The Impacts of the Contextual Substitutions in Vietnamese Micro-text Augmentation

Huu-Thanh Duong[1(✉)] and Trung-Kiet Tran[2]

[1] Faculty of Information Technology, Ho Chi Minh City Open University, Ho Chi Minh City, Vietnam
thanh.dh@ou.edu.vn

[2] Department of Fundamental Studies, Ho Chi Minh City Open University, Ho Chi Minh City, Vietnam
kiet.tt@ou.edu.vn

Abstract. The deep learning models rely on a huge amount of annotated training data to learn multiple layers of the features or representations and also avoid overfitting. However, the annotated dataset is unavailable, especially for the low resource languages. Building them is a tedious, time-consuming and expensive task. Thus, data augmentation has been mentioned as a perfect approach to generate the annotated data from the limited data without user intervention. In this paper, we evaluate the importances and the impacts of the contextual words to enhance the training data based on a pre-trained model which we build based on the reviews extracting the e-commerce websites in Vietnamese. We experiment on the sentiment analysis problem to evaluate the effectiveness of our approach.

Keywords: Data augmentation · Deep learning · Sentiment analysis · Contextual substitution

1 Introduction

Data augmentation (DA) techniques first have been used in image classification to enrich the image training data such as rotate, scale, translate, noise, image mixup, etc. They are gradually equipped in natural language processing (NLP) for the semi-supervised learning approach and deep learning models with the limited training data, especially suitable for the low-resource languages as Vietnamese in order with overfitting avoidance.

Deep learning (DL) has emerged as a spotlight, has been used widely and achieved state-of-the-art performance in computer vision and NLP by learning multiple layers of features or representations of the data. Actually, deep learning has been developed from artificial neural networks appearing since the 1990s.

P. Cong Vinh and N. Huu Nhan (Eds.): ICTCC 2021, LNICST 408, pp. 32–39, 2021.
https://doi.org/10.1007/978-3-030-92942-8_3

In that period, the neural network training has only one or two layers (shallow neural networks) as if having more layers, the training is more complicated and expensive in computation. Based on Zhang et al. [1], the rapid growth of deep learning (deep neural networks) in recent years is as: (1) the availability of computing power due to the advances in hardware, (2) the availability of hug amounts of training data and (3) the power and flexibility of learning intermediate representations.

Therefore, one of the barriers of DL models relies on a large amount of the annotated training data, but these datasets are scarce, unavailable in a domain and collecting the annotated training dataset is time-consuming and expensive. DA is an effective solution to generate new training data being unchanged the labels and improves generalization of the DL models.

2 Related Works

There are many approaches proposed to augment training data such as lexicon substitution, sentence shuffling, generative text, back translation, syntax-tree transformation, word or sentence embeddings mixup.

Lexicon substitution techniques are the simple approaches such as shuffling the words or sentences of the original text, or swapping the sentences between the texts in the same label. Wei J et al. [3] proposed a simple approach to generate new data, including random replacement, random deletion, random insertion and random swap. Random replacement takes a random word from the original text and replaces it by a synonym based on the thesaurus, Random deletion removes a random word in the text with a probability p, Random Insertion takes a random word not being a stop word and gets one of its synonym words to insert into a random position in the text, Random Swap swaps randomly two words in the text. These techniques are simple, easy to implement, no need for any other resources and actually improve the accuracy of the algorithms in practice although they might generate meaningless sentences.

Back translation leverages translation machines to obtain new training data R. Sennrich et al. [9], A. Sugiyama et al. [10], M. Fadaee et al. [11]. Firstly, the text is translated from its original language into the intermediate language, then it will be back-translated into the original language to obtain new data which retains the meaning of the text. The translated text has never been exactly the same as the original text. This is easy to understand and implement, still retains the meaning of the original text, and achieves good performance. In [12] proves that this is an effective approach to generate new training data and improve the performance of the classifiers. The weakest point of this method is to need an effective translator.

Inspired by cropping and rotating techniques of data augmentation in image processing, syntax-tree transformation generates the dependency tree of the original text, then using some rules transforms it into the augmented text which retains the meaning, such as changing from active voice to passive voice. G. G. Sahin [13] used a dependency tree to remove dependency links and move the tree

fragments around the root. In [12], the authors also boosted the performances of the classifiers.

Word Embedding is a crucial tool for machine learning and DL algorithms to convert text into vectors. The words which are the same context will be near each other in vector space. Ideally, vectors of words have the same meaning being near each other. There are two approaches for word embedding as frequency-based embeddings and prediction based embeddings, we can use prediction-based embeddings to find and replace a word by its similar word. Word embedding-based substitution technique also replaces a word by other contextual words by using the pre-trained word embeddings such as Word2Vec, GloVe, Bert, etc. to select the most similar words in the embedding space. This approach reduces the cost to build the annotated data, thesaurus, wordnets, ontology, etc. However, being a black-box approach, some opposite words are near each other in vector space such as "tốt" (good) and "tệ" (bad), so it may affect the performances of some problems such as sentiment analysis.

K. Liu [4] based on the semantic similarity in the context in Word2Vec model to train synonym substitution list and replace the unknown words in the original words with synonyms in financial public opinion. S. Kobayashi [5] proposed the contextual data augmentation for the labeled sentences, this offers a list of substitute words predicted by a bidirectional language model according to the context. In the experiment, the author proved this solution improved the classifiers using CNN (Convolutional Neural Network) and RNN (Recurrent Neural Network). Furthermore, in embedding space can apply the mixup methods. The mixup technique is the latest approach by combining the word or sentence embeddings in the flow of the text classification. L. Sun et al. [7] and O. Kashefi et al. [8] proposed the mixup approaches and proved it significantly improved the performance of deep learning models.

In this paper, we approach the contextual substitutions to obtain new data. The rapid growth of web technologies has created a huge amount of data on the internet, mining them helps to resolve many problems effectively in machine learning, especially for low-resource languages. Our main contribution builds a pre-trained model for the Vietnamese reviews by collecting reviews from e-commerce websites and performs various experiments to evaluate the impacts of the contextual substitutions via deep learning models.

The rest of this paper is organized as follows: Sect. 3 presents the proposed approach, Sect. 4 shows the experimental results, the conclusions and future works are presented in Sect. 5.

3 The Approach

Word2Vec (W2V), which is one of the first models of Word Embedding, is a method to represent a vector of a word based on around words (contextual words). This uses a neural network having two layers (input and output) and one hidden layer. There are two models: Skip-gram and CBOW (Continuous Bag of Word).

The input of the skip-gram approach is the current word and the output is a word which has the most relationship to the current word. The vectors of input and output layers are similar to the number of dimensions and have a form of one-hot vectors. The number of dimensions of the hidden layer is equal to the predefined size (embedding size) and less than the size of input and output layers. Output layer uses a softmax activation function for every part of the vector. CBOW is the same as Skip-gram, but input is a list of words (contextual words) of the current word and output is the most related word of the current word.

We have built a pre-trained model in Vietnamese with W2V based on the data as reviews extracted from e-commerce websites, this is used to determine the contextual words of the replacing words in the text. The first approach replaces n words in the text randomly with the contextual words for new samples.

Furthermore, we use $tf \times idf$ scores to choose the replacing words instead of random choices. The words which have low $tf \times idf$ scores are uninformative, so replacing them usually does not affect the meaning of the original text. The original text is tokenized and calculated the $tf \times idf$ scores based on the list of reviews (comments) extracted from the well-known e-commerce websites in Vietnamese such as the gioididong.com, tiki.vn, shopee.vn, etc. In order to obtain new training data, we conduct n times the replacement of the words ordered by $tf \times idf$ scores in ascending order.

Data augmentation with mixup has first been proposed by Lei Zhang et al. [1] for image classification and shown an effective approach. They generate the synthetic samples through linearly interpolating a pair of input images and their corresponding labels randomly as the following equations, where x_i, x_j are the input vectors, y_i, y_j are the corresponding labels and the λ is the mixup-ratio having the value between [0, 1].

$$\hat{x} = \lambda x_i + (1 - \lambda)x_j \tag{1}$$

$$\hat{y} = \lambda y_i + (1 - \lambda)y_j \tag{2}$$

This approach is also equipped in NLP and achieves promising results (Hongyu Guo [6], L. Sun et al. [7]). The mixup is a cross-language approach based on embedding space, thus this is a potential approach for our problems. Our work injects the mixup layer to obtain the synthetic samples before feeding to the output layer to predict the results.

4 The Experiments

We use the datasets mentioned in [12] (dataset 1 (9280 positive, 6870 negative), dataset 2 (2380 strong positive, 2440 positive, 2380 negative), dataset 3 (15000 for each positive and negative), dataset 4 (5000 for each positive and negative)) and has gotten 100 samples of each polarity for training. Firstly, it conducts the preprocessing techniques into the datasets to remove some noise information, normalize and clean the text such as lowercasing, emojis substitution, negation

handling, intensification handling. We have also collected above 100 thousands of reviews from the e-commerce websites, this dataset is used to build the pre-trained model as Word2Vec and calculate the $tf \times idf$ weight of a word to determine for replacement.

The next experiment conducts n times for each sample to generate n new samples, each time chooses m words randomly and is replaced by their contextual words. This method generates 1000 new samples from 100 original samples of each polarity. In order to evaluate the impacts of the replacing words, we apply $tf \times idf$ weight to choose the replacing words in a sample. The $tf \times idf$-based substitution only replaces the uninformative words with the low $tf \times idf$ scores by the contextual word in W2V. The final experiments, these methods incorporate a mixup embedding approach in the mixup layers above the output layer.

The deep neural network includes two hidden layers (with 16 hidden units each) and one output layer, the hidden and output layers use the `relu` and `sigmoid` activation functions respectively. Since the output is a probability the loss function is `binary_crossentropy`. The training is performed for 20 epoches in mini-batches of 512 samples. Table 1 is the accuracies of our approaches with standard deviation of 10 runs, where the first column performs with the original data (the baseline result), the second column replaces randomly with the contextual words, *tfidf*-based substitution is presented in the third column.

Table 1. The mean of the accuracies with standard deviation in 10 runs: (1) The original data (baseline results), (2) Random substitution by the contextual words, (3) *tfidf*-based substitution by the contextual words, (4) Random substitution by the contextual words + mixup, (5) *tfidf*-based substitution by the contextual words + mixup.

Datasets	(1)	(2)	(3)	(4)	(5)
Dataset 1	0.7636 ± 0.050	0.8443 ± 0.004	0.8396 ± 0.012	0.8482 ± 0.0012	**0.8530 ± 0.0008**
Dataset 2	0.4148 ± 0.056	**0.5155 ± 0.011**	0.4492 ± 0.047	0.4031 ± 0.0020	0.4262 ± 0.0001
Dataset 3	0.7418 ± 0.054	0.7798 ± 0.003	0.7821 ± 0.005	**0.8066 ± 0.0002**	0.7983 ± 0.0003
Dataset 4	0.7567 ± 0.020	0.7817 ± 0.006	0.7861 ± 0.007	**0.7911 ± 0.0002**	0.7849 ± 0.0006

Figure 1 shows the visualization of the results, where the x axis is the experimental datasets, the y axis is the mean of the accuracies over 10 runs, each dataset runs all our experiments (see the caption of Fig. 1). The accuracy of the approaches overcomes the baseline result (the first columns of each dataset). The $tf \times idf$-based substitution is greater than random-based substitution (dataset 3 and dataset 4) so selecting good words to replace the contextual words is an essential point and affects the quality of the augmented data. However, other cases are in the opposite (dataset 1 and dataset 2). As earlier mentioned, word embeddings is a black-box approach, the opposite words may be near each other in embedding space such as "hài_lòng" (pleased) and "thất_vọng" (disappointed). In social reviews, some important words are low $tf \times idf$ score,

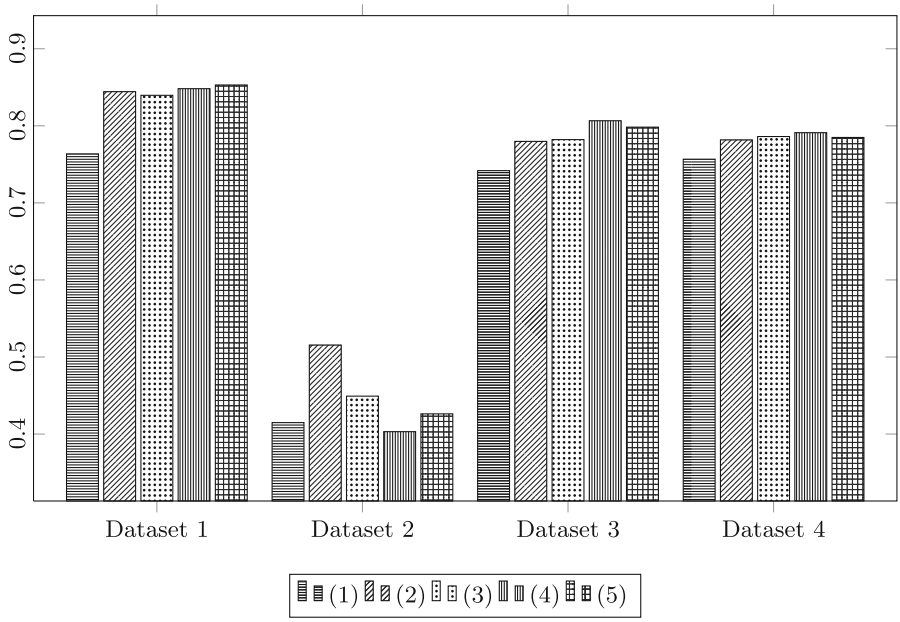

Fig. 1. The mean of the accuracies

for example the "hài_lòng" (pleased) word, which is commonly used in positive reviews, has low $tf \times idf$ score, so contextual words sometimes change the meaning of the text.

The accuracy of the contextual substitution which combines with the mixup embedding improves the performance of the datasets compared with only performing the contextual substitution. Although the accuracies decrease a little bit for dataset 2 (for two combinations) and dataset 4 (for combining with $tf \times idf$-based approach), the results of this combination are more stable with very minimum deviation for all datasets.

5 Conclusions and Future Works

This paper collects a set of reviews from well-known e-commerce websites and builds a pre-trained model for the contextual words. We conduct the various experiments to evaluate the effectiveness of the contextual substitution in generating new samples in Vietnamese. The experimental results show that selecting the word in the original texts and selecting a word in the contextual words for substitution affects the quality of the augmented data and the performances of the algorithms. The experiments have shown promising results, this utilizes the abundance of the internet data and can apply for cross languages, and saves cost to build the thesaurus or wordnets of the specific languages for the substitution approach.

In the future, we enhance more data to boost better quality and generalization of the pre-trained models and investigate novel methods to augment data, especially applying to Vietnamese.

Acknowledgment. I would like to thank the Center for Science and Technology Development Young - Thanh Doan City. HCM City (TST). This work is part of the "Vietnamese data augmentation for sentiment analysis based on deep learning" project supported by TST.

References

1. Zhang, L., Wang, S., Liu, B.: Deep learning for sentiment analysis: a survey. Wiley Interdisc. Rev.: Data Mining Knowl. Discovery **8**(4), e1253 (2018)
2. Han, S., Gao, J., Ciravegna, F.: Neural language model based training data augmentation for weakly supervised early rumor detection. In: Advances in Social Networks Analysis and Mining (ASONAM) 2019 IEEE/ACM International Conference on, pp. 105–112 (2019)
3. Wei, J., Zou, K.: EDA: Easy data augmentation techniques for boosting performance on text classification. In: ICLR 2019–7th International Conference on Learning Representations (2019)
4. Liu, K., Ergu, D., Cai, Y., Gong, B., Sheng, J.: A new approach to process the unknown words in financial public opinion. Proc. Comput. Sci. **162**, 523–531 (2019). https://doi.org/10.1016/j.procs.2019.12.019
5. Kobayashi, S.: Contextual augmentation: data augmentation by words with paradigmatic relations. In: Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol. 2 (Short Papers), New Orleans, Louisiana, pp. 452–457 (2018). https://doi.org/10.18653/v1/N18-2072
6. Guo, H., Mao, Y., Zhang, R.: Augmenting data with mixup for sentence classification: an empirical study. ArXiv, arXiv:1905.08941 (2019)
7. Sun, L., Xia, C., Yin, W., Liang, T., Yu, P., He, L.: Mixup-transformer: dynamic data augmentation for NLP Tasks. In: Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain (Online), pp. 3436–3440 (2020). https://doi.org/10.18653/v1/2020.coling-main.305
8. Kashefi, O., Hwa, R.: Quantifying the evaluation of heuristic methods for textual data augmentation. In: Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020), Online, pp. 200–208 (2020). https://doi.org/10.18653/v1/2020.wnut-1.26
9. Sennrich, R., Haddow, B., Birch, A.: Improving neural machine translation models with monolingual data. In: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Berlin, Germany, pp. 86–96 (2016). https://doi.org/10.18653/v1/P16-1009
10. Sugiyama, A., Yoshinaga, N.: Data augmentation using back-translation for context-aware neural machine translation. In: Proceedings of the Fourth Workshop on Discourse in Machine Translation (DiscoMT 2019), Hong Kong, China, pp. 35–44 (2019). https://doi.org/10.18653/v1/D19-6504
11. Fadaee, M., Bisazza, A., Monz, C.: Data augmentation for low-resource neural machine translation. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Vol. 2: Short Papers), Vancouver, Canada, pp. 567–573 (2017). https://doi.org/10.18653/v1/P17-2090

12. Duong, H.-T., Nguyen-Thi, T.-A.: A review: preprocessing techniques and data augmentation for sentiment analysis. Comput. Soc. Netw. **8**(1), 1–16 (2020). https://doi.org/10.1186/s40649-020-00080-x
13. Sahin, G.G., Steedman, M.: Data augmentation via dependency tree morphing for low-resource languages. In: Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, pp. 5004–5009 (2018). https://doi.org/10.18653/v1/D18-1545