A Simple and General Strategy for Referential Problem in Low-Resource Neural Machine Translation

Anonymous authors
Paper under double-blind review

Abstract

This paper aims to solve a series of referential problems in sequence decoding caused by data sparsity and corpus scarce in low-resource Neural Machine Translation (NMT), including pronoun missing, reference error, bias and so on. It is difficult to find the essential reason of these problems because they are only shown in the prediction results and involve all aspects of the model. Different from the usual solutions based on complex mathematical rule setting and adding artificial features, we expect to turn the problems in the predictions into noise as much as possible, and use adversarial training to make the model find the balance between the noise and the golden samples, instead of exploring the reason of the problem during the complex training. In this paper, only a simple noise-based preprocessing operation and a slight modification of the adversarial training can make the model generalize to a series of referential problems in low-resource NMT task. On Korean-Chinese, Mongolian-Chinese and Arabic-Chinese tasks, the evaluation of BLEU score and the accuracy of pronouns in sequence have been significantly improved.

1 Introduction

The problem of referential errors exist in most Nature Language Processing (NLP) tasks, which is caused by inadequate training, incomplete semantic structure in corpus, and lack of the ability to capture complex context. In NLP, we usually have to use many tricks to alleviate or narrow the gap between prediction distribution and truth in sequence-to-sequence tasks. Among them, the performance of prediction results may come from any link of training, the process from the noise of corpus(Koehn & Khayrallah (2018)) to the performance of embedding (Liu et al. (2016)), from the compression ability of encoder to the fidelity and efficiency of semantic information of decoder with the help of attention mechanism(Christopher et al. (2015)), from the generalization ability of the model to the readability of the translation(Marco & Brenden (2017)), each specific problem needs specific methods to improve the model. However, for low resource task, the essential problems caused by the scarcity of corpus and data sparsity often cause a series of problems in the interrelated and tedious translation model. Such problems not only need many tricks to alleviate, but more importantly, this chain reaction makes researchers unable to find the essence of the problem accurately. Referential resolution is one of the extremely difficult problems. The common practice is to predict the antecedent of the referent and the reference relationship through deep neural networks. Some contributions(Qingyu et al. (2018); Shanheng & Hwee (2007); Chen & Vincent (2013)) show the neural network's ability to represent the pronouns and antecedents in the vector space much more than the traditional methods. However, they all need to use a lot of mathematical knowledge to set complex training rules and add more or less artificial features, which make the referential problem out of reach. It is straightforward to capture the referential relationships in sequences and paragraphs through deeper and more complex network structures(Durrett & Klein (2013); Kenton et al. (2017)), but complex models not only confuse the training, but also make some specific tasks impractical. On this basis, in order to enhance the model's ability to capture referential relationships, reinforcement learning (RL)(William et al. (2018)) enables the model to accurately correct the relationship between antecedents and pronouns through policy iterations within a limited training period(Qingyu et al. (2017)). On the other hand, referential ambiguity and prediction bias are particularly serious in low-resource translation tasks. The reasons may come from many aspects, such as sparse vocabulary, missing semantics and some non-specific named entities are not sensitive to pronouns. We take a 'real and usual' example to illustrate the impact of the accuracy of referential relations on translation in Korean-Chinese machine translation task.

test sequence: @교수는 매우 기뻤고 남자 친구는 선물을 사서 항상 가지고 다녔습니다.. @ (The professor was very happy, her boyfriend bought her a gift and she always carried it with her.)

Transformer_basic (after training based on daily corpus) is decoded as:

@ 教授很幸福，他的男朋友给她买了礼物并且总是随身携带。@. (The professor was very happy that his boyfriend bought her a gift and always carried it with him.)

We can see that there are two typical referential errors in this example. There is inherent bias in the corpus, which causes the first 'her' to be translated as 'his' according to the probability candidate set. When the second 'her' is associated, partly based on the first pronoun 'his' and partly based on 'professor', so the next prediction bias and the first reference error continued to the pronoun 'him'. Then, in the case of losing a 'she', the last 'him' also appeared to be ambiguous. In addition, we were surprised to find that when the prediction result obtained the wrong pronoun, the impact on the translation was not only in the position of the wrong pronoun, but also transmitted to the entire subsequent sequence. It is because the decoder will perform a new greedy search in the vocabulary for the current pronoun and make new predictions based on context semantics.

In this paper, we use simple preprocessing methods instead of complex mathematical rule settings to solve a series of problems such as ambiguous references in translation in low-resource NMT. The core of the proposed strategy is that we have added a pseudo sequence which is obvious and contrary to the facts, so that the model can correct errors or bias to this type of reference relationship in adversarial training. Specifically, we adopt a method of adding noise (see section 2.1) to make the model dynamically generalize this noise through adversarial training(Wu et al. (2018)). This strategy is similarly presented in the work(Yatu et al. (2019b;a)), they add corpus of different granularity to the training data in the form of noise to filter out which granularity is most suitable for the current decoding process. We believe that this type of strategy can be transferred to many NLP problems, not just referential relationship problems. The difference is that the strategy of noise addition in this paper is essentially different from simply training the original data multiple times. To put it simply, the effect of multiple training on the same sequence and the updating of different parameters of the similar sequence is quite different(Belinkov & Bisk (2018); Koehn & Khayrallah (2018)). The contributions of this paper can be summarized in the following three points:

- We propose a strategy that takes the focused and unresolved targets as noise. In this paper, reference-related noise is added to the training data in the form of a pseudo-sequence.
- We normalize the referential relationship and the pronoun accuracy to the BLEU score, instead of adding complex mathematical rules to the loss function and evaluation matrix.
- In order to match the rationality of this strategy, that is, to allow the model to have extra interest and focus to pronouns during the training process, we add a focus module on the basis of the Generative Adversarial Networks(GAN) model to focus on the referential relationship in sequence decoding. We use value iteration network(VIN) as a focus module because GAN has the essence of RL training. In VIN, the incorrect referential prediction corresponds to a low reward, whereas the low reward corresponds to a low value. This is what the focus module wants to emphasize.

At this point, a simple data preprocessing operation and a focused module for GAN training, we expect to use this strategy to get rid of the dilemma of complex rule design or loss of semantics like hard debiasing method(Tolga et al. (2016)). In Section 2, we will introduce the details of the model and discuss the necessity of key modules. Then we introduce the verification experiments in Section 3 and Section 4, including preprocessing methods and analysis of experimental results. Finally, we briefly summarize the portability and conclusion of the method.

2 Model Description

The model we present is mainly divided into three parts: generation module G, focus module F, and discrimination module D. Similar to the usual GAN module, G based on RL strategy(Volodymyr et al. (2013)) is used to transform the source-side embedding to the target-side sequence using the policy gradient algorithm. This generation relationship will be described in Section 2.2. In order to clearly present the training process of the proposed strategy and model, we will divide into three parts to connect and explain the logic of the entire strategy: the preprocessing for obtaining noise, the RL training to enhance the accuracy of the reference relationship and the noise.

2.1 Preprocessing-Noise

To be straightforward, we want the model to generalize the noise, so we directly add the corresponding noise to the training data to familiarize the model with it. Here, noise is about several major referential problems that arise in the process of low resource MT. Generally, there are three types of referential errors: pronoun missing and overlapping, referential errors, and bias referential. The missing and overlapping of pronouns is similar to the other components, which is largely due to underfitting. Translation models can usually solve such problems with the help of multiple iterations of training or regular optimization. For referential errors, we still take the $test\ sequence$ as an example, first, this paper copies and tags the training data as pseudo-sequences (only the reference pronouns in the data need to be tagged), and then the pronouns of the pseudo-sequences are masked and replaced. This ensures that pronouns can be fully generalized without distortion.

pseudo-sequence 1 - $replace$: translation: (@The professor was very happy that ~~her~~(his)(its)boyfriend bought ~~her~~(his)(its) a present and took ~~it~~ (him)(her) with him.@)

pseudo-sequence 2 - $mask$: translation: (@The professor was very happy that ~~her~~(@mask@) boyfriend bought ~~her~~(@mask@) a present and took ~~it~~(@mask@) with him.@)

Note: In both pseudo-sequences[1], all pronouns are replaced by possible pronouns or mask symbols $@mask@$. Due to the different grammatical structure, the last 'him' in translation does not actually appear in Korean. The bias problem in translation are sensitive and cumbersome in low-resource tasks. Such problems not only involve the accuracy of pronouns, but also affect the prediction accuracy of the entire sequence. In view of this problem, we also boil down to these two forms of noise.

2.2 Reinforcement Learning Training

The overall network structure is shown in Figure 1. The entire adversarial training is guided by RL algorithms to optimize model parameters, which is also a common strategy of GAN in sequence generation tasks. This is consistent with why we use VIN, so VIN can be perfectly integrated into the entire adversarial training.

The first thing to be clear is how the RL algorithm is mapped to the sequence generation task. Here we only list some mappings that are more concerned in NMT. In a typical RL algorithm, the following standard variables (agent, police, action, state, reward) usually correspond to (generator, parameter, prediction of each iteration, hidden units, BLEU score of predicted sequence) in the sequence (x, y) generation task. The translation model as G is used to sense the environment state s of the network when it is mapped as an agent. Such an action a updates the entire state parameters θ by a fixed training police. The reward is based on the distribution gap between the predicted sequence and the gold sequence, which is the BLEU score. The entire training objective O_θ can be expressed as two expectations about maximum and minimum:

$$O_\theta = \begin{cases} E_{groundtruth} G \sim logD(x,y) & min \\ E_{Discriminator} D \sim log(1 - D(x,y)) & max \end{cases} \tag{1}$$

[1]In this paper, the initial effective proportions of the two pseudo-sequences in the data were determined in the experiment. At the beginning of the adversarial training, $original$: $replace$: $mask$ = 8: 1: 1. During the training process, the two noises of each epoch increase by 1% respectively, which corresponds to a reduction of 2% of the original data.

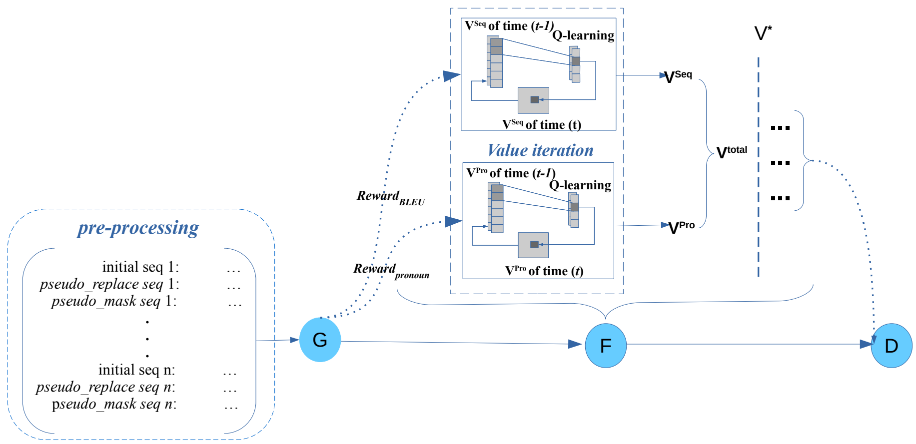

Figure 1: Model architecture. Contains preprocessing form and model details.

G uses a preprocessed corpus to update the random hidden layer states and rewards. The module F is used to evaluate G's output according to the rewards generated by RL. D discriminates the corresponding sequence based on the value generated by F, which is consistent with the usual GAN process. In order to prevent D from always getting negative feedback, G and D are trained alternately, and the sampling method of directional search is used to serve the gradient calculation, and the weight of D is appropriately limited. For the detailed derivation of the GAN training process, please refer to the detailed description in study(Wu et al. (2018); Yang et al. (2018)).

2.3 Focus module and Discriminate module

In this paper, the main problem we address is the referential relationship, so it is particularly sensitive to referential noise added in preprocessing. This is due to the fact that we use the score of the sequence BLEU and the reference BLEU as the evaluation criteria for rewards. In other words, predictions with correct reference relations and higher BLEU score will yield a higher reward.

The module F is between the G and the D. The main contribution of this module is to give priority to D to identify sequences with less reward according to the reward generated by G, where less reward correspond to inaccurate reference relations. We refer to the implementation of VIN in the work (Yatu et al. (2019a)) and adopt two simple CNN to realize the entire value iteration process. Different from work (Yatu et al. (2019a)), we pay attention to two aspects of reward in our method: BLEU rewards for the entire sequence V_t^{Seq} and BLEU rewards for referential relationships V_t^{Pro}:

$$V_t^{Seq} = max_a Q(s,a) \qquad = max \left[R_B(s,a) + \sum_{N}^{t=1} P(s|s_{t-1},a)V_{t-1} \right] \tag{2}$$

$$V_t^{Pro} = max_a Q(s,a) \qquad = max \left[R_P(s,a) + \sum_{N}^{t=1} P(s|s_{t-1},a)V_{t-1} \right] \tag{3}$$

$$V_{total} = (1-\alpha)V^{Seq} + \alpha V^{Pro} \tag{4}$$

where Q indicates the value of action a under state s at t-th timestep, the reward $R_{B/P}(s,a)$ and transition probabilities p are obtained from G. N represents the sequence length. The value of the sequence is obtained by the accumulation of rewards within a state. The total value V_{total} dynamically combines the V_t^{Pro} and V_t^{Seq} into a representative value according to α, where α is the prediction accuracy of the current training cycle model. This value is used to compare with V^*, which represents the value of the pre-trained model, to determine the current batch training priority.

Since the output participating in the optimal value comparison requires a 0-dimensional tensor, we need to fuse the two values proportionally. We directly control the measurement of the value of F based on the accuracy of the current iteration, so that the model can generate effective value according to the training status.

The core of the module F is to iteratively generate the value of the input reward that can represent the current training cost. Some algorithms that predict behavior though value selection can be considered, such as Q-learning(Jesse & Eric (2020)), Sarsa(Yinhao et al. (2013)), and Deep Q-Network(Hong et al. (2018)). Considering that the sequence decoding in this paper belongs to one-step generation, Q-learning[2] is used in decoding. Q-learning can be understood as the accumulation of action's rewards in timestep t, but this accumulation will decay according to λ .

$$Q(s,t) = r_{t+1} + \lambda^2 r_{t+2} + ... + \lambda^{t+n} r_{t+n+2} \tag{5}$$

The responsibility assumed by module D is relatively simple, that is, identifying the generated sequence selected by F and ground truth, so that the sequence of interest can be preferentially entered into the next round of iterative training. In view of the excellent performance of CNN in binary classification tasks, here we use a simple CNN as a D to form GAN.

3 Experimental Settings

For the data used for verification, the part-of-speech tagging tool is a prerequisite. Researchers need to make a wise choice between some open source projects[3] and targeted construction projects.

3.1 Experimental Data

We verify the effectiveness of the proposed approach on three low-resource corpora: Mongolian-Chinese (Mn-Ch, 0.2M), Korean-Chinese (Kr-Ch, 0.1M), Arabic-Chinese (Ar-Ch, 2.2M). The data comes from CLDC, machine translation track of evaluation campaign CWMT2017 and OPUS in LREC2012[4], respectively. The composition of the corpus is distributed in news, daily life, and government document.

3.2 Experimental Setup

We select the baseline system from two perspectives: model and strategy. In order to highlight the effectiveness of the strategy, we choose Transformer_basic(Ashish et al. (2017)), which performs best in multiple languages, and it has a good performance in focusing on the overall semantic information. In terms of model, the model in this paper is based on adversarial training, so we use two related typical GAN models as the baseline system, BR-CSGAN(Yang et al. (2018))[5] and F-GAN(Yatu et al. (2019a))[6], and basically maintain the parameters in the original baseline system in order to clearly observe the experimental results. Some minor adjustments are made to cater to the inherent experimental conditions. For example, because the mask strategy was added in the preprocessing stage, the setting of Dropout was canceled. We also increased the batchsize to 128 to allow the noise and the original data to be fully trained, and all models are trained on up to single Titan-X GPU.

4 Verification

The validity of the training strategy and model will be verified from four questions:

[2] Monte-Carlo search algorithm (MC) is used in GAN to evaluate intermediate states and directly select behavioral strategies, such as Policy Gradients, which can only be used for model updating in training.

[3] Mn-Ch: CRF++: https://github.com/othman-zennaki/RNN_Pos_Tagger,
Kr-Ch: https://sourceforge.net/projects/hannanum/,
Ar-Ch: http://opennlp.apache.org/.

[4] http://opus.nlpl.eu/,
https://object.pouta.csc.fi/OPUS-MultiUN/v1/moses/ar-zh.txt.zip.

[5] https://github.com/ZhenYangIACAS/NMT_GAN

[6] https://github.com/jiyatu/Filter-GAN.git

- How to verify the role of the proposed strategy and model in reference relations?
- How to ensure the accuracy and fluency of the prediction sequences on the premise of improving the reference relationship?
- Does the additional module F affect the efficiency of the entire training?

4.1 Three Verification Indicators Are Used to Solve the Above Problems

$BLEUforreferential(BLEU\_Pro)$: Unlike intuitive cognition, we believe that the rigid identification of pronouns corresponding to the source and target will weaken the role of the pronoun in the entire sequence. The most straightforward evaluation matric BLEU score is also used to measure the accuracy of the referential relationship. Different from the sequence BLEU, we mask out the rest except pronouns. Such a calculation method can not only accurately and comprehensively reflect the influence of the referential relationship on the translation, but also avoid the introduction of complex mathematical rule indicators.

$BLEUforsequence(BLEU\_Seq)$: The model still needs to ensure the accuracy of the entire sequence when solving the referential relationship, which is the original intention of machine translation.

$Trainingefficiency$: We record three indicators that most intuitively reflect the training process of translation model: the convergence process of loss, the trend of accuracy, and the training time.

4.2 Verification Results and Analysis

As mentioned in Section 4.1, in order to meet the original intention of the NMT task, we use the most direct machine translation matrix BLEU score instead of complex antecedent speculation and F-score. This is also consistent with the original intention of this paper to simplify the process of measuring reference relations.

4.2.1 BLEU for Referential and BLEU for Sequence

We have calculated the BLEU score of different systems in three low-resource tasks in the original state and the increased noise state, including BLEU_Pro and BLEU_Seq, as shown in Table 1.

Table 1: The performance of different systems on the two BLEU scores, including the effect of noise preprocessing on the GAN-based system.

system		Mn-Ch		Kr-Ch		Ar-Ch	
		BLEU_Pro	BLEU_Seq	BLEU_Pro	BLEU_Seq	BLEU_Pro	BLEU_Seq
Transformer_basic		56.3	28.5	47.7	**24.7**	60.1	30.8
BR-CSGAN	-	47.5	27.4	42.5	23.3	57.7	30.1
	+pre_noise	50.8	27.9	44.1	23.9	59.2	31.3
F-GAN	-	57.9	29.1	38.8	20.4	62.5	31.3
	+pre_noise	58.6	32.3	41.7	21.2	66.7	32.4
Our	-	48.2	31.2	42.3	22.6	62.9	30.8
	+pre_noise	**64.3**	**34.8**	**48.5**	24.3	**67.5**	**33.7**

First, we explore the sensitivity of different systems to noise preprocessing strategy. It is easy to find that the noise strategy in each system can bring 0.5 to 3.6 BLEU_Seq score improvements to the model, and such improvements are mainly distributed in the adversarial training system. This is because the adversarial mechanism enables the model to dynamically train noise in a limited training period and generate generalization capabilities. For BLEU_Pro, there is a maximum of 6.1 BLEU score improvement.

On the other hand, we observe that after the model is preprocessed, the BLEU_Pro score has the highest improvement and the corresponding highest improvement is also achieved with the BLEU_Seq score. We believe this is not accidental, because after improving the referential accuracy, the subsequent decoding of the model will explore new candidate spaces for the correct referents, which is very important for the effectiveness of the general greedy search algorithm.

The proposed approach also performs a good ability in most tasks without the cooperation of noise preprocessing, which is due to the seamless connection between the module F and RL. The result show that our model can quickly converge to a more optimized state during insufficient training cycles.

4.2.2 Training Efficiency

For the statistical results after noise preprocessing (Figure2), in order to show the trend of accuracy in the training process clearly , we increase the sampling node span, so the fluctuation of the curve in the graph will become more obvious.

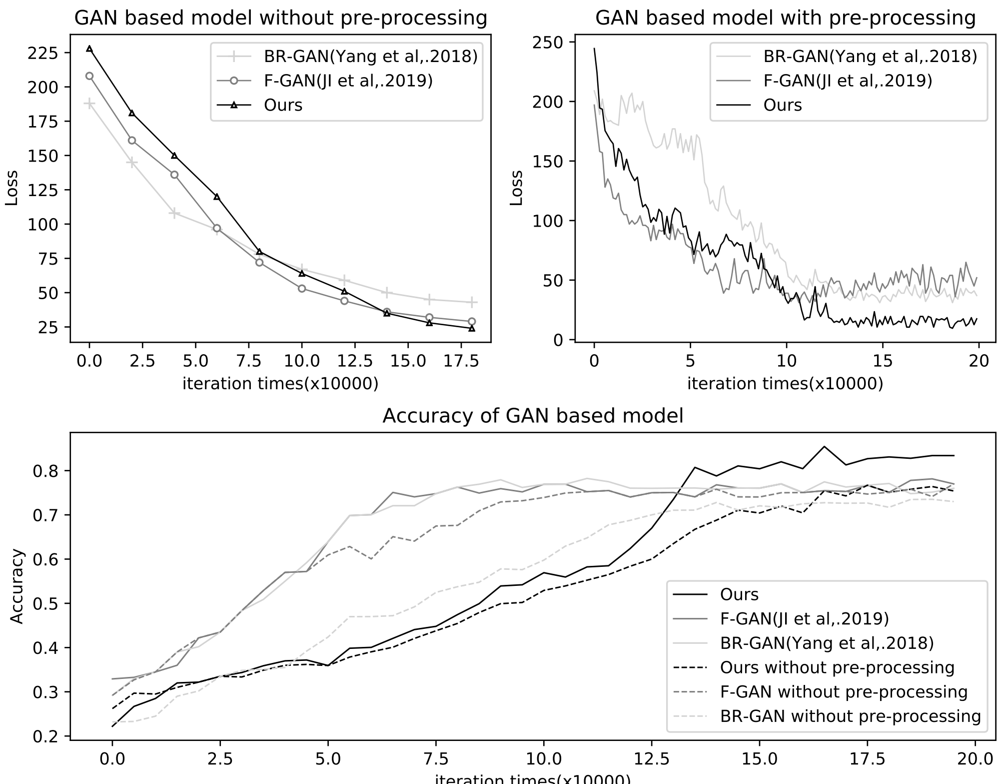

Figure 2: The influence of noise preprocessing strategy on the change of loss and the trend of accuracy during training. The figure shows the two most intuitive training indicators, training loss and accuracy rate during 20*10,000 iterations. Note that because the network structure of the baseline system Transformer_basic is different from other baseline systems and our system in this paper, the model efficiency in terms of training efficiency is not comparable.

The loss of the three adversarial models converges quickly at the beginning of training, This is why we use GAN as the original model. The advantages of the adversarial mechanism allow us to eliminate some suspicious factors in order to clearly observe the noise strategy effect. It also can quickly converge under the cooperation of preprocessing strategy, and finally achieve a significantly lower loss.

We also counts the adversarial training time of each model in different corpora, see table (2). The

Table 2: The performance of training efficiency of each system in different tasks

	BR-CSGAN		F-GAN		Ours	
	-	+pre_noise	-	+pre_noise	-	+pre_noise
Mn-Ch	31	43	29	27	27	**23**
Kr-Ch	24	37	20	**19**	21	20
Ar-Ch	54	68	50	44	50	**41**

experimental results are relatively clear, and the results observed in the table can be summarized into three analyses:

- Among the three adversarial systems, the model with the focus module has significantly less training time than BR-GAN and F-GAN, which is attributed to the value modules focus on noise.
- The added noise does not add extra training time to the model.
- The proposed model shows better training efficiency on almost all tasks, whether on the initial model or after adding noise.

4.2.3 Heat Map for Reference

A heat map mapping of a typical example sentence is given here to illustrate the decoding effect of the proposed model, see Figure (3). The pronouns highlighted by gray rectangles can be more

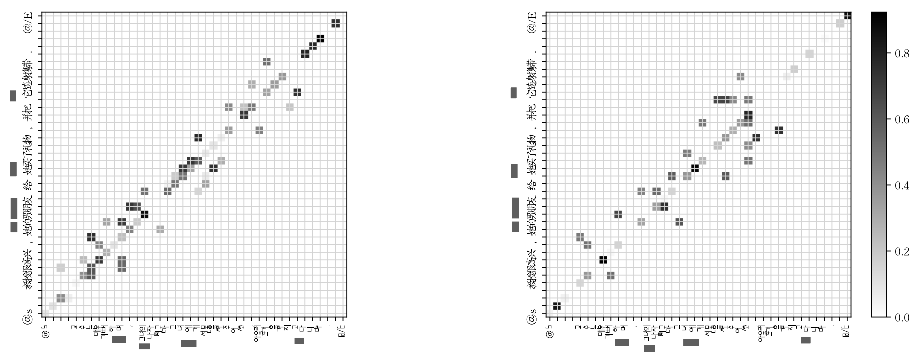

Figure 3: An example of a heat map after decoding and reordering: *left*-with noise preprocessing, *right*-without noise preprocessing.

accurately mapped to the target language in the proposed model, and provide a richer candidate set space. The deeper color of these candidate words in the heat map indicates that the accuracy of the candidate words provided is higher. Under this premise, the subsequent decoding will not deviate from the golden answer a lot, which is conducive to improving the accuracy and fluency of the whole sequence. In addition, such corrections are not isolated. For non-pronoun terms, it is easy to observe that their prediction is also affected by the reference relationship to a certain extent, especially for words related to the pronoun, whose prediction is directly determined by the predictive ability of the pronoun. There are also inherent biases in the sequence, such as 'he' referring to 'professor' in the right part, which is based on the inherent bias and collocation that already exists in the vocabulary. In fact, the golden fact here is 'her', which is corrected in our model (left). This can be attributed to the addition of pseudo sequences with 'her', 'it' and 'he' as pronouns in the noise preprocessing.

5 Summary

This paper is devoted to solving the problems of inaccurate reference relations caused by sparse vocabulary in low-resource NMT task, including incorrect reference relationship and bias. The main contribution is to use a simple preprocessing operation combined with adversarial learning to improve the translation accuracy of pronouns in machine translation, thereby avoiding the setting of complex mathematics and language rules. In terms of BLEU score, it is verified that the proposed strategy shows impressive results both in the prediction result of the whole sequence and the reference relationship, and it not only does not bring extra training cost to the model, but also saves training time to a certain extent.

The motivation of this paper is to convert the problems encountered into noise and generalize the problems through adversarial training. We look forward to exploring a more general training objective in future work to extend this problem-solving approach and strategy to more NLP tasks.

References

Vaswani Ashish, Shazeer Noam, Parmar Niki, Uszkoreit Jakob, Jones Llion, N. Gomez Aidan, Kaiser Lukasz, and Polosukhin. Illia. Attention is all you need. In Advances in neural information processing systems, NIPS, pp. 6000--6010, 2017.

Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. In International Conference on Learning Representations, ICLR, 2018. URL https://openreview.net/forum?id=BJ8vJebC-.

Chen Chen and Ng. Vincent. Chinese zero pronoun resolution: Some recent advances. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 1360--1365, 2013.

D Manning Christopher, Luong Minhthang, and Pham Hieu. Effective approaches to attention-based neural machine translation. Computing and Language, arXiv:1508.04025, 2015. URL https://arxiv.org/abs/1508.04025.

G Durrett and D. Klein. Easy victories and uphill battles in coreference resolution. In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 1971--1982, 2013.

Z W Hong, S Y Su, and T Y Shann. A deep policy inference q-network for multi-agent systems. In Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems., pp. 1388--1396, 2018.

Clifton Jesse and Laber Eric. Q-learning: Theory and applications. Annual Review of Statistics and Its Application, 7(1):279--301, 2020.

Lee Kenton, He Luheng, Lewis Mike, and Zettlemoyer. Luke. End-to-end neural coreference resolution. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 188--197, 2017.

P Koehn and H Khayrallah. On the impact of various types of noise on neural machine translation. In Meeting of the Association for Computational Linguistics, ACL, pp. 74--83, 2018.

Kang Liu, Shizhu He, Siwei Lai, and Jun Zhao. How to generate a good word embedding. Alternation, 31(6):5--14, 2016. doi: 10.1109/MIS.2016.45.

Baroni Marco and M Lake Brenden. Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks. Computing and Language, arXiv:1711.00350, 2017. URL https://arxiv.xilesou.top/abs/1711.00350.

Yin Qingyu, Zhang Yu, Zhang Weinan, and Liu. Ting. Chinese zero pronoun resolution with deep memory network. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 1309--1318, 2017. doi: 10.18653/v1/D17-1135.

Yin Qingyu, Zhang Yu, Zhang Weinan, Liu Ting, and Yang Wang. William. Deep reinforcement learning for chinese zero pronoun resolution. In Meeting of the Association for Computational Linguistics, ACL, pp. 569--578, 2018.

Zhao Shanheng and Tou Ng Hwee. Identification and resolution of chinese zero pronouns: A machine learning approach. In Proceedings of the 2007 Conference on Empirical Methods in Natural Language Processing, EMNLP, pp. 541--550, 2007.

Bolukbasi Tolga, Chang Kai-Wei, Y. Zou James, Saligrama Venkatesh, and Kalai. Adam. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In Advances in neural information processing systems, NIPS, pp. 4349--4357, 2016.

Mnih Volodymyr, Kavukcuoglu Koray, Silver David, Graves Alex, Antonoglou Ioannis, Wierstra Daan, and Riedmiller. Martin. Playing atari with deep reinforcement learning. Computing and Language, arXiv:1312.5602, 2013.

YangWang William, Li Jiwei, and He. Xiaodong. Deep reinforcement learning for nlp. In Meeting of the Association for Computational Linguistics, ACL, pp. 19--21, 2018.

L Wu, Y Xia, F Tian, and et al. Adversarial neural machine translation. In Asian Conference on Machine Learning, ACML, pp. 534--549, 2018.

Zhen Yang, Wei Chen, Feng Wang, and Bo. Xu. Improving neural machine translation with condition sequence generative adversarial nets. In North American chapter of the Association for Computation Linguistics, NAACL, pp. 1335--1346, 2018.

Ji Yatu, Hou Hongxu, Chen Junjie, and Wu Nier. Adversarial training for unknown word problems in neural machine translation. ACM Transactions on Asian and Low-Resource Language Information Processing, TALLIP, 19(1):1--12, 2019a. doi: 10.1145/3342482.

Ji Yatu, Hou Hongxu, Wu Nier, and Chen Junjie. Exploring the advantages of corpus in neural machine translation of agglutinative language. In International Conference on Artificial Neural Networks, ICANN, pp. 326--336, 2019b.

Wang Yinhao, S Li Tzuuhseng, and Lin. Chihjui. Backward q-learning: The combination of sarsa algorithm and q-learning. Engineering Applications of Artificial Intelligence, 26(9):2184--2193, 2013.