Proceedings of CONF-MLA 2026 Symposium: Intelligent Systems and Automation: AI Models, IoT, and Robotic Algorithms
DOI: 10.54254/2755-2721/2026.LD38489

Selective Segmentation Strategies for Chinese-to-English Translation of Long Patent Sentences Based on ParaPat

Xiongwei Shen

School of Information and Communication Engineering, Communication University of China, Beijing, China
202311123018@mails.cuc.edu.cn

Abstract. Long-sentence translation is a challenging task in the processing of Chinese-English patent texts. The traditional whole-sentence translation method (NOSPLIT) and the fully mechanical segmentation method (ALLSPLIT) are prone to information loss, contextual disruption, or under-translation. This study proposes the RLSEGMENT strategy, which uses reinforcement learning to train a policy network to selectively segment candidate sentence boundaries and adopts sentence-level BLEU gain as the reward to optimize translation quality. The experiment uses the ParaPat English-Chinese parallel patent corpus. The results show that the overall BLEU score of RLSEGMENT is 11.9833, which is higher than that of NOSPLIT (10.0127) and ALLSPLIT (9.9518). These results show that RLSEGMENT achieves better BLEU performance and uses fewer segments than ALLSPLIT. It also reduces suspected under-translation compared with NOSPLIT, suggesting that selective segmentation is more practical than uniformly splitting every candidate boundary for patent long-sentence translation. This study analyzes key experimental metrics, visualized results, and model behavior, providing a reference for research on long-sentence translation.

Keywords: machine translation, Chinese-English translation, long patent sentences, selective segmentation, reinforcement learning

1. Introduction

Neural machine translation is an important research direction in the field of natural language processing. In recent years, with the development of deep learning models and the Transformer architecture, machine translation has made significant progress in general text scenarios [1]. However, in professional fields such as patent texts, due to the complex sentence structure, long length, and abundant technical expressions, directly translating entire sentences is prone to information omission, under-translation, and deviations in context understanding [2-4]. Therefore, improving the stability and completeness of long-sentence translation holds significant research value.

Previous studies have shown that long-sentence input significantly increases the processing difficulty of neural machine translation. Appropriate sentence segmentation can alleviate this problem to a certain extent [2, 5]. However, the segmentation strategy itself has trade-offs: not

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.LD38489

segmenting at all makes it difficult to model long-range dependencies, while excessive segmentation may disrupt semantic coherence and thereby affect the overall translation quality. Therefore, the key issue in long-sentence translation lies in how to select an appropriate segmentation position among different candidate boundaries. Meanwhile, automatic evaluation metrics such as BLEU are still widely used for comparing the performance of different methods [6].

Based on the above issues, this paper uses the ParaPat English-Chinese parallel patent corpus as the experimental data source to study the issue of selective segmentation in Chinese-English patent long-sentence translation [7]. In terms of the method, the paper draws on the idea of reinforcement learning in sentence boundary decision-making and design the RLSEGMENT strategy [8]. The paper makes SPLIT/CONTINUE decisions for candidate punctuation boundaries and compare it with the NOSPLIT and ALLSPLIT methods to analyze the effect of selective segmentation on long sentences of different length intervals and its impact on translation quality.

2. Dataset and experimental setup

2.1. Dataset

This experiment uses the ParaPat English-Chinese parallel patent corpus as the data source [7]. ParaPat provides aligned Chinese and English patent texts. In this work, it is used for Chinese-English patent translation and BLEU-based evaluation. Since many patent sentences are long and contain complex technical expressions, the corpus is used to test whether segmentation can help preserve information in long-sentence translation.

During preprocessing, sentence pairs with source token lengths between 180 and 511 are retained. The preprocessing steps include text normalization, length filtering, and length-ratio filtering. Sentence pairs with obvious noise or abnormal source-target length ratios are removed. Candidate segmentation boundaries are selected from three Chinese punctuation marks: ",", ";", and ":". If these marks appear inside brackets or quotation marks, they are not used as segmentation boundaries.

The final dataset contains 20,000 training samples, 300 development samples, and 1,000 test samples. The samples are divided into three source-length intervals: 180–255, 256–383, and 384–511 tokens. The three intervals account for 88.0%, 10.0%, and 2.0% of the data, respectively. The same length-band setting is used in the training, development, and test sets, so the performance of different methods can be compared across sentence lengths.

2.2. Experimental environment

The experiments are run on a cloud Linux server. The main GPU is an NVIDIA GeForce RTX 5090 with 32 GB of GPU memory. The server also provides 25 vCPUs, an Intel Xeon Platinum 8470Q processor, and 90 GB of memory. This environment is used for model inference, segment caching, segmentation-plan caching, and reinforcement learning training. The software versions are CUDA 12.8, Python 3.12.3, and PyTorch 2.8.0+cu128.

3. Model and methods

All three strategies use the same translation backbone, Helsinki-NLP/opus-mt-zh-en, which is based on Marian NMT. Keeping the translator unchanged makes the comparison focus on the segmentation strategy rather than on differences between translation systems [9, 10].

DOI: 10.54254/2755-2721/2026.LD38489

3.1. Baseline methods

Two baseline methods are used for comparison. NOSPLIT directly translates the whole sentence and records the BLEU score and the number of suspected under-translated sentences. ALLSPLIT mechanically segments the sentence at all candidate boundaries, translates each segment independently, and then concatenates the translated segments. The average number of segments and the SPLIT Action Rate are recorded to evaluate the influence of excessive segmentation on translation performance [2, 5].

3.2. RLSEGMENT strategy segmentation

RLSEGMENT uses a reinforcement learning framework for selective sentence segmentation. A policy network is used to decide whether each candidate boundary should be split or kept. This strategy aims to balance information retention and under-translation control [5, 8].

3.2.1. Feature input

For each candidate boundary, the input features include token embedding, boundary position, punctuation type, and previous-action embedding. These features tell the policy network where the boundary is, what kind of punctuation it is, and what decision has just been made before the current position.

3.2.2. Reward function

The reward function is defined based on the sentence-level BLEU gain as follows:

$$R=BLEU_{hypothesis} - BLEU_{NOSPLIT} \quad (1)$$

This reward directly compares the BLEU score of the current segmentation plan with the NOSPLIT baseline. If the value is positive, the selected segmentation plan improves the sentence-level translation result [6].

3.2.3. Training and inference

During training, the policy network is trained for 10 epochs, with 1,800 episodes in each epoch. AdamW is used as the optimizer, and the learning rate is set to 3e-4. During inference, the trained policy network selects a segmentation plan for each input sentence. Segment caching and segmentation-plan caching are used to avoid repeated translation and improve inference efficiency.

3.2.4. RLSEGMENT process

To show the workflow of RLSEGMENT more clearly, this section describes its main running process, as shown in Figure 1.

DOI: 10.54254/2755-2721/2026.LD38489

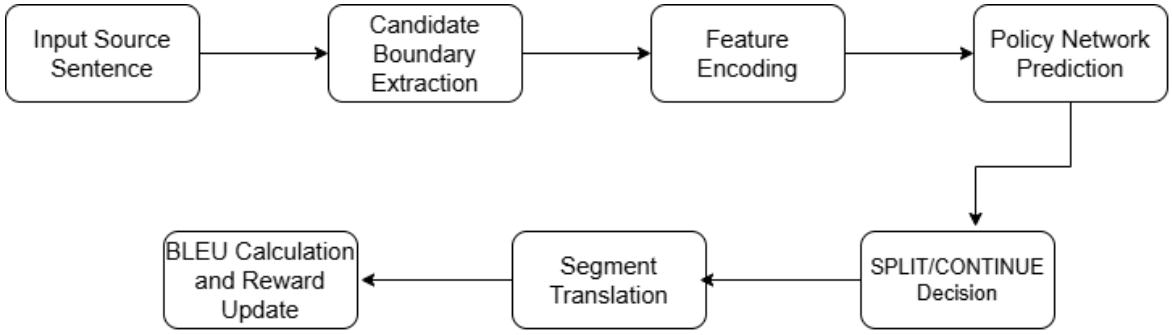


Figure 1. RLSEGMENT flowchart (picture credit: original)

The workflow shows how RLSEGMENT makes SPLIT or CONTINUE decisions at candidate boundaries based on boundary features and previous actions. This strategy helps the model avoid both no segmentation and full mechanical segmentation in long-sentence translation [5, 8].

4. Experimental results and analysis

4.1. Core indicators

Table 1: reports the test-set results of NOSPLIT, ALLSPLIT, and RLSEGMENT. The compared metrics include BLEU score, average number of segments, SPLIT Action Rate, number of positive-gain sentences, and number of under-translated sentences measured by hypothesis/reference length ratio < 0.8.

Table 1. Comparison of core indicators

Method	BLEU	Average Number of Segments	SPLIT Action Rate	Number of Positive-gain Sentences	Under-translated Sentences
NOSPLIT	10.0127	1	0	0	33
ALLSPLIT	9.9518	15	1.0	0	1
RLSEGMENT	11.9833	3.798	0.1995	613/1000	5

According to Table 1, RLSEGMENT obtains 11.9833 BLEU. NOSPLIT obtains 10.0127 BLEU, and ALLSPLIT obtains 9.9518 BLEU. Therefore, RLSEGMENT improves BLEU by about 1.97 points over NOSPLIT and about 2.03 points over ALLSPLIT.

The average number of segments in RLSEGMENT is 3.798, while ALLSPLIT has 15 segments on average. This means that RLSEGMENT does not split all candidate boundaries. At the sentence level, 613 out of 1,000 test samples obtain positive gains. The number of suspected under-translated sentences decreases from 33 in NOSPLIT to 5 in RLSEGMENT. ALLSPLIT has only 1 such case, but its BLEU score is lower than RLSEGMENT.

DOI: 10.54254/2755-2721/2026.LD38489

4.2. Analysis of key figures

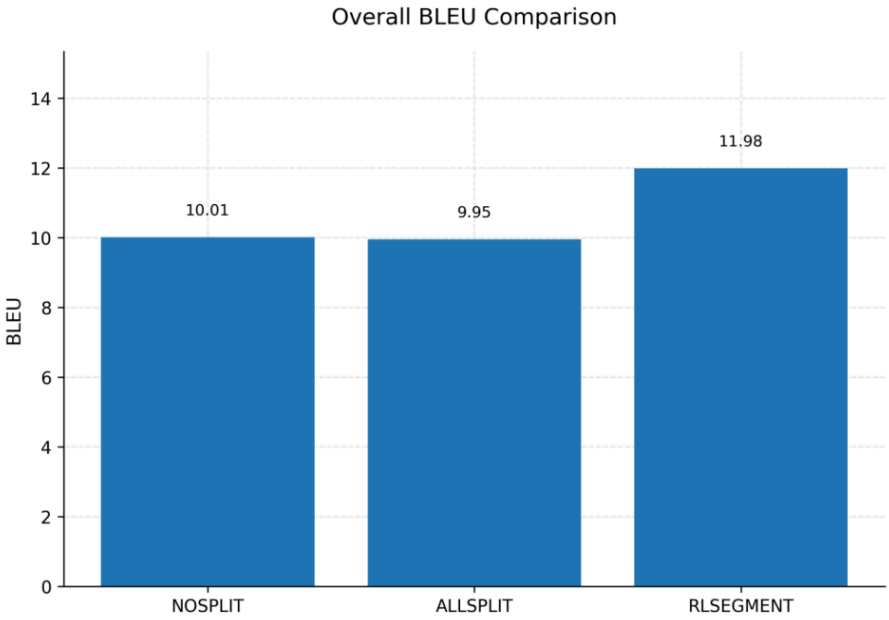


Figure 2. Overall BLEU comparison (picture credit: original)

Figure 2 shows the overall BLEU scores of the three methods. RLSEGMENT reaches 11.98 BLEU. NOSPLIT reaches 10.01 BLEU, and ALLSPLIT reaches 9.95 BLEU. The order in Figure 2 is the same as the result in Table 1.

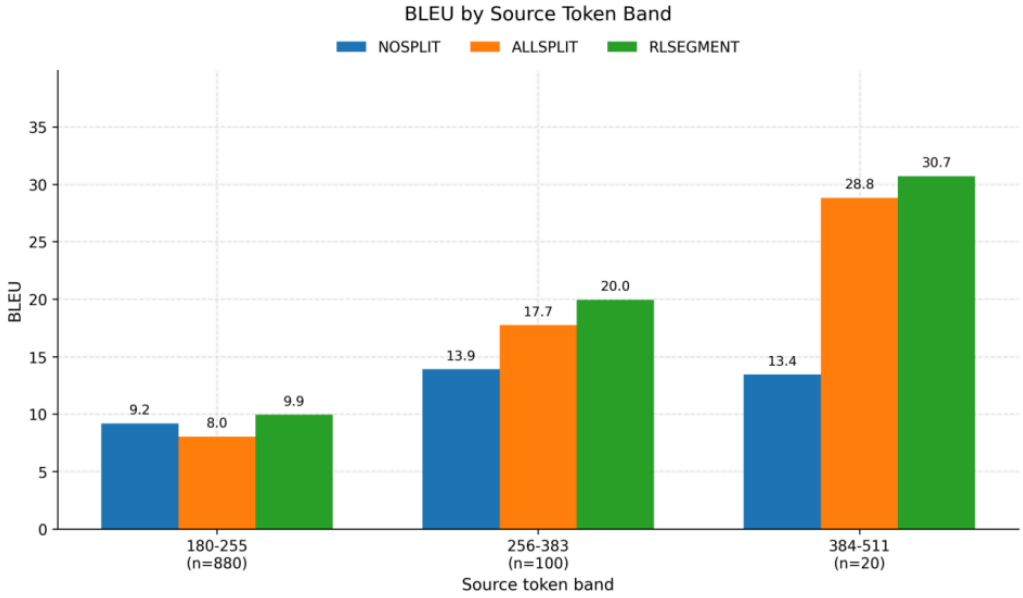


Figure 3. Comparison of BLEU scores across different sentence length intervals (picture credit: original)

Figure 3 shows the BLEU scores in different source-length intervals. In the 180–255 token interval, the difference among the three methods is small. In the 256–383 token interval,

DOI: 10.54254/2755-2721/2026.LD38489

RLSEGMENT is higher than both baselines. In the 384–511 token interval, the gap becomes larger. This pattern shows that the benefit of segmentation mainly appears in longer patent sentences.

4.3. Text analysis

The results show that RLSEGMENT is between NOSPLIT and ALLSPLIT in segmentation strength. NOSPLIT keeps the whole sentence unchanged, while ALLSPLIT splits every candidate boundary. RLSEGMENT selects only part of the candidate boundaries, so its average number of segments is much lower than ALLSPLIT but higher than NOSPLIT.

The best validation BLEU appears at epoch 7. This checkpoint is used for the final test evaluation. At this point, the training BLEU and validation BLEU are close, and no obvious training instability is observed.

5. Conclusion

This experiment compares three segmentation strategies for Chinese-English patent long-sentence translation: NOSPLIT, ALLSPLIT, and RLSEGMENT. In the test set, RLSEGMENT obtains the highest BLEU score. It improves BLEU by 1.97 points over NOSPLIT and by 2.03 points over ALLSPLIT. Its average number of segments is 3.8, and its SPLIT Action Rate is about 20%.

The length-band results show that the improvement is larger in the 256–383 and 384–511 token intervals. This result is consistent with the experimental goal, because the method is designed for long patent sentences. RLSEGMENT reduces suspected under-translation compared with NOSPLIT. However, ALLSPLIT has the lowest length-ratio warning count because it splits all candidate boundaries and tends to produce longer concatenated outputs.

Overall, RLSEGMENT is not a full-splitting method. It selects only part of the candidate boundaries and keeps the number of segments at a moderate level. In this experiment, this strategy gives the best BLEU score among the three methods. Future work can adjust the reward design for extremely long patent sentences and more complex sentence structures.

References

[1] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems, 30*.

[2] Koehn, P., & Knowles, R. (2017). Six challenges for neural machine translation. *Proceedings of the First Workshop on Neural Machine Translation*, 28-39.

[3] Goto, I., Chow, K. P., Lu, B., Sumita, E., & Tsou, B. K. (2013). Overview of the patent machine translation task at the NTCIR-10 workshop. *Proceedings of the 10th NTCIR Conference*.

[4] Long, Z., Utsuro, T., Mitsuhashi, T., & Yamamoto, M. (2016). Translation of patent sentences with a large vocabulary of technical terms using neural machine translation. *Proceedings of the 3rd Workshop on Asian Translation*, 47-57.

[5] Pouget-Abadie, J., Bahdanau, D., van Merriënboer, B., Cho, K., & Bengio, Y. (2014). Overcoming the curse of sentence length for neural machine translation using automatic segmentation. *Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation*, 78-85.

[6] Papineni, K., Roukos, S., Ward, T., & Zhu, W. J. (2002). BLEU: A method for automatic evaluation of machine translation. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, 311-318.

[7] Soares, F., Stevenson, M., Bartolome, D., & Zaretskaya, A. (2020). ParaPat: The multi-million sentences parallel corpus of patents abstracts. *Proceedings of the Twelfth Language Resources and Evaluation Conference*, 3769-3774.

[8] Srinivasan, S., & Dyer, C. (2021). Better Chinese sentence segmentation with reinforcement learning. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 293-302.

DOI: 10.54254/2755-2721/2026.LD38489

[9] Tiedemann, J., & Thottingal, S. (2020). OPUS-MT – Building open translation services for the world. *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, 479-480.
[10] Junczys-Dowmunt, M., Grundkiewicz, R., Dwojak, T., Hoang, H., Heafield, K., Neckermann, T., Seide, F., Germann, U., Aji, A. F., Bogoychev, N., Martins, A. F. T., & Birch, A. (2018). Marian: Fast neural machine translation in C++. *Proceedings of ACL 2018, System Demonstrations*, 116-121.