Improve Neural Machine Translation by Building Word Vector with Part of Speech

Jinyingming Zhang[1] , Jin Liu[1, *] and Xinyue Lin[1]

Abstract: Neural Machine Translation (NMT) based system is an important technology for translation applications. However, there is plenty of rooms for the improvement of NMT. In the process of NMT, traditional word vector cannot distinguish the same words under different parts of speech (POS). Aiming to alleviate this problem, this paper proposed a new word vector training method based on POS feature. It can efficiently improve the quality of translation by adding POS feature to the training process of word vectors. In the experiments, we conducted extensive experiments to evaluate our methods. The experimental result shows that the proposed method is beneficial to improve the quality of translation from English into Chinese.

Keywords: Machine translation, parts of speech, word vector.

1 Introduction

Natural language processing is a comprehensive interdisciplinary subject integrating linguistics, mathematics, computer science and cognitive science. Machine Translation is the flagship of recent successes and advances in natural language processing (NLP). Its practical applications have spurred the interest in this topic.

Machine translation denotes the translation of text from one language into another by using computer technology. The translated language is called the source language, and the language which is the result of the translation is called the target language. Machine translation is the process that completing the conversion from the source language to the target language. In recent years, deep learning has developed rapidly, and machine translation based on artificial neural networks is gradually emerging. These trends make a rapid promotion in machine translation.

From early directly literal translation to today's neural network-based machine translation, many domestic and foreign scholars have done a lot of research on it. The traditional machine translation method requests linguists build the rules between source language and target language. However, this method requires a large number of rules to build a reliable translation system. Moreover, it also includes idioms, rare words, context and so on, which requires the professional knowledge. All of these cost enormous human resource and material resource.

[1] College of Information Engineering, Shanghai Maritime University, Shanghai, China.

[*] Corresponding Author: Jin Liu. Email: jinliu@shmtu.edu.cn.

Received: 06 March 2020; Accepted: 06 June 2020.

In 2016, Google used neural network to translate the languages instead of using the traditional statistical-based machine translation (SMT). Although the effect of neural network machine translation has surpassed SMT to some extent, the quality of neural machine translation (NMT) still need improving.

When the machine translation model generates a result, it cannot be avoided that some irrational results emerge sometimes. For example, there are many words in English that have multiple POS. However, word vector does not take POS into consideration in the training process. This situation will bring confusion to the word vector. Because of the word vector is applied to the machine translation model, it will also have an adverse effect on the translation. This paper will calculate the POS of each word in the English sentence through the POS tagging model, and then train each word with its POS as an entirety.

2 Related work

NMT has achieved good performances in machine translation [Kalchbrenner and Blunsom (2013); Luong, Sutskever, Le et al. (2014); Chorowski, Bahdanau, Serdyuk et al. (2015)]. Many methods such as the rule-based machine translation [Yngve (1957)], memory-based translation, mechanic translation by analogy principle, language modeling, paraphrase detection word embedding extraction [Sato and Nagao (1990)] and end-to-end learning [Chrisman (1991)] are applied to this field. In SMT, deep neural networks began to show promising results [Sutskever, Vinyals and Le (2014)] summarizes the successful application of feedforward neural networks in the framework of phrase-based SMT systems.

The neural network machine translation system is implemented as an encoder-decoder network with recurrent neural networks. The encoder is a bidirectional neural network with gated recurrent units [Rezaeinia, Rahmani, Ghodsi et al. (2019)] that receives an input sequence $x = (x_1,...,x_m)$ and then respectively calculates a forward sequence of hidden states $(\overrightarrow{h_1}, \dots, \overrightarrow{h_m})$, and a backward sequence $(\overleftarrow{h_1}, \dots, \overleftarrow{h_m})$. The hidden states $\overrightarrow{h_k}$ and $\overleftarrow{h_k}$ are concatenated to obtain the annotation vector h_k. The decoder is a recurrent neural network that predicts a target sequence $y = (y_1,...,y_m)$. Fig. 1 shows the structure of a typical English-Chinese translation system that consists of the encoder-decoder network. The input is an English sentence, which is compiled into a vector $z = (z_1,...,z_m)$ by the encoder, and then the vector z is decoded into a syntactic sentence by the decoder.

The most important module existing in the NMT system is language model. This module endows the basic language competence to the whole system and produces word vectors that represent all words digitally. Initially, one-hot vector was the common way that was applied to represent individual word. However, the vector generated by this way was sparse and had a high redundancy. This method would generate a huge number of parameters to represent the words. Therefore, it was prone to waste much computational resource and might cause the dimension explosion. On the other hand, this method failed to describe the connection between two vectors with the similar semantics. Due to the shortages mentioned above, word vector became a better substitute.

Bengio et al. [Bengio, Ducharme, Vincent et al. (2003); Yngve (1957)] used neural networks to train language models predicting the probability distribution of the nth word by inputting the first n-1 words, each of which was represented by a word vector.

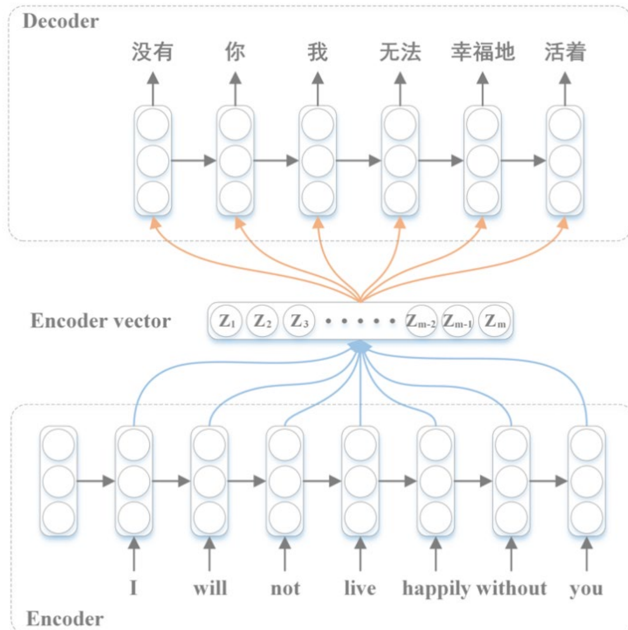

Figure 1: Structure of English-Chinese translation system

Mikolov et al. [Mikolov, Chen, Corrado et al. (2013)] improved this structure and proposed two methods CBOW and Skip-Gram. The CBOW model takes the one-hot vectors of the current word and its context as input. Then these vectors are combined to do dimension conversion simultaneously. In this way, the context information can be added to the outputted representation of the current word. The structure of the CBOW model is shown in Fig. 2.

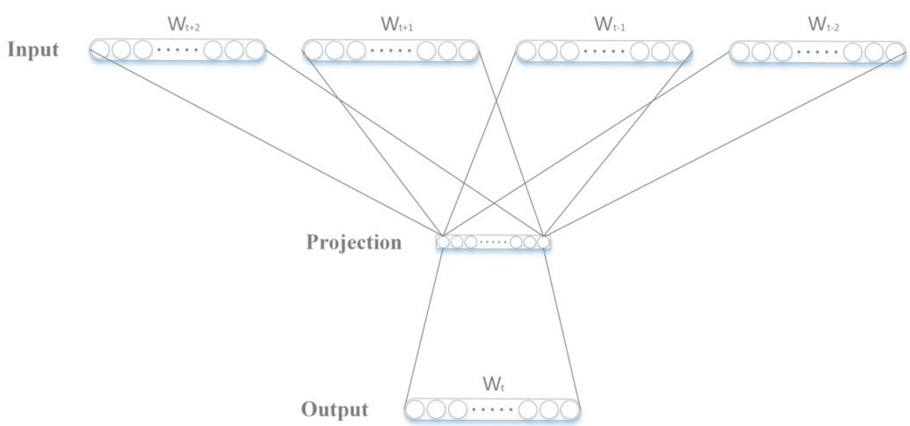

Figure 2: CBOW model for word2vec

As shown in Fig. 3, the Skip-Gram model takes the current word as input and predicts the context of the word. Hierarchical Softmax and Negative Sampling are two training methods combined with two models above, forming four implementations of word2vec.

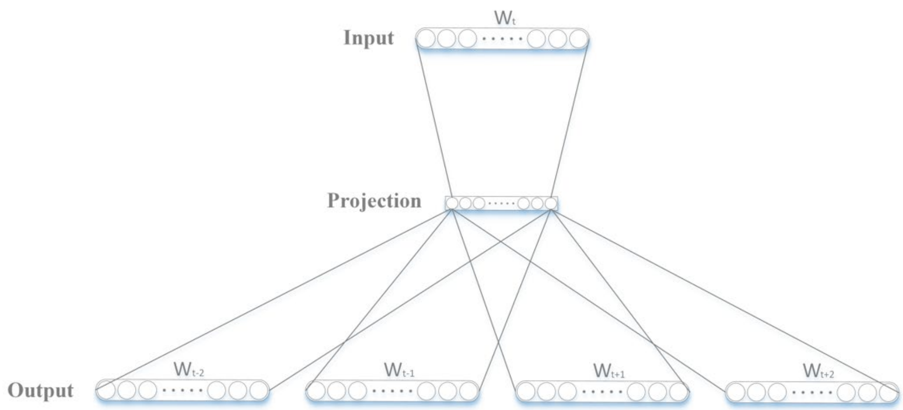

Figure 3: Skip-Gram model for word2vec

Word2Vec, which has been proven to be an effective tool for the distributed representation of words (word embeddings), is usually applied to find the linguistic context. It has beacome the most regular language model applied in NLP tasks. At the same time, many improved methods based on word2vec sprung out in recent years.

McCann B et al. [McCann, Bradbury, Xiong et al. (2017)] used a deep LSTM encoder from an attention-based sequence-to-sequence model trained for machine translation (MT) to contextualize word vectors.

Word2Vec was significantly improved by a new type of deep contextualized word representation [Peters, Neumann, Iyyer et al. (2018)] that models could learn the complex characteristics of word (e.g., syntax and semantics) and how to use these vary across linguistic contexts. Their word vectors captured the internal states of a deep bidirectional language model (biLM), which was pre-trained on a large text corpus.

Miranda et al. [Miranda, Pasti and Castro (2019)] proposed to use a Self-Organizing Map (SOM) to cluster the word vectors generated by Word2Vec so as to find topics in the texts.

Nearest neighbors in word embedding models are commonly observed to be semantically similar, but the relations between them may have a great difference. Hershcovich et al. [Hershcovich, Toledo, Halfon et al. (2019)] investigated the extent to which word embedding models preserved syntactic interchangeability. And they used POS as a proxy for syntactic interchangeability.

Although word vector representations are well developed tools for NLP and machine learning tasks, they are prone to carrying and amplifying bias which can perpetrate discrimination in various applications. To solve this problem, Dev et al. [Dev and Phillips (2019)] explored a new simple way to detect the most stereotypically gendered words in an embedding. Furthermore, for the gender bias exhibited in ELMo's contextualized word vectors Zhao et al. [Zhao, Wang, Yatskar et al. (2019)] explored two methods to mitigate such gender bias and showed that the bias demonstrated on WinoBias could be eliminated Cho et al. [Cho, Van Merriënboer, Gulcehre et al. (2014)] applied RNN encoder-decoder to learn the phrase representation, which creatively used encoder-decoder architecture to obtain word vectors.

In sentiment classification, Word2Vec and GloVe are reliable word embedding methods that are usually applied to NLP tasks. However, these methods ignore the sentiment information of texts. Rezaeinia et al. [Rezaeinia, Rahmani, Ghodsi et al. (2019)] proposed a novel method, Improved Word Vectors (IWV), which increased the accuracy of pre-trained word embeddings in sentiment analysis.

3 Method

In this paper, we calculate the POS of each word through POS tagging model in the source language and integrate POS feature into the word vector. After adding POS feature, the goal of the CBOW is to maximize the follow equation:

$$\mathrm{L} = \sum_{w_{pos} \in c} \log p\left(w_{pos} \middle| Context(w_{pos})\right) \quad (1)$$

where w_{pos} is a word vector that incorporates POS feature. And Skip-Gram needs to maximize the equation:

$$\mathrm{g}(w) = \prod_{\widetilde{w_{pos}} \in Context(w_{pos})} \prod_{u_{pos} \in \{w_{pos}\} \cup NEG^{\widetilde{w_{pos}}}(w_{pos})} p\left(u_{pos} \middle| \widetilde{w_{pos}}\right) \quad (2)$$

where $w_{pos}$and u_{pos} are also word vectors that incorporate POS feature.

This article uses the NLTK [Bird and Loper (2004)] tool to segment words in the source language (English). The abbreviation used for the POS is not the same as the general abbreviation. For example, the adjectives are generally abbreviated as “adj”, but here is “JJ”. The POS used by the tool and its abbreviations are shown in Tab. 1.

By the NLTK tool, we obtain the POS corresponding to each word in each sentence of the source language in the corpus. In order to distinguish the same words of different POS, this paper uses underline to combine the words with their corresponding POS. For example, the word “tear”, as shown in Fig. 4, turns into “tear_NN” because the abbreviation corresponding to the noun is "NN" when we assume that “tear” is a noun here. If we assume that “tear” is a verb, verb corresponding to the abbreviation “VB”, then the word turns into “tear_VB”.

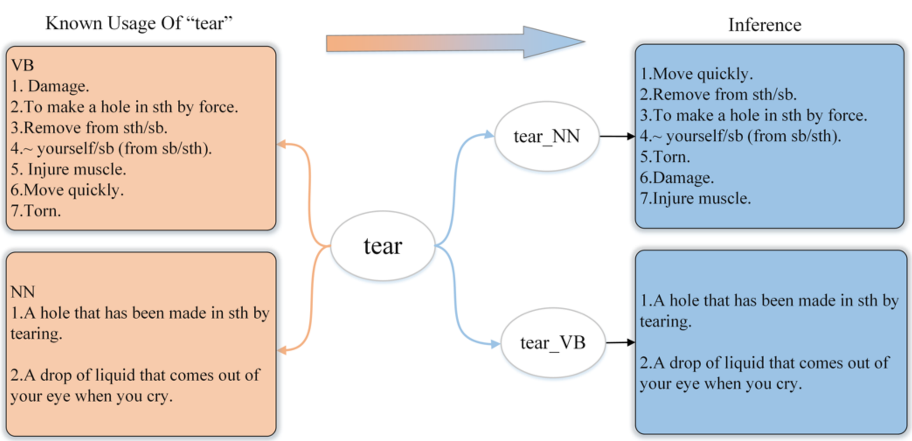

Figure 4: Word according to POS conversion diagram

Table 1: Label of POS feature

Abbre viation	POS	example	Abbre viation	POS	Example
CC	Coordinating conjunction	and	PRP$	Possessive pronoun	her
CD	Cardinal number	twenty-four	RB	Adverb	occasionally
DT	Determiner	the	RBR	Adverb, comparative	further
EX	Existential	there	PBS	Adverb, superlative	best
FW	Foreign word	dolce	RP	Particle	aboard
IN	Preposition	on	SYM	Symbol	%
JJ	Adjective	new	To	To	to
JJR	Adjective comparative	bleaker	UH	Interjection	Goodbye
JJS	Adjective superlative	calmest	VB	Verb, base form	ask
LS	List item marker	A	VBD	Verb, past tense	dipped
MD	Modal	can	VBG	Verb, gerund or present participle	telegraphing
NN	Noun, single or mass	year	VBN	Verb, non-3rd person singular present	multihued
NNS	Noun, plural	undergradu ates	VBP	Verb, 3rd person singular present	predominate
NNP	Proper noun, singular	Alison	VBZ	Verb, 3rd person singular present	bases
NNPS	Proper noun, plural	Americans	WDT	Wh-determiner	who
PDT	Predeterminer	all	WP	WH pronoun	that
POS	Possessive ending	‘	WPS	WH pronoun possessive	whose
PRP	Personal pronoun	hers	WRB	WH adverb	when

If the original sentence is: "*Neither would the time lag data collation affect the health education and disease prevention programme.*" Then it should be converted to the following form: "*Neither_DT would_MD the_DT time_NN lag_NN data_NNS collation_NN affect_VBP the_DT health_NN education_NN and_CC disease_NN prevention_NN programme_NN._.*"

In this paper, each line of the source language file is processed one by one in order, and the converted sentence will be recorded in a new file. Then we replace the original file with the new one and generate a new training set, cross-validation set and test set along with the target language file.

4 Experiments

The English corpus with POS feature and the target Chinese corpus is used as the training set. We train the model for 24 Epoches through OpenNMT on WMT17 English-Chinese dataset. Each epoch contains 13365 iterations. We set the learning rate as 0.0001 and the training process cost 32 hours on GTX2080Ti. The experimental results (BLEU score) of the model from the 13th training cycle are shown in Tab. 2:

Table 2: Translated model generated by word vector with POS Feature and original word vector BLEU score comparison

Epoch	BLEU (original word vector)	BLEU (word vector with POS Feature)	BLEU (Difference)
13	25.47	25.58	+0.11
14	26.36	26.46	+0.1
15	26.07	26.12	+0.05
16	26.26	26.27	+0.01
17	26.23	26.36	+0.13
18	26.20	26.30	+0.1
19	26.23	26.34	+0.11
20	26.25	26.30	+0.05
21	26.34	26.37	+0.03
22	26.26	26.37	+0.11
23	26.27	26.30	+0.03
24	26.29	26.34	+0.05

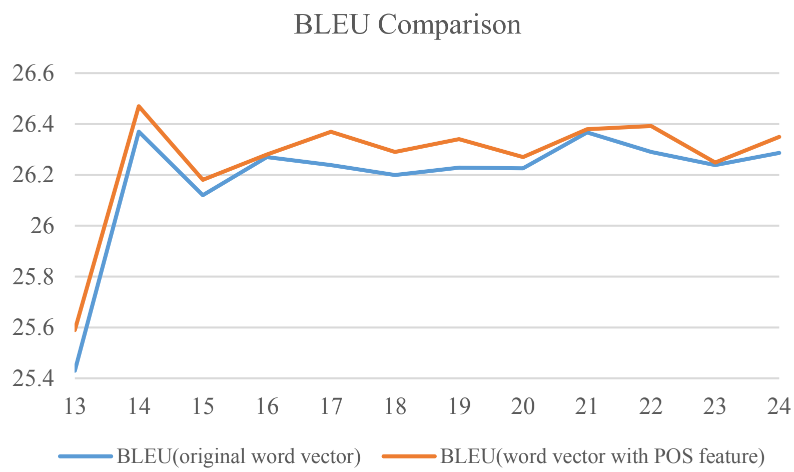

Figure 5: A translation model with word vector added POS feature and original word vector BLEU score line chart

As can be seen from the Tab. 2 and Fig. 5, the model with the POS feature performs better than the one without POS feature. We can learn from the data that this method of constructing word vectors is indeed effective. Moreover, we also apply our training method to other translation model. From Tab. 3, we can see that the models with POS feature achieve average +0.5 than the origin models. These results indicate that our method is a universal approach for improving the translation performance. The translation results are shown in Fig. 6.

Table 3: Some other translation model based on POS feature compared with their base models

Model	BLEU	BLEU (Difference)
RNNSearch	20.31	+0.75
RNNSearch (POS)	21.06	
Seq2atten	21.78	+0.15
Seq2atten (POS)	21.93	
Transformer	25.81	+0.62
Transformer (POS)	26.43	

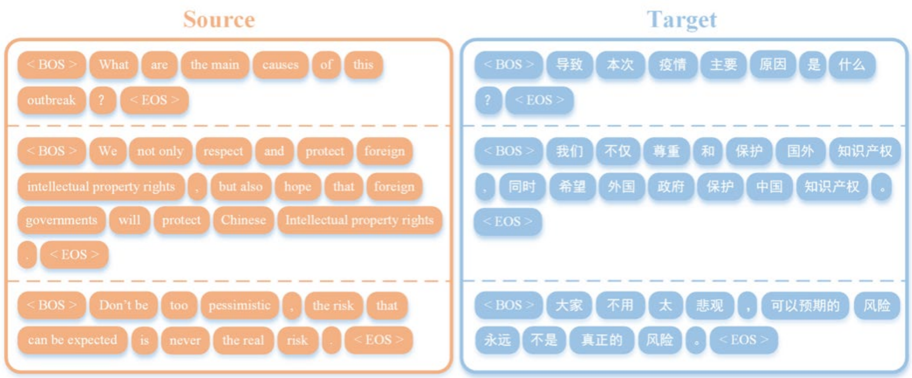

Figure 6: A trial translation results after the integration of POS

5 Conclusion

In this paper, we propose a novel word vector training method by adding POS feature for the word vectors training. This method pays more attention to the different meanings of the same words under different parts-of-speech (POS), which can improve the quality of machine translation. The experimental result verifies that this method contributes a lot to the performance of machine translation.

Funding Statement: This work is supported by the National Natural Science Foundation of China (61872231, 61701297).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

Bengio, Y.; Ducharme, R.; Vincent, P.; Vincent, P.; Janvin, C. (2003): A neural probabilistic language model. *Journal of Machine Learning Research*, vol. 3, pp. 1137-1155.

Bird, S.; Loper, E. (2004): NLTK: the natural language toolkit. *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*.

Cho, K.; Van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F. et al. (2014): Learning phrase representations using RNN encoder-decoder for statistical machine translation. *Computer Science.*

Chorowski, J. K.; Bahdanau, D.; Serdyuk, D.; Cho, K.; Bengio, Y. (2015): Attention-based models for speech recognition. *Advances in Neural Information Processing Systems*. vol. 2015, pp. 577-585.

Chrisman, L. (1991): Learning recursive distributed representations for holistic computation. *Connection Science*, vol. 3, no. 4, pp. 345-366.

Dev, S.; Phillips, J. (2019): Attenuating bias in word vectors. *arXiv preprint arXiv:1901.07656, 2019.*

Hershcovich, D.; Toledo, A.; Halfon, A.; Slonim, N. (2019): Syntactic interchangeability in word embedding models. *arXiv preprint arXiv:1904.00669.*

Kalchbrenner, N.; Blunsom, P. (2013): Recurrent continuous translation models. *EMNLP 2013-2013 Conference on Empirical Methods in Natural Language Processing,* vol. 3, pp. 1700-1709.

Luong, M. T.; Sutskever, I.; Le, Q. V.; Vinyals, O.; Zaremba, W. (2014): Addressing the rare word problem in neural machine translation. *Bulletin of University of Agricultural Sciences & Veterinary Medicine Cluj Napoca Veterinary Medicine*, vol. 27, no. 2, pp. 82-86.

McCann, B.; Bradbury, J.; Xiong, C.; Socher, R. (2017): Learned in translation: Contextualized word vectors. *Advances in Neural Information Processing Systems.*

Mikolov, T.; Chen, K.; Corrado, G.; Dean, J. (2013): Efficient estimation of word representations in vector space. *Proceedings of Workshop at ICLR*, vol. 2013.

Miranda, G. R. D.; Pasti, R.; Castro, L. N. D. (2019): Detecting topics in documents by clustering word vectors. *16th International Conference on Distributed Computing and Artificial Intelligence*, vol. 1003.

Peters, M. E.; Neumann, M.; Iyyer, M.; Gardner, M.; Clark, C. et al. (2018): Deep contextualized word representations*. arXiv preprint arXiv:1802.05365.*

Rezaeinia, S. M.; Rahmani, R.; Ghodsi, A.; Veisi, H. (2019): Sentiment analysis based on improved pre-trained word embeddings. *Expert Systems with Applications*, vol. 117, pp. 139-147.

Sato, S.; Nagao, M. (1990): Toward memory-based translation. *Proceedings of the 13th conference on Computational Linguistics*, vol. 3, pp. 247-252.

Sutskever, I.; Vinyals, O.; Le, Q. V. (2014): Sequence to sequence learning with neural networks. *Advances in Neural Information Processing Systems*.

Yngve, V. H. (1957): The technical feasibility of translating languages by machine. *American Institute of Electrical Engineers, Part I: Communication and Electronics*, vol. 75, no. 11, pp. 994-999.

Zhao, J.; Wang, T.; Yatskar, M.; Cotterell, R.; Ordonez, V. et al. (2019): Gender bias in contextualized word embeddings. *arXiv preprint arXiv:1904.03310.*