BERT [5] to deal with the difficulty of lacking labeled data. [3] proposed a simple joint model based on BERT, on which [14] based on to introduce a Stack-Propagation framework, which can better incorporate the intent semantic information to guide slot tagging and make the model more interpretable.

Vietnamese is a low-resource language in these research topics. Some research done on private Vietnamese datasets made predictions on intent and slots separately [13,15] using LSTM and Conditional Random Field (CRF). Recently, PhoATIS dataset [4] has been publicly available for intent classification and slot tagging. This dataset was translated into Vietnamese from the ATIS dataset, a popular dataset for this field. The joint model that was introduced along with the PhoATIS, i.e. JointIDSF model extending the work of [3], extracted features using a pre-trained language model and used an intent-slot attention layer to explicitly incorporate intent context information into slot tagging.

However, the intent accuracy of the JointIDSF model was not significantly improved over its baseline model (JointBERT-CRF). In JointIDSF, only the intent context was used to enhance the quality of slot tagging. In this paper, we propose a joint model, called BiJoint-BERT-NLU, that leverages the bi-directional relationships of these two tasks. In this model, beside the intent-slot attention layer to explicitly incorporate intent context information into slot tagging, we build an advanced intent classification layer that leverages the tagging slot outputs to improve the intent accuracy of usersays.

The rest of the paper is organized as follows. Section 2 presents the proposed joint model along with the baseline one. Section 3 goes in depth into the related works done in this field of research. In Sect. 4, the experiments and evaluation results on PhoATIS will be reported and discussed with state-of-the-art baseline models. Finally, the paper draws some conclusions of the work in Sect. 5.

2 Our Model

In this section, we present our proposed model with extensions from the baseline model JointBERT-CRF [3].

2.1 Baseline Model

The baseline model [3] uses BERT encoder followed by a feed-forward layer for Intent classification and a Conditional Random Field layer for Slot tagging (Fig. 1). These 2 tasks are performed separately in this model.

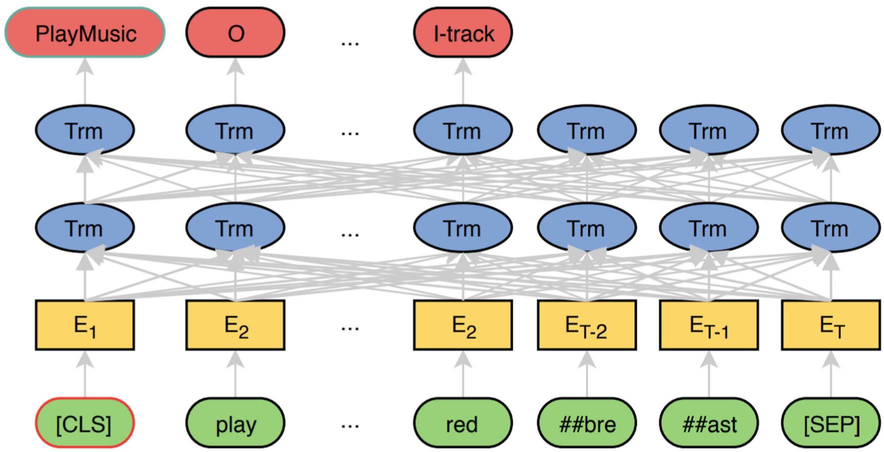

Fig. 1. Architecture of the baseline joint model for intent classification and slot tagging

Given an input sentence with n tokens $x_1...x_n$, a special classification token [CLS] is inserted as the first token of the sequence, resulting in the sequence $x_0, x_1...x_n$ (x_0 corresponds to the [CLS] token). The sequence is then passed to the BERT encoder to form a contextualized embedding $h_0, h_1...h_n$ with h_i corresponds to token x_i.

Intent Classification. The contextualized embedding h_0 of [CLS] is then fed into a single-layer feed-forward neural network (FFNN_{IC}), whose output size is the number of intent labels, to create the probability vector y^i

$$y^i = softmax(FFNN_{\text{IC}}(h_0))$$

Slot Tagging. The contextualized embedding for the other tokens $(h_1...h_n)$ is fed to another feed-forward network ($\mathrm{FFNN_{ST}}$) with the output size, i.e. the number of slot labels.

$$y_j^s = softmax(FFNN_{\mathrm{ST}}(h_j))$$

The output is then fed into a linear-chain CRF layer to predict slot types of the tokens.

The objective of the model is to maximize the probability $p(y^i, y^s|x)$. The model is trained by minimizing the cross-entropy loss.

$$p(y^i, y^s|x) = p(y^i|x)\prod_{j=1}^{n}(p(y_j^s|x))$$

2.2 Proposed Model: BiJoint-BERT-NLU

In this paper, we propose a new model, called BiJoint-BERT-NLU, for the intent classification and slot tagging tasks for Vietnamese, illustrated in Fig. 2. In this model, we introduce mechanisms to incorporate information from one task to help with the performance of the other.

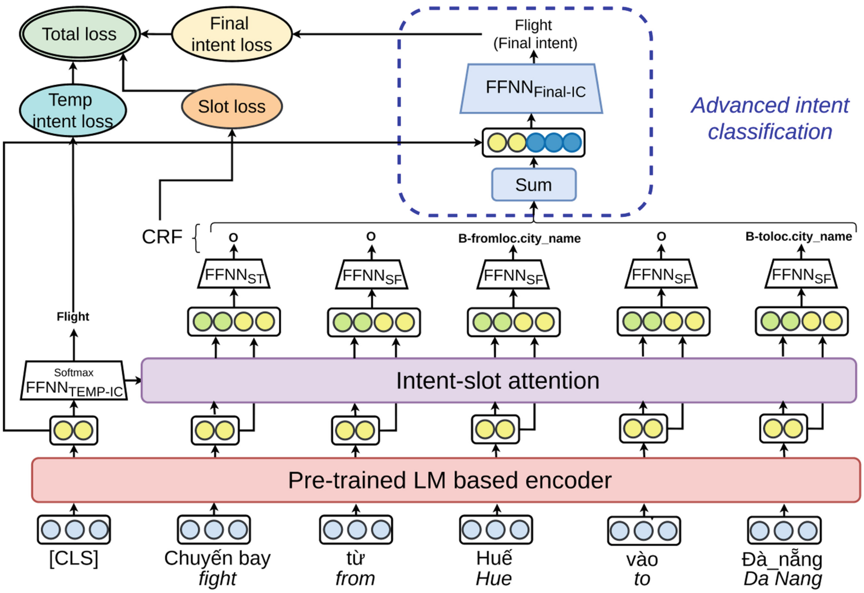

Fig. 2. Architecture of the proposed joint model for intent classification and slot tagging

Intent-Slot Attention Layer. We adopt the Intent-slot attention proposed by [4] to incorporate intent context information into slot tagging to enhance the performance. We also consider this model as the second baseline model, an extension of the original baseline model presented in Sect. 2.1. However, in this paper, the intent prediction used in this layer is just temporary and only used to provide more context for slot tagging. In other words, this prediction is not part of the output of the model. We compute a separate cross-entropy loss for the intent classification and call it Temporary Intent Loss ($\mathcal{L}_{\text{Temp-IC}}$).

This layer creates an intent label embedding by multiplying the intent probability vector y^i with a weight matrix W. The intent label embedding is then used together with the contextualized embedding h_i to create intent-specific vector s_i (with $i \in 1, 2, ...n$)

$$w = Wy^i$$

$$\alpha_i = \frac{exp(w^T h_i)}{\sum_{j=1}^{n} exp(w^T h_i)}$$

$$s_i = \alpha_i w$$

The contextualized embedding of each token ($h_1 ... h_n$) is then concatenated with the intent-specific vector s_i. The newly created vector is then passed to the feed-forward layer and CRF to produce the slot type prediction

$$v_i = s_i \circ h_i$$

$$y_j^s = softmax(FFNN_{\text{ST}}(v_j))$$

Advanced Intent Classification Layer. This component's idea is to use the output of the slot tagging layer to help better improve the accuracy of intent classification. We do not use the previously computed intent vector because that prediction does not take account for the slot types of tokens. The output prediction for slot tagging (y_j^s with $j \in 1, 2...n$) is summed up into a single vector. Note that the vectors being added are probability vectors of tokens presented in the input sequence. We use summation to firstly keep the dimension of the vector unchanged and secondly, adding probability vectors will retain information about the frequency of each slot type in the given utterance. This vector is then concatenated with h_0 and then passed into a feed-forward network to produce the final intent prediction. The output of this layer will be the actual intent the model predicts and counted towards intent accuracy.

$$q = \sum_{j=1}^{n} y_j^s$$

$$y_{\text{final}}^i = softmax(FFNN_{\text{final-IC}}(q \circ h_0))$$

The Final Intent Loss ($\mathcal{L}_{\text{Final-IC}}$) is then computed for this advanced intent classification layer. The output of this layer will be the final decision for the intent classification task although both cross-entropy losses are calculated for the temporary and final intent classification layers.

Joint Training. The cross-entropy losses of the temporary intent classification, the slot tagging and the advanced intent classification are denoted as $\mathcal{L}_{\text{Temp-IC}}$, $\mathcal{L}_{\text{ST}}$, $\mathcal{L}_{\text{Final-IC}}$ respectively. Although the result of temporary intent classification result is not used for the output of intent classification task, the $\mathcal{L}_{\text{Temp-IC}}$ is still calculated as the cross-entropy loss of the temporary intent prediction for slot tagging and contributes to the total loss ($\mathcal{L}_{\text{Total}}$). The final training objective, i.e. Total Loss, is the weighted sum of these losses as follows:

$$\mathcal{L}_{\text{Total}} = \lambda_{\text{Temp-IC}}\mathcal{L}_{\text{Temp-IC}} + \lambda_{ST}\mathcal{L}_{\text{ST}} + \lambda_{\text{Final-IC}}\mathcal{L}_{\text{final-IC}}$$

These three weight coefficients ($\lambda_{\text{Temp-IC}}$, λ_{ST} and $\lambda_{\text{Final-IC}}$) can be tuned as other hyperparameters of the model.

3 Related Works

BERT. Word embedding has been widely used in many natural language processing tasks, it helps with capturing the semantic meaning of words using a relatively low dimension vector. Since these word embeddings are trained on a very large corpus, this feature representation improves the performance on a variety of NLP tasks, especially when the dataset is small. BERT (Bidirectional Encoder Representations from Transformers) is by far one of the most powerful and most popular pretrained language representation models. Unlike previous language models such as GloVe [12] or FastText [1], BERT can learn contextual relations between words, meaning the same word can have different vector representations in different sentences. This makes BERT exceptional when used to train on down-stream tasks. For that reason, many BERT-based language models are introduced such as RoBERTa [9] and its multilingual variant and Vietnamese monolingual variant, XLM-R and PhoBERT.

Joint Model. Recent research shows that constructing and training joint models that handle multiple tasks simultaneously increases the performance significantly, compared to doing so independently. For intent classification and slot tagging tasks, the joint model aims to capture the joint distributions of intent and slot labels, with respect also to the words in the utterance, their local context, and the global context in the sentence. Recurrent neural networks (RNNs) have been frequently used in the field. In more recent years, the transformer architecture has become more and more prominent because of its ability to capture long range dependency. Following that is the task-specific component. For example, the Diet architecture [2], after the input vectors are fed into transformer encoder, the contextualized embedding will be fed into a CRF layer to predict slot types of tokens and similarity block is used to classify the intent label, and these two components work independently. However, in [4], the author feeds the intent prediction to the slot tagging layer instead. In our proposed model BiJoint-BERT-NLU, the prediction of one task is partly determined by that of the other.

4 Experiments

In this section, we did the experiments on our proposed model, i.e. BiJoint-BERT-NLU, and on the two baseline ones, i.e. (i) JointBERT-CRF by [3] (Joint BERT model with a CRF layer for slot tagging) and (ii) JointIDSF by [4] (the extension of JointBERT-CRF with an intent-slot attention layer).

4.1 Datasets

To evaluate the effectiveness of the proposed method, we use the PhoATIS dataset (Table 1), the only public dataset for Vietnamese. This dataset is a Vietnamese version of ATIS, a popular dataset for this field of research. The dataset was created through three steps according to the original paper. The first step is to manually translate the data into Vietnamese (this is done by one NLP researcher and two research engineers that achieve 7.0+ IELTS score). The second one is to project intent and slot annotations into the translation. The final step is to correct the inconsistencies that appear in the dataset. Since the utterances of PhoATIS are annotated at syllable level (in Vietnamese, white space is not only used as border between words but it's also used as border between syllables that compose words), in the original paper, the word-level variant is obtained through using RDRSegmaneter [11] from VnCoreNLP [17].

Table 1. Statistics of PhoATIS dataset

	Train	Dev	Test
Intent #	24	17	21
Slot Types #	82	70	71
Utterance #	4,478	500	893

The training set, evaluation set and test set contains 4,478, 500 and 893 utterances respectively. The statistics of the PhoATIS dataset is shown by Table 1, which is computed on the dataset on the public github. In the original paper, the dataset is said to consist of 28 intent labels, but according to our observation on the public dataset, the training set (both syllable-level and word-level variants) has only 24 intent labels.

4.2 Experiment Preparation

The metrics that we use to measure the effectiveness of our methods are Intent accuracy, Slot F1-score and Sentence accuracy. With regards to Sentence accuracy, a sentence is said to be correctly classified if both the intent as well as all of the slots are correctly predicted.

For experimenting all models including the two baseline models and our proposed model, we use AdamW [10] with $\epsilon = 1e-8, \beta_1 = 0.9$ and $\beta_2 = 0.999$ as our optimizer and set the batch size to 32. All of the models are trained for 50 epochs using NVIDIA GeForce GTX 2080 Ti.

4.3 Experimental Results

We use grid search on the evaluation set to find the best loss weight coefficient combination for the proposed model, as illustrated in Table 2. To be specific, we select the best 3 combinations with the highest average of Intent accuracy and Slot F1-score then apply them to the test set. For each hyper-parameter combination, we train the model for 50 epochs and select the checkpoint achieving the highest average score over the evaluation set to apply to the test set.

Table 2. The result of the loss weight coefficients grid search with the loss weight coefficients correspond to $\mathcal{L}_{\mathrm{Temp-IC}}/\mathcal{L}_{\mathrm{ST}}/\mathcal{L}_{\mathrm{Final-IC}}$ respectively. The best combinations are chosen based on Average score of Intent acc and Slot F1 on evaluation set. Intent acc., Slot F1 and Sent acc. are results on test set

Loss weight co-effs ($\mathcal{L}_{\mathrm{Temp-IC}}/\mathcal{L}_{\mathrm{ST}}/\mathcal{L}_{\mathrm{Final-IC}}$)	Encoder	Average on dev (%)	Intent acc. (%)	Slot F1 (%)	Sent acc. (%)
0.2/0.4/0.4	XLM-R	97.12	97.42	**95.21**	86.45
0.4/0.3/0.3	XLM-R	97.11	97.42	94.74	85.89
0.2/0.3/0.5	XLM-R	97.05	**97.65**	95.19	**86.90**
0.4/0.4/0.2	PhoBert	97.30	97.65	**95.22**	86.67
0.3/0.6/0.1	PhoBert	97.22	97.65	94.76	86.23
0.2/0.3/0.5	PhoBert	97.19	**98.43**	95.09	**87.23**

Table 3. Results of the best performing model on PhoATIS test set compare to JointBERT-CRF and JointIDSF on syllable-level (XLM-R) or word-level (PhoBERT) variant of test set

Model	Encoder	Intent acc. (%)	Slot F1 (%)	Sent acc. (%)
Joint-Bert-CRF [3]	XLM-R	97.42	94.62	85.39
JointIDSF [4]	XLM-R	97.56	94.95	86.17
BiJoint-BERT-NLU (our model*)	XLM-R	**97.65**	**95.19**	**87.35**
Joint-Bert-CRF [3]	PhoBert	97.40	94.75	85.55
JointIDSF [4]	PhoBert	97.62	94.98	86.25
BiJoint-BERT-NLU (our model*)	PhoBert	**98.43**	**95.01**	**87.23**

We also initialized our models as well as the baseline using 2 different pretrained language models, in this case, we chose XLM-R and PhoBERT, to see the significance of impact the pretrained language model has on our architecture. When using XLM-R encoder, we conduct experiments on syllable-level variants of PhoATIS while when using PhoBERT, we do so on word-level. The results of this experiment are illustrated by Table 2. The model with the best performance on test set regarding Sentence accuracy for both XLM-R encoder and PhoBERT encoder has $\mathcal{L}_{\text{IC}} = 0.2/\mathcal{L}_{\text{ST}} = 0.3/\mathcal{L}_{\text{final-IC}} = 0.5$, achieving 86.90% and 87.23% respectively. This result will be used to compare with JointBERT-CRF and JointIDSF as illustrated in Table 3 and Fig. 3.

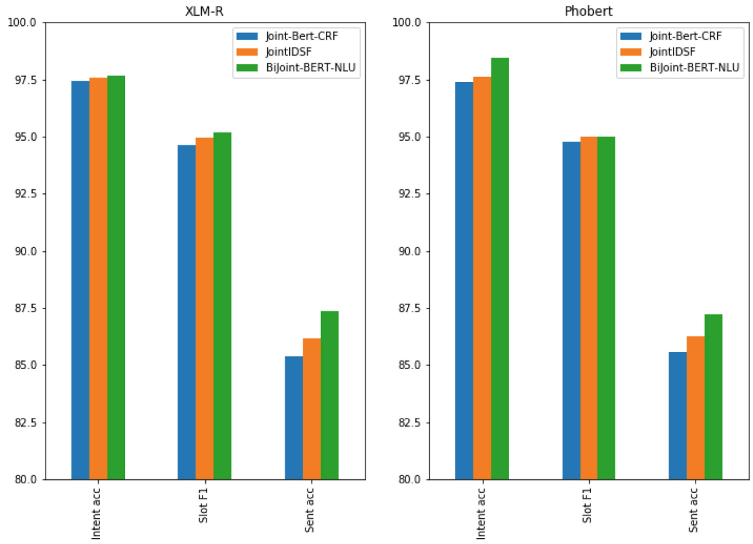

Fig. 3. Results of the best performing model on the PhoATIS test set compared to JointBERT-CRF and JointIDSF on syllable-level (XLM-R) and word-level (PhoBERT). The bar representing JointBERT-CRF, JointIDSF and BiJoint-BERT-NLU is colored blue, orange and green respectively. (Color figure online)

For each pretrained encoder, we compare our model (BiJoint-BERT-NLU) with JointBERT-CRF and JointIDSF. All three of these architectures extract semantic meaning from BERT-based language models (BERT, XLM-R), but JointBERT-CRF solves intent classification and slot tagging independently. As illustrated in Table 3 and Fig. 3, our proposed model achieves the best score on all three metrics: Intent accuracy, Slot F1 and Sentence accuracy. The intent accuracy of our model is higher by about 1% compared to both baseline models using the PhoBERT encoder (word-level variant of the dataset). The sentence accuracy of the proposed model is about 2% better than the JointBERT-CRF and about 1% better than the JointIDSF model on both syllable and word level. The slot F1 is slightly better than both baseline models. This result shows that the advanced intent classification layer has a great impact on the quality of intent classification and sentence accuracy, but does not degrade the quality of the slot tagging task.

5 Conclusion

In this paper, we introduce an advanced BERT-based joint model, i.e. BiJoint-BERT-NLU, to improve the quality of Vietnamese intent classification and slot tagging. Two baseline state-of-the-art models are: (i) JointBERT-CRF (not explicitly model the relationship between these two tasks) and (ii) JointIDSF (only explicitly utilize the output of intent classification to aid slot tagging). In the proposed model, we model the bi-directional relationships of the two tasks by: (i) adopting an intent-slot attention layer to explicitly incorporate the simple intent output (with a temporary intent loss) into slot tagging (with a slot tagging loss) from the JointIDSF model, and (ii) introducing an advanced intent classification layer (with a final intent loss) that uses the slot tagging results to improve the accuracy of intent classification. The slot tagging outputs of all tokens, i.e. slot probability, will be summed up for each slot to build the final slot vector for the intent classifier. Grid search is used to optimize coefficients of the three losses during the training phase. We did some experiments on the recently (and only) published PhoATIS dataset, the Vietnamese version of ATIS. The experimental results show that the proposed model BiJoint-BERT-NLU using PhoBERT encoder on word-level variant of the dataset gives a significant enhancement of Intent accuracy (about 1%) compared to state-of-the-art baseline models, i.e. JointBERT-CRF and JointIDSF. The Sentence accuracy has a considerable improvement, about 1% to 2%, on both syllable-level (using XLM-R encoder) and word-level variant. This result shows that the advanced intent classification layer has a great impact on the quality of intent classification and sentence accuracy.

Acknowledgments. This work is supported by Naver Corporation and School of Information and Communication Technology, Hanoi University of Science and Technology.

References

1. Bojanowski, P., Grave, E., Joulin, A., Mikolov, T.: Enriching word vectors with subword information. arXiv:1607.04606 [cs], June 2017. http://arxiv.org/abs/1607.04606
2. Bunk, T., Varshneya, D., Vlasov, V., Nichol, A.: DIET: lightweight language understanding for dialogue systems. arXiv:2004.09936 [cs], May 2020. http://arxiv.org/abs/2004.09936
3. Chen, Q., Zhuo, Z., Wang, W.: BERT for joint intent classification and slot filling. arXiv:1902.10909 [cs], February 2019. http://arxiv.org/abs/1902.10909
4. Dao, M.H., Truong, T.H., Nguyen, D.Q.: Intent detection and slot filling for vietnamese. arXiv:2104.02021 [cs], June 2021. http://arxiv.org/abs/2104.02021
5. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of deep bidirectional transformers for language understanding. arXiv:1810.04805 [cs], May 2019. http://arxiv.org/abs/1810.04805
6. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. **9**(8), 1735–1780 (1997). https://doi.org/10.1162/neco.1997.9.8.1735

7. Kim, Y.: Convolutional neural networks for sentence classification. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1746–1751. Association for Computational Linguistics, Doha, Qatar, October 2014. https://doi.org/10.3115/v1/D14-1181, https://aclanthology.org/D14-1181
8. Kurata, G., Xiang, B., Zhou, B., Yu, M.: Leveraging sentence-level information with encoder LSTM for semantic slot filling. arXiv:1601.01530 [cs], August 2016. http://arxiv.org/abs/1601.01530
9. Liu, Y., et al.: RoBERTa: a robustly optimized BERT pretraining approach. arXiv:1907.11692 [cs], July 2019. http://arxiv.org/abs/1907.11692
10. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv:1711.05101 [cs, math], January 2019. http://arxiv.org/abs/1711.05101
11. Nguyen, D.Q., Nguyen, D.Q., Vu, T., Dras, M., Johnson, M.: A fast and accurate vietnamese word Segmenter. arXiv:1709.06307 [cs], December 2017. http://arxiv.org/abs/1709.06307
12. Pennington, J., Socher, R., Manning, C.: GloVe: global vectors for word representation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1532–1543. Association for Computational Linguistics, Doha, Qatar, October 2014. https://doi.org/10.3115/v1/D14-1162, https://aclanthology.org/D14-1162
13. Pham, D., Le, H., Quan, T.: Transfer learning for a Vietnamese dialogue system. In: 2019 11th International Conference on Knowledge and Systems Engineering (KSE), pp. 1–5, October 2019. https://doi.org/10.1109/KSE.2019.8919425. ISSN: 2164-2508
14. Qin, L., Che, W., Li, Y., Wen, H., Liu, T.: A stack-propagation framework with token-level intent detection for spoken language understanding. arXiv:1909.02188 [cs], September 2019. http://arxiv.org/abs/1909.02188
15. Trang, N.T.T., Ky, N.H., Son, H., Hung, N.T., Huan, N.D.: Natural language understanding in Smartdialog: a platform for Vietnamese intelligent interactions. In: Proceedings of the 2019 3rd International Conference on Natural Language Processing and Information Retrieval, pp. 158–163. NLPIR 2019, Association for Computing Machinery, New York, NY, USA, June 2019. https://doi.org/10.1145/3342827.3342857
16. Vaswani, A., et al.: Attention is all you need. arXiv:1706.03762 [cs], December 2017. http://arxiv.org/abs/1706.03762
17. Vu, T., Nguyen, D.Q., Nguyen, D.Q., Dras, M., Johnson, M.: VnCoreNLP: A Vietnamese natural language processing toolkit. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations, pp. 56–60 (2018). https://doi.org/10.18653/v1/N18-5012, http://arxiv.org/abs/1801.01331, arXiv: 1801.01331
18. Wang, Y., Shen, Y., Jin, H.: A Bi-model based RNN semantic frame parsing model for intent detection and slot filling. arXiv:1812.10235 [cs], December 2018. http://arxiv.org/abs/1812.10235
19. Yao, K., Peng, B., Zhang, Y., Yu, D., Zweig, G., Shi, Y.: Spoken language understanding using long short-term memory neural networks. In: 2014 IEEE Spoken Language Technology Workshop (SLT), pp. 189–194, December 2014. https://doi.org/10.1109/SLT.2014.7078572
20. Zhang, X., Zhao, J., LeCun, Y.: Character-level convolutional networks for text classification. Adv. Neural Inf. Process. Syst. **28** (2015), https://proceedings.neurips.cc/paper/2015/hash/250cf8b51c773f3f8dc8b4be867a9a02-Abstract.html