DrFrattn: Directly Learn Adaptive Policy from Attention for Simultaneous Machine Translation

Libo Zhao[1,2], **Jing Li**[1,3*], **Ziqian Zeng**[2*]
[1]Department of Computing, Hong Kong Polytechnic University
[2]Shien-Ming Wu School of Intelligent Engineering, South China University of Technology
[3]Research Centre for Data Science & Artificial Intelligence, Hong Kong Polytechnic University
libozh.zhao@connect.polyu.hk, jing-amelia.li@polyu.edu.hk, zqzeng@scut.edu.cn

Abstract

Simultaneous machine translation (SiMT) necessitates a robust read/write (R/W) policy to determine the optimal moments for translation, thereby balancing translation quality and latency. Effective timing in translation can align source and target tokens accurately. The attention mechanism within translation models inherently provides valuable alignment information. Building on this, previous research has attempted to modify the attention mechanism's structure to leverage its alignment properties during training, employing multi-task learning to derive the read/write policy. However, this multi-task learning approach may compromise the efficacy of the attention mechanism itself. This raises a natural question: why not directly learn the read/write policy from the well-trained attention mechanism? In this study, we propose DrFrattn, a method that directly learns adaptive policies from the attention mechanism. Experimental results across various benchmarks demonstrate that our approach achieves an improved balance between translation accuracy and latency.

1 Introduction

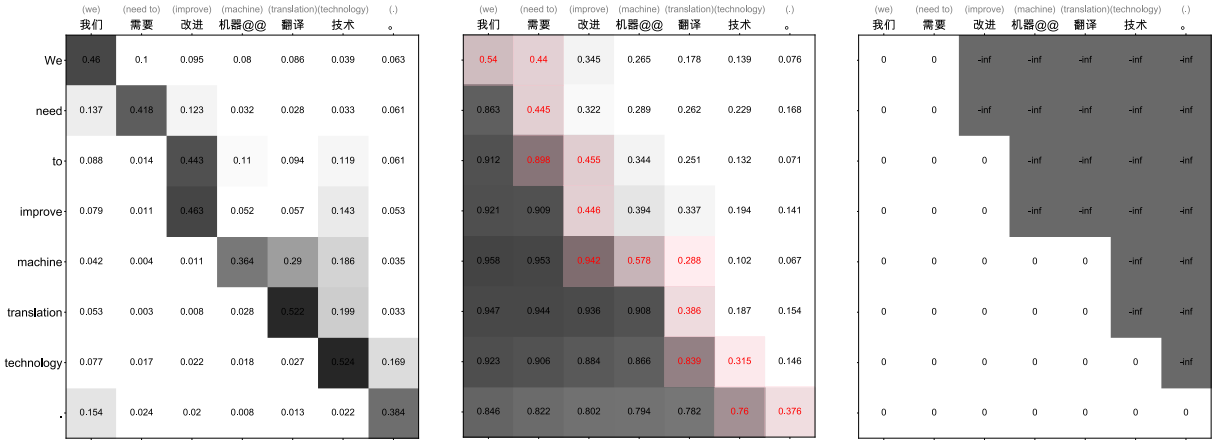
Simultaneous Machine Translation (SiMT) (Kolss et al., 2008; Gu et al., 2017) presents distinct challenges by generating target tokens in real-time while processing streaming source tokens. In contrast to traditional machine translation (MT) (Bahdanau et al., 2015; Vaswani et al., 2017), which has access to the entire source text, SiMT operates under a read/write (R/W) policy. This policy determines whether to produce target tokens immediately or delay output to wait for more source tokens. The read/write policies in simultaneous translation can be categorized into two types: prefixed and adaptive. Prefixed approaches, such as the wait-k policy (Ma et al., 2018; Elbayad et al., 2020; Zhang et al., 2021b), rely on simple, rule-based read/write decisions. While these methods are easier to implement, they typically yield limited translation outcomes, especially under low-latency scenarios. In contrast, adaptive methods (Gu et al., 2017; Dalvi et al., 2018; Zheng et al., 2019, 2020; Ma et al., 2020; Zhang and Feng, 2022c; Guo et al., 2023; Zhao and Zeng, 2024; Chen et al., 2024; Zhao et al., 2024), tailor read/write decisions based on the current contextual information, thereby achieving a more effective balance between translation quality and latency.

Optimal timing for WRITE operations in simultaneous translation often coincides with the precise alignment of an upcoming target token with an existing source token. In the Transformer model (Vaswani et al., 2017), the cross-attention mechanism is specifically designed to have this alignment capability, as demonstrated in Figure 1 (a). This feature enables the model to effectively match corresponding segments between the source and target languages, facilitating accurate and contextually aware translations. Consequently, numerous adaptive policies leveraging the cross-attention mechanism (Arivazhagan et al., 2019; Ma et al., 2020; Zhang et al., 2020; Zhang and Feng, 2022a,b; Zhang et al., 2022; Papi et al., 2023) have been developed. However, these methods typically involve modifying the attention mechanism's structure using multi-task training to attain the read/write policy, which may inadvertently compromise the overall effectiveness of the attention mechanism in capturing relevant features for translation tasks.

On another front, Zhao et al. recently introduced DaP-SiMT, an approach for deriving read/write decisions directly from supervised signals, providing a novel solution in the realm of simultaneous translation. This innovation and the existing limitations of the previous attention-based methods prompt a question: why not directly learn the read/write policy from the well-trained attention

* Corresponding author.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 34882–34895



(a) Cross Attention Matrix (b) Cumulative Attention Matrix (c) Cross Attention Mask

Figure 1: An Zh→En example of the Cross Attention Matrix, Cumulative Attention Matrix and Cross Attention Mask. The red elements denote a potential read/write path, determined by a predefined threshold λ (0.5 in this case).

mechanism itself? Inspired by this, we propose DrFrattn, a method to directly learn an adaptive read/write policy from attention. Specifically, we first conduct a systematic analysis of the feasibility of using the cross-attention matrices from various decoder layers of a Transformer as labels for read/write decisions. Subsequently, we employ lightweight parameters to model these automatically generated labels to develop a high-quality decision-maker. Additionally, due to the accessibility of cross-attention matrices during training, we can readily generate batches of cross-attention masks from these read/write paths, facilitating efficient prefix-to-prefix training of the translation model and enhancing translation performance. Our primary contributions are as follows:

1. We introduce DrFrattn, a novel method for learning adaptive policies directly from the cross-attention mechanism, where attention-based read/write supervision can be constructed automatically during forward propagation.
2. We devise a method for efficient prefix-to-prefix training for simultaneous translation models using the read/write paths and cross-attention masks derived from the cross-attention matrix generated during training.
3. Experiments on multiple benchmarks show that our method achieves a superior accuracy-latency trade-off.

2 Related Works

Adaptive policies optimize read/write operations by predicting these actions based on the current source and target prefixes, thereby enhancing the balance between latency and translation quality. The DaP-SiMT framework (Zhao et al., 2023) autonomously generates read/write supervisions by exploiting future information divergence to train a decision-making network. PsFuture (Zhao et al., 2024) proposes a novel zero-shot adaptive read/write policy, which utilizes the inherent capabilities of the translation model to make read/write decisions without any additional training.

Furthermore, numerous methods utilize the cross-attention mechanism, leveraging its capability to align corresponding source and target tokens, to develop adaptive read/write policies. Techniques such as MILK (Arivazhagan et al., 2019) and MMA (Ma et al., 2020) adaptively learn the real-time probabilities of write operations at specific moments through the attention mechanism. GMA (Zhang and Feng, 2022a), ITST (Zhang and Feng, 2022b), and wait-info (Zhang et al., 2022) employ the inherent alignment capability of the attention mechanism to predict aligned source positions, quantify waiting latency, and assess information weight, respectively, for crafting adaptive policies. The MU method (Zhang et al., 2020) constructs Meaningful Unit Chunking data based on the attention mechanism and makes read/write decisions by determining whether the current source part forms meaningful units. EDATT (Papi et al., 2023) directly utilizes attention for read/write decisions in simultaneous speech translation, achieving notable performance.

3 Preliminary

3.1 Full-sentence MT and SiMT

The Transformer architecture (Vaswani et al., 2017) addresses full-sentence translation by mapping a source-target pair $\mathbf{x} = (x_1, x_2, ..., x_N)$ and $\mathbf{y} = (y_1, y_2, ..., y_T)$ from input embeddings to latent spaces and then autoregressively generating the target sequence. This optimization targets the reduction of cross-entropy loss, represented by:

$$\mathcal{L}_{\text{mt}} = -\sum_{t=1}^{T} \log p\left(y_t \mid \mathbf{x}, \mathbf{y}_{<t}\right). \quad (1)$$

For SiMT, which employs a monotonic function $g(t)$ to determine the necessary source input to predict each subsequent target token, the loss function is:

$$\mathcal{L}_{\text{simt}} = -\sum_{t=1}^{T} \log p\left(y_t \mid \mathbf{x}_{\leq g(t)}, \mathbf{y}_{<t}\right). \quad (2)$$

3.2 Cross-attention Mechanism

Translation models leverage cross-attention to prioritize source elements contributing to each target token. Calculations utilize attention scores α_{ij} between target hidden states $\mathbf{s}$ and source states $\mathbf{z}$:

$$\alpha_{ij} = \text{softmax}\left(\frac{s_i W^Q \left(z_j W^K\right)^\top}{\sqrt{d_k}}\right), \quad (3)$$

where W^Q and W^K are projection parameters, and d_k is the dimension of inputs.

3.3 Prefix-to-Prefix Training and wait-k Policy

Prefix-to-Prefix Training (P2P) is pivotal in SiMT for predicting target tokens from limited source prefixes. Studies (Ma et al., 2018) have shown suboptimal performances without P2P training, particularly at low latency scenarios.

Wait-k policy (Ma et al., 2018), the most widely used fixed policy, commences by processing k source tokens and subsequently alternating between WRITE and READ action. The function $g(t)$ for the wait-k policy is:

$$g_{waitk}(t; k) = \min\{t + k - 1, N\}. \quad (4)$$

Multi-path Wait-k (Elbayad et al., 2020) is an efficient technique for wait-k training. It randomly samples different k values between batches during model optimization. By employing a unidirectional attention encoder with a tailored upper triangular masked cross-attention mechanism, the multi-path wait-k model not only enables efficient prefix-to-prefix training, but also supports incremental decoding during inference, thereby avoiding repeated re-encoding of prefix tokens with each new source token and substantially reducing computational overhead. After introducing the cross-attention mask M, the computation of cross-attention in a Transformer model can be calculated as follows:

$$\tilde{\alpha}_{ij} = \text{softmax}\left(\frac{s_i W^Q \left(z_j W^K\right)^\top}{\sqrt{d_k}} + m_{ij}\right), \quad (5)$$

$$m_{ij} = \begin{cases} 0 & \text{if } g(t) \geq i, \\ -\infty & \text{if } g(t) < i. \end{cases} \quad (6)$$

4 Method

4.1 Cross-Attention Analysis

Potential of Cross-attention in Guiding R/W Decisions Many previous approaches leverage the inherent alignment capability of cross-attention to obtain the read/write policy through multi-task training. This inspires us to explore the potential of cross-attention itself as a supervisory signal for simultaneous translation read/write decisions. As illustrated in Figure 1, we visualize the cross-attention matrix of a widely adopted multi-path wait-k translation model on one Zh→En example. By applying simple transformations to the cross-attention matrix, as specified in Equation 7, the Cumulative Attention Matrix C can be derived. Based on this matrix, a high-quality read/write path can be determined by an appropriate predefined threshold, as shown in Figure 1 (b). Specifically, starting from the top-left corner, a reading operation is performed when the value exceeds the threshold, otherwise, a writing operation is executed. This demonstrates the potential of the Cumulative Attention Matrix as the supervisory signal for read/write decisions.

$$c_{ij} = 1 - \sum_{k=1}^{j} \alpha_{ik}, \quad \forall i, j. \quad (7)$$

Which Layer's Cross-Attention is Optimal for R/W Decisions? Transformer models (Vaswani et al., 2017) consist of multiple decoder layers, each equipped with multi-head attention mechanisms. Every layer and each head within it has its own cross-attention mechanisms, which adaptively differentiate learning objectives during training to

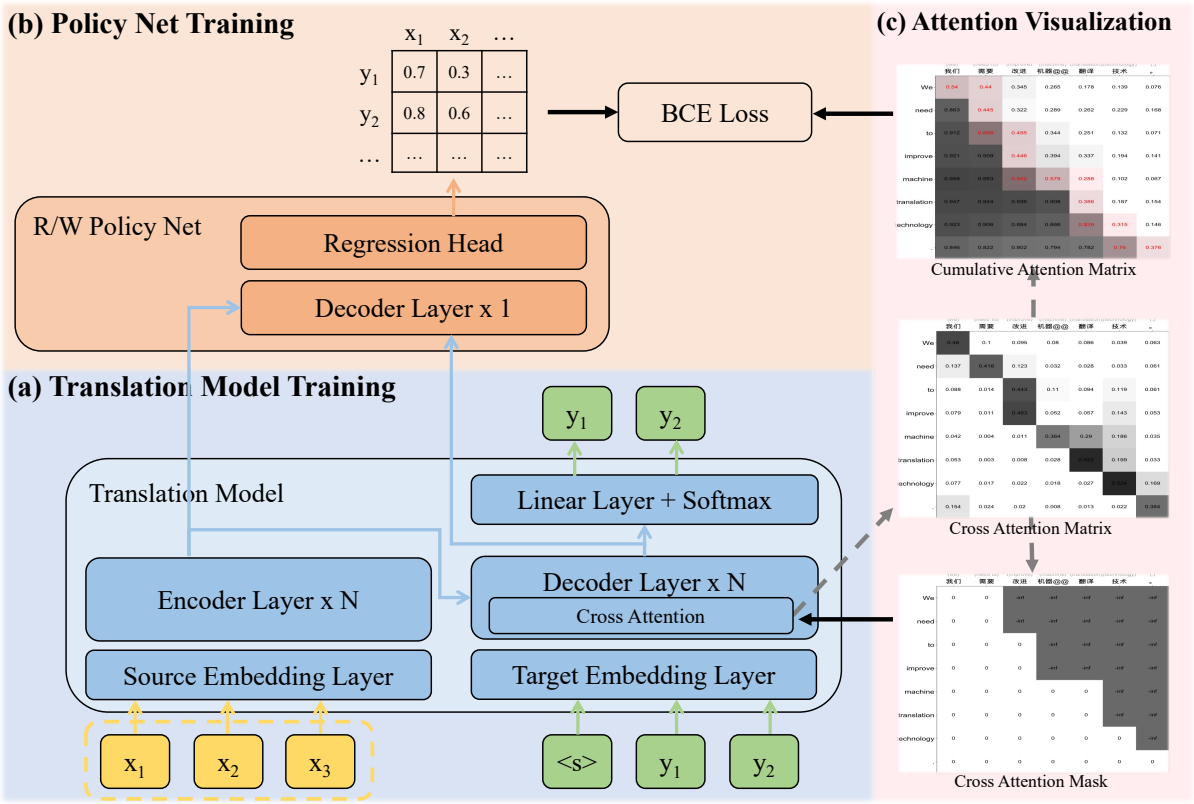


Figure 2: An overall schematic of the proposed DrFrattn method.

capture varying levels of representations. Therefore, it is necessary to explore which attention is optimal for guiding read/write decisions. To simplify the problem, and inspired by the success of EDATT (Papi et al., 2023) in real-time speech translation, we opt to average the attention across all heads within the same layer when selecting attention heads.

To quantify which layer's cross-attention produces the highest-quality read/write paths, we utilize the Negative Log-Likelihood (NLL) vs. Average Lagging (AL) curve. AL(Ma et al., 2018) is a widely used metric for translation latency. For a given parallel sentence, we derive the read/write path, denoted as $\{g(1), g(2), \ldots, g(T)\}$, under the read/write policy and calculate the negative log-likelihood (NLL) of the translation along the path, as defined by Equation. (2). By aggregating these NLL scores and their corresponding AL scores across the dataset, we generate NLL vs. AL curves for different read/write policies. As illustrated in Figure 3, we plot the NLL vs. AL curves for paths obtained from the Cumulative Attention Matrix of different layers and the wait-k method, based on the multi-path wait-k model on the Zh→En validation dataset. The results show that layer 5 (out

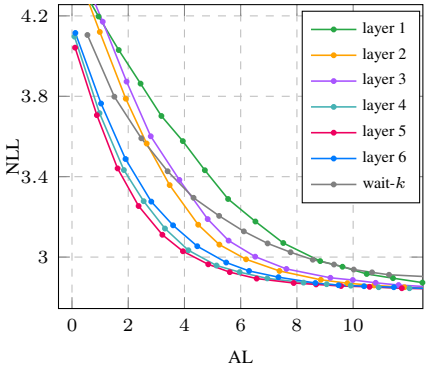


Figure 3: NLL vs. AL curves derived from different decoder layers.

of 6 layers) achieves the best performance, with significantly higher read/write path quality compared to the wait-k method. This finding further validates the potential of the Cumulative Attention Matrix as the supervisory signal for guiding read/write decisions. In this work, all supervised signals for read/write decisions are derived from the 5th decoder layer.

Improving R/W Supervision Signals Using Temperature-Adjusted Softmax We observe that the attention distribution in some examples is insufficiently focused, leading to unclear boundaries

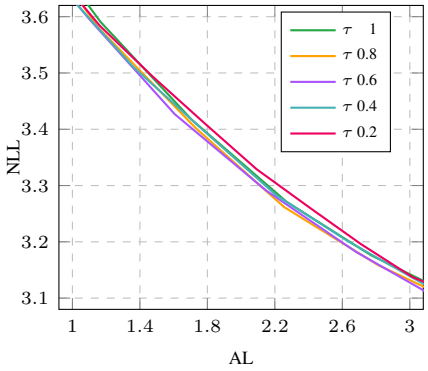


Figure 4: NLL vs. AL curves caculated based on different softmax temperature τ.

in the Cumulative Attention Matrix for certain tokens and resulting in ambiguous read/write decisions. To address this issue, we use the temperature-adjusted Softmax with a parameter τ when computing cross-attention, as shown in Equation 8. When $\tau < 1$, the attention distribution becomes sharper and more focused, while $\tau = 1$ reduces the formulation to the standard Softmax. We also employ the NLL vs. AL metric to explore the optimal temperature parameter τ. As illustrated in Figure 4, exploration experiments on the Zh→En valid dataset using the multi-path wait-k model show that a temperature of 0.6 achieves the best NLL score for low-latency read/write paths. Similar results are observed on other datasets, and in this study, we use a temperature of 0.6 to construct read/write supervision signals.

$$\text{Softmax}_\tau(z_i) = \frac{\exp\left(\frac{z_i}{\tau}\right)}{\sum_j \exp\left(\frac{z_j}{\tau}\right)}. \quad (8)$$

4.2 The DrFrattn Policy Net

Since the complete Cumulative Attention Matrix is inaccessible during inference because it requires the full source sentence, which is unavailable during simultaneous translation, we follow DaP-SiMT (Zhao et al., 2023) and introduce an additional decoder layer to model the read/write supervision signal, thereby deriving the read/write policy net. During policy network training, the parameters of the backbone translation model remain frozen. As illustrated in Figure 2 (b), for each parallel sentence pair, we compute a predicted cumulative attention matrix and a ground truth cumulative attention matrix. The predicted matrix is optimized toward the ground truth using the binary cross-entropy (BCE) loss.

4.3 Cross-Attention-Based Shift-k Training

The above Cumulative Attention Matrix serves as an easily obtainable supervision signal for read/write decisions, which can be batch-extracted during the forward propagation of the translation model. This enables the efficient computation of read/write paths for each parallel corpus pair and the corresponding cross-attention masks. These motivate us to employ the masks in prefix-to-prefix training to reduce the gap between simultaneous translation training and inference, thereby improving performance.

Specifically, given a Cumulative Attention Matrix and a sampled threshold λ within a predefined range $[\lambda_1, \lambda_2]$, a read/write path $\{g_{\text{DrFrattn}}(1), g_{\text{DrFrattn}}(2), \ldots, g_{\text{DrFrattn}}(T)\}$ can be derived. To prevent hallucinations caused by an excessively large λ, which can result in premature writing operations before sufficient source information has been received, we utilize the wait-1 path as the safeguard read/write path, formulated as follows:

$$g_{\text{DrFrattn}}(t) = \max\{g_{\text{DrFrattn}}(t), g_{\text{wait1}}(t)\}. \quad (9)$$

Drawing on the multi-path wait-k method, where uniform sampling of the value of k enables robust performance across various latency levels, we propose a shift-k training approach. Notably, the derived read/write path g_{DrFrattn} is randomly shifted to the right by k tokens, resulting in a new path:

$$\tilde{g}_{\text{DrFrattn}}(t) = \min\{g_{\text{DrFrattn}}(t) + k, |X|\}, \quad (10)$$

$$k \sim \text{Uniform}(\{0, 1, \ldots, |X|\}). \quad (11)$$

The shifted path is subsequently used to compute the cross-attention mask M_{DrFrattn} based on the Equation 6 and the mask is incorporated into the training process of the simultaneous translation model, as shown in Figure 2 (a).

It is worth mentioning that the threshold λ directly influences the delay level of the read/write path, with smaller λ values corresponding to paths with greater delays. By adjusting the range of λ values, $[\lambda_1, \lambda_2]$, it is possible to control the delay range of the read/write path to some extent. However, since the relationship between λ and the delay level is nonlinear and difficult to determine precisely, we introduce the shift-k operation to achieve finer control over delay levels in the training process. This also allows the read/write path to cover a broader range of translation situations, leading

to a more robust simultaneous translation model. Specifically, when $[\lambda_1, \lambda_2]$ is adjusted to values greater than 1, the wait-1 safeguard path is consistently triggered. Combined with the shift-k operation, the read/write path degenerates into the conventional wait-k path. When $[\lambda_1, \lambda_2]$ is adjusted to values less than 0, the read/write path transitions into an offline translation path. Therefore, by appropriately tuning $[\lambda_1, \lambda_2]$, the proposed shift-k training method can simulate diverse read/write paths. In Section 6.1, we further investigate the impact of the hyperparameters λ_1 and λ_2 on results.

4.4 Overall Procedure of the DrFrattn Method

The overall procedure of applying the proposed DrFrattn method is summarized as follows:

1. **(Optional) Training the Simultaneous Translation Backbone Model** First, the proposed shift-k training method is used to train a simultaneous translation model from scratch, as illustrated in Figure 2 (b). Alternatively, any pre-trained simultaneous translation model with a cross-attention mechanism, such as the multi-path wait-k model, can be employed as the backbone model.
2. **Training the R/W Policy Net** Based on the backbone model, an additional decoder layer is introduced to fit the read/write supervision signals discussed in Section 4.1, as shown in Figure 2 (a), while keeping the backbone model's parameters frozen. Although an independent policy network can also be used to fit these supervision signals, this work focuses on the additional decoder layer for efficient training.
3. **Inferring** Based on the trained policy net and the backbone translation model, simultaneous translation tasks can be performed. Additionally, following (Zhao et al., 2023), another hyperparameter is introduced in the read/write decision-making process to limit the maximum number of consecutive READ operations for certain languages, thereby improving their performance. The inference process is summarized in Appendix C.

5 Experiments

5.1 Datasets

WMT2022 Zh→En[1]. We use a subset with 25M sentence pairs for training[2], from which 1500 unique sentence pairs are extracted as the validation set. We first tokenize the Chinese and English data using the Jieba Chinese Segmentation Tool[3] and Moses[4], respectively, and then apply BPE with 32000 merge operations. We employ the dev set of 956 sentence pairs from BSTC (Zhang et al., 2021a) as the test set.

WMT15 De→En[5]. All 4.5M sentence pairs from this dataset are used for training, and are tokenized using 32K BPE merge operations. We use newstest2013 (3000 sentence pairs) for validation and report results on newstest2015 (2169 sentence pairs).

IWSLT15 En→Vi[6]. All 133K sentence pairs from this dataset (Luong and Manning, 2015) are used for training. We use TED tst2012 (1553 sentence pairs) for validation and TED tst2013 (1268 sentence pairs) as the test set. Following the settings in (Ma et al., 2020), we adopt word-level tokenization and replace rare tokens (frequency < 5) with <unk>. The vocabulary sizes are 17K for English and 7.7K for Vietnamese, respectively.

5.2 System Settings

The models used in our experiments are introduced as follows. To ensure a fair comparison, all our implementations are adapted from the Fairseq Library (Ott et al., 2019) and we carefully select strong baseline systems. All methods are built based on Transformer(Vaswani et al., 2017) with a unidirectional encoder—and employ a cross-attention mask (as described in Equation 5) during forward propagation to enable efficient prefix-to-prefix training.

Multi-path Wait-k (Elbayad et al., 2020): a fixed policy, which improves wait-k by randomly sampling different k during training.

ITST (Zhang and Feng, 2022b): an adaptive policy, which models the SiMT task as a transport problem of information from source to target.

DaP-SiMT (Zhao et al., 2023): an adaptive policy, which learns from automatically constructed

[1] www.statmt.org/wmt22

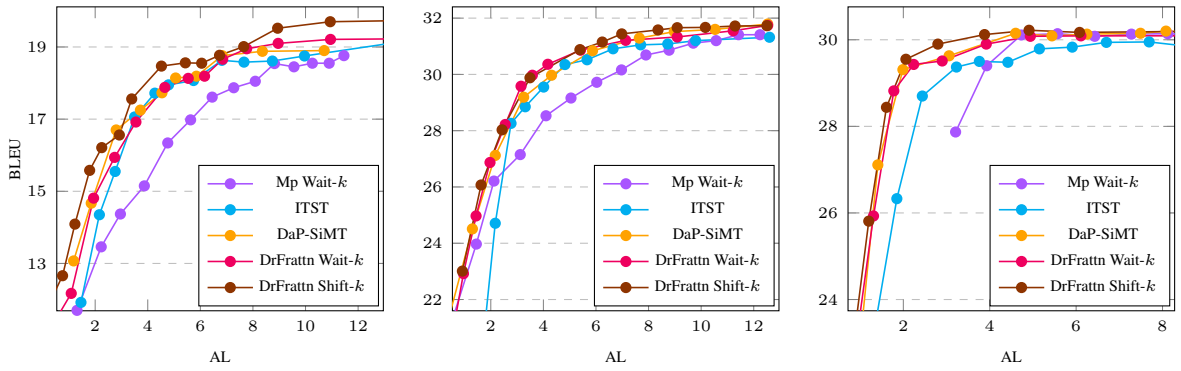
[2] The data sources include casia2015, casict2011, casict2015, datum2015, datum2017, neu2017, News Commentary V16, ParaCrawl V9.

[3] https://github.com/fxsjy/jieba

[4] https://github.com/moses-smt

[5] www.statmt.org/wmt15

[6] nlp.stanford.edu/projects/nmt



(a) Zh→En, Transformer-Big (b) De→En, Transformer-Base (c) En→Vi, Transformer-Small

Figure 5: Comparison of BLEU vs. AL curves between multi-path (abbreviated as Mp) wait-k, ITST, DaP-SiMT, and our proposed DrFrattn approach on three language pairs.

read/write supervision signals, inspired by the translation behavior of human experts.

For the Zh→En experiments, we utilize the transformer big architecture, while the base and small architectures are used for De→En and En→Vi experiments respectively. In relation to the threshold range $[\lambda_1, \lambda_2]$ used during the training process of the main experiments, we select the configurations that yield the best performance on the validation set, $[0.4, 0.85]$ for Zh→En, $[0.4, 0.85]$ for De→En, and $[-0.15, 0.85]$ for En→Vi, respectively.

For evaluation, following ITST and DaP-SiMT, we report case-insensitive BLEU (Papineni et al., 2002) scores to assess translation quality and Average Lagging (AL/token) (Ma et al., 2018) to measure latency. Regarding the maximum number of continuous read actions in our method, we empirically select the best-performing configurations, which are no constraint, 4, no constraint for Zh→En, De→En, En→Vi respectively.

5.3 Main Results

We compare the proposed DrFrattn method with previous approaches for three language pairs, as shown in Figure 5. DrFrattn Wait-k and DrFrattn Shift-k correspond to translation models based on the multi-path wait-k model and the cross-attention-based shift-k training method introduced in Section 4.3, respectively.

First, it is evident that the DrFrattn Wait-k experiment outperforms the multi-path wait-k experiment by a substantial margin. With both the backbone translation models being the multi-path wait-k model, the proposed DrFrattn adaptive policy demonstrates superior performance compared to the fixed wait-k policy, effectively exploiting the latent translation potential of the backbone model. Additionally, DrFrattn Wait-k performs comparably to previous top SiMT methods, DaP-SiMT and ITST, and even surpasses them in certain latency scenarios, further validating the effectiveness of the proposed DrFrattn policy.

Second, the DrFrattn Shift-k experiment exhibits more robust results across all translation directions. Particularly in the Zh→En experiment, DrFrattn Shift-k significantly surpasses all other methods. This highlights the efficacy of our cross-attention-based shift-k training method. This training approach fully leverages the inherent accessibility of cross-attention in the model training process and the efficacy of read/write paths derived from it. Furthermore, the random sampling of the shift-k value across samples facilitates more extensive coverage of different latency scenarios, thereby enhancing the model's robustness across various conditions.

6 Analysis

6.1 Ablation Study

Effect of the temperature parameter In Section 4.1, we mentioned that to improve the distribution of the cumulative attention matrix and achieve better corresponding read/write paths, we incorporate a temperature parameter τ in the softmax function during cross-attention computation. In this part, we conduct experiments to evaluate the impact of the temperature parameter τ on the final translation performance. Figure 6 presents the experiment results for De→En and En→Vi translation tasks. Compared to the default temperature value of 1, setting τ to 0.6 consistently improves BLEU scores across all translation directions and

AL	1.5±0.1	2.5±0.1	3.5±0.1	4.3±0.1	5.2±0.1	6.2±0.1	7.8±0.1	8.5±0.1	9.9±0.1
DrFrattn Accuracy	**0.921**	**0.917**	**0.919**	**0.907**	**0.908**	**0.905**	**0.904**	**0.909**	**0.915**
Wait-k Accuracy	0.841	0.843	0.825	0.812	0.819	0.806	0.803	0.819	0.822

Table 1: The prediction accuracy of the DrFrattn policy net compared with the wait-k method

latency scenarios. This demonstrates the effectiveness of the temperature parameter τ, which enhances the quality of read/write supervision signals and leads to a better policy network.

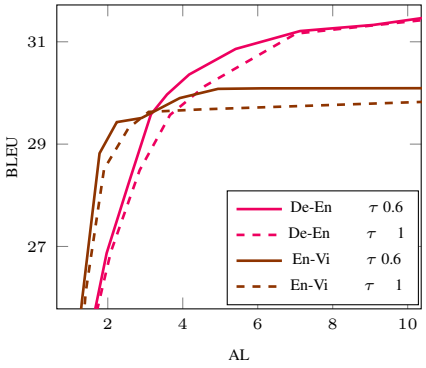


Figure 6: BLEU vs. AL curves comparing among DrFrattn Wait-k experiments with varying softmax temperature τ.

Effect of the shift-k training threshold range In Section 4.3, we introduced the hyperparameter threshold range $[\lambda_1, \lambda_2]$ to control the variability of read/write path latencies during shift-k training. To prevent meaningless read/write paths in cases of excessively low latency, we set λ_2 to 0.85, which corresponds to a read/write latency with the AL range approximately from 0 to 1. Especially, when λ value is less than 0, the read/write path is considered the offline scenario, allowing us to set λ_1 as a negative value to introduce a certain proportion of offline training. Figure 7 illustrates the impact of the threshold range on the En→Vi experiment. It shows that, regardless of the hyperparameter settings, the model performs well in low-latency scenarios. However, when λ_1 is set to -0.15, the model achieves better performance in mid- to high-latency conditions. We hypothesize that the inclusion of offline training improves the translation capability of the En→Vi model. This experiment demonstrates that, when applying the shift-k method, carefully tuning the threshold range hyperparameter can enhance translation performance.

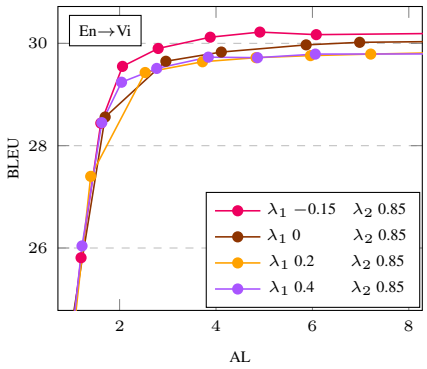


Figure 7: Effect of the cross-attention-based shift-k training threshold range $[\lambda_1, \lambda_2]$.

6.2 The Prediction Accuracy of the Read/Write Policy Net

In the DrFrattn method, the read/write policy network plays a critical role in ensuring appropriate read/write decisions, which in turn balances translation quality and latency effectively. This section examines the prediction accuracy of the policy network. It is important to note that the values predicted by the policy net, like those in the Cumulative Attention Matrix shown in Figure 1 (b), are continuous. Our primary focus is not on the precise prediction of these values, but rather on whether suitable read/write paths can be obtained based on the prediction matrix, given similar read/write latencies. We aim to quantify the differences between the read/write paths derived from the prediction matrix and the ground truth matrix. Therefore, we use the area accuracy for evaluating the similarity of the two paths by aligning them on the same matrix and calculating the area enclosed by both paths. The accuracy is then calculated as follows:

$$Area = \sum_{t=1}^{T} |g_{pred}(t) - g_{ground}(t)| \quad (12)$$

$$Accuracy = 1 - \frac{Area}{|\mathbf{x}||\mathbf{y}|} \quad (13)$$

When the predicted path exactly matches the ground truth, the enclosed area is zero, resulting in an accuracy of 1. We evaluate accuracy on the Zh→En test set at different AL values and compare it with the wait-k path. Results in Table 1

show that the well-trained policy network in the DrFrattn method significantly outperforms the wait-k method, demonstrating the network's ability to learn and apply valuable contextual information for making accurate read/write decisions.

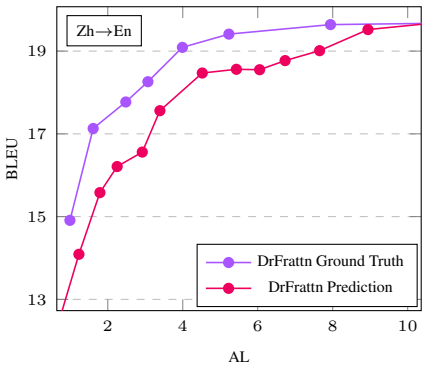


Figure 8: BLEU vs. AL curves comparing between DrFrattn with ground truth cumulative attention value and standard DrFrattn with predicted cumulative attention value.

6.3 Upper Bound of the DrFrattn Policy Net

The evaluation of DrFrattn Policy Net's upper bound performance quantifies the impact of modeling inaccuracies. This is done by substituting the predicted cumulative attention values with the true ones, computed based on the complete source sentence using Equation 3 and 7 during inferring. As shown in Figure 8, DrFrattn's upper bound performance, notably outperforms the results from the learned policy model. This indicates that the proposed method still has substantial potential for improvement, with significant scope to further approach the upper bound and thereby achieve better performance.

7 Conclusion

In this paper, we present the DrFrattn policy for simultaneous translation tasks, a novel approach for learning adaptive policies directly from the cross-attention mechanism. Additionally, we propose an innovative method for efficient and effective prefix-to-prefix training for simultaneous translation models. Experimental results across multiple benchmarks demonstrate that DrFrattn achieves an optimal balance between translation quality and latency, outperforming previous top approaches in SiMT.

Limitations

In this work, the proposed DrFrattn method does not allow for the joint training of the backbone translation model and the read/write policy, unlike many existing methods. This separation of training processes introduces additional complexity to the overall training workflow. However, this approach also offers increased flexibility, as the DrFrattn policy can be applied to any attention-based backbone translation model, allowing it to be integrated into a variety of architectures. While the split-phase training procedure may make the training process more cumbersome, it significantly enhances the adaptability of the method, enabling the use of DrFrattn with different translation models without requiring model-specific modifications.

Ethics Statement

After careful review, to the best of our knowledge, we have not violated the ACL Ethics Policy.

Acknowledgements

This work was supported by the Research Grants Council of the Hong Kong Special Administrative Region, China (Project No. PolyU/25200821), the Innovation and Technology Fund (Project No. PRP/047/22FX), the PolyU Internal Fund from RC-DSAI (Project No. 1-CE1E), a gift fund from Huawei (N-ZGM3), the National Natural Science Foundation of China (No. 62406114), the Fundamental Research Funds for the Central Universities (2024ZYGXZR074), Guangdong Basic and Applied Basic Research Foundation (2025A1515011413).

References

Naveen Arivazhagan, Colin Cherry, Wolfgang Macherey, Chung-Cheng Chiu, Semih Yavuz, Ruoming Pang, Wei Li, and Colin Raffel. 2019. Monotonic infinite lookback attention for simultaneous machine translation. *arXiv preprint arXiv:1906.05218*.

Dzmitry Bahdanau, Kyung Hyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015*.

Xinjie Chen, Kai Fan, Wei Luo, Linlin Zhang, Libo Zhao, Xinggao Liu, and Zhongqiang Huang. 2024. Divergence-guided simultaneous speech translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 17799–17807.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, and Stephan Vogel. 2018. Incremental decoding and training methods for simultaneous translation in neural machine translation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 493–499, New Orleans, Louisiana. Association for Computational Linguistics.

Maha Elbayad, Laurent Besacier, and Jakob Verbeek. 2020. Efficient wait-k models for simultaneous machine translation. *arXiv preprint arXiv:2005.08595*.

Jiatao Gu, Graham Neubig, Kyunghyun Cho, and Victor O.K. Li. 2017. Learning to translate in real-time with neural machine translation. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1053–1062, Valencia, Spain. Association for Computational Linguistics.

Shoutao Guo, Shaolei Zhang, and Yang Feng. 2023. Learning optimal policy for simultaneous machine translation via binary search. *arXiv preprint arXiv:2305.12774*.

Muntsin Kolss, Stephan Vogel, and Alex Waibel. 2008. Stream decoding for simultaneous spoken language translation. In *INTERSPEECH*, pages 2735–2738.

Minh-Thang Luong and Christopher D Manning. 2015. Stanford neural machine translation systems for spoken language domains. In *Proceedings of the 12th International Workshop on Spoken Language Translation: Evaluation Campaign*.

Mingbo Ma, Liang Huang, Hao Xiong, Renjie Zheng, Kaibo Liu, Baigong Zheng, Chuanqiang Zhang, Zhongjun He, Hairong Liu, Xing Li, et al. 2018. Stacl: Simultaneous translation with implicit anticipation and controllable latency using prefix-to-prefix framework. *arXiv preprint arXiv:1810.08398*.

Xutai Ma, Juan Miguel Pino, James Cross, Liezl Puzon, and Jiatao Gu. 2020. Monotonic multihead attention. In *International Conference on Learning Representations*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Sara Papi, Matteo Negri, and Marco Turchi. 2023. Attention as a guide for simultaneous speech translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13340–13356, Toronto, Canada. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. ACL '02, page 311–318, USA. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Ruiqing Zhang, Xiyang Wang, Chuanqiang Zhang, Zhongjun He, Hua Wu, Zhi Li, Haifeng Wang, Ying Chen, and Qinfei Li. 2021a. Bstc: A large-scale chinese-english speech translation dataset. *arXiv preprint arXiv:2104.03575*.

Ruiqing Zhang, Chuanqiang Zhang, Zhongjun He, Hua Wu, and Haifeng Wang. 2020. Learning adaptive segmentation policy for simultaneous translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2280–2289, Online. Association for Computational Linguistics.

Shaolei Zhang and Yang Feng. 2022a. Gaussian multi-head attention for simultaneous machine translation. *Preprint*, arXiv:2203.09072.

Shaolei Zhang and Yang Feng. 2022b. Information-transport-based policy for simultaneous translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, Online and Abu Dhabi. Association for Computational Linguistics.

Shaolei Zhang and Yang Feng. 2022c. Reducing position bias in simultaneous machine translation with length-aware framework. *arXiv preprint arXiv:2203.09053*.

Shaolei Zhang, Yang Feng, and Liangyou Li. 2021b. Future-Guided Incremental Transformer for Simultaneous Translation. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(16):14428–14436.

Shaolei Zhang, Shoutao Guo, and Yang Feng. 2022. Wait-info Policy: Balancing Source and Target at Information Level for Simultaneous Machine Translation. *Preprint*, arXiv:2210.11220.

Libo Zhao, Kai Fan, Wei Luo, Wu Jing, Shushu Wang, Ziqian Zeng, and Zhongqiang Huang. 2023. Adaptive policy with wait-k model for simultaneous translation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4816–4832, Singapore. Association for Computational Linguistics.

Libo Zhao, Jing Li, and Ziqian Zeng. 2024. PsFuture: A pseudo-future-based zero-shot adaptive policy for simultaneous machine translation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1869–1881, Miami, Florida, USA. Association for Computational Linguistics.

Libo Zhao and Ziqian Zeng. 2024. Dap-simt: divergence-based adaptive policy for simultaneous machine translation. *International Journal of Machine Learning and Cybernetics*, pages 1–20.

Baigong Zheng, Kaibo Liu, Renjie Zheng, Mingbo Ma, Hairong Liu, and Liang Huang. 2020. Simultaneous translation policies: From fixed to adaptive. *arXiv preprint arXiv:2004.13169*.

Baigong Zheng, Renjie Zheng, Mingbo Ma, and Liang Huang. 2019. Simpler and faster learning of adaptive policies for simultaneous translation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1349–1354, Hong Kong, China. Association for Computational Linguistics.

A Case Study

Here, we present specific cases to demonstrate the effectiveness of the proposed method, as illustrated in Figure 9 and Figure 10. It can be observed that at certain time steps, the proposed DrFrattn method makes more reasonable read/write decisions than the previous top SiMT approach, DaP-SiMT, enabling accurate translation with lower latency.

Source	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话	动作	。		
	(with)		(dialogue)	(state)		(we)		(can)	(go)	(based on)	(some)	(bool)	(rule)	(go)	(trigger)	(this)	(conversation)	(action)	(.)		
Step	**Streaming Inputs**																				**Outputs**
1	有	了																			With
2	有	了	对话	状态																	the
3	有	了	对话	状态																	dialogue
4	有	了	对话	状态	呢	我们															state
5	有	了	对话	状态	呢	我们															we
6	有	了	对话	状态	呢	我们	就	可以													can
7	有	了	对话	状态	呢	我们	就	可以	去	基于											go
8	有	了	对话	状态	呢	我们	就	可以	去	基于											to
9	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个					trigger
10	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个					this
11	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				conversation
12	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				based
13	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				on
14	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				some
15	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				Bo@@
16	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				ol@@
17	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				ean
18	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				rules
19	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				.
20	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话	动作	。	<eos>	<eos>

Figure 9: Case No.226 in BSTC Zh→En test set, evaluated using DaP-SiMT method, with $\lambda = 0.26$.

Source	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话	动作	。		
	(with)		(dialogue)	(state)		(we)		(can)	(go)	(based on)	(some)	(bool)	(rule)	(go)	(trigger)	(this)	(conversation)	(action)	(.)		
Step	**Streaming Inputs**																				**Outputs**
1	有	了																			With
2	有	了	对话	~~状态~~																	the
3	有	了	对话	状态																	dialogue
4	有	了	对话	状态	呢	~~我们~~															state
5	有	了	对话	状态	呢	我们															we
6	有	了	对话	状态	呢	我们	就	可以													can
7	有	了	对话	状态	呢	我们	就	可以	去	基于											go
8	有	了	对话	状态	呢	我们	就	可以	去	基于											to
9	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	~~这个~~					trigger
10	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个					this
11	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				conversation
12	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				based
13	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				on
14	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				some
15	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				Bo@@
16	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				ol@@
17	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				ean
18	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				rules
19	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话				.
20	有	了	对话	状态	呢	我们	就	可以	去	基于	一些	布尔	规则	去	触发	这个	对话	动作	。	<eos>	<eos>

Figure 10: Case No.85 in BSTC Zh→En test set, evaluated using DrFrattn shift-k method, with $\lambda = 0.5$. The red text in strikethrough in the table indicates where the DrFrattn method makes better read/write decisions during inference compared to the DaP-SiMT approach.

B Numerical Results

The numerical main results are presented in Table 2.

	Main Results (Figure 5)									
	Mp Wait-k		ITST		DaP-SiMT		DrFrattn wait-k		DrFrattn shift-k	
	AL	BLEU	AL	BLEU	AL	BLEU	AL	BLEU	AL	BLEU
Zh→En	1.31	11.7	0.7	8.91	1.18	13.07	1.09	12.17	0.76	12.66
	2.23	13.46	1.46	11.92	1.85	14.67	1.94	14.81	1.23	14.09
	2.96	14.37	2.16	14.35	2.8	16.7	2.74	15.94	1.79	15.58
	3.87	15.15	2.76	15.55	3.72	17.25	3.55	16.92	2.92	16.56
	4.76	16.34	3.5	17.06	4.54	17.73	4.65	17.88	3.39	17.56
	5.63	16.98	4.27	17.72	5.06	18.14	5.54	18.13	4.52	18.47
	6.45	17.61	4.79	17.95	5.85	18.19	6.85	18.66	5.43	18.56
	7.27	17.87	5.74	18.07	6.83	18.76	7.76	18.95	6.73	18.77
	8.09	18.05	6.82	18.63	8.36	18.88	8.96	19.1	7.65	19.01
	8.82	18.54	7.66	18.58	10.71	18.9	10.95	19.21	8.94	19.52
	9.56	18.45	8.74	18.61			14.37	19.23	10.94	19.7
	10.26	18.55	9.96	18.75					14.36	19.74
	10.9	18.55	13.68	19.15						
	11.46	18.76								
	AL	BLEU	AL	BLEU	AL	BLEU	AL	BLEU	AL	BLEU
De→En	0.47	21.08	1.57	19.2	0.49	21.65	1.44	24.97	1.63	26.07
	1.45	23.97	2.17	24.71	1.3	24.51	1.97	26.87	2.43	28.03
	2.12	26.21	2.77	28.26	2.17	27.12	2.55	28.22	3.49	29.88
	3.12	27.15	3.31	28.85	3.25	29.19	3.15	29.58	5.39	30.88
	4.1	28.53	4.01	29.55	4.31	29.97	3.58	29.97	6.24	31.15
	5.05	29.16	4.82	30.35	5.87	30.84	4.17	30.36	6.98	31.44
	6.03	29.72	5.66	30.52	7.65	31.29	5.41	30.86	8.36	31.58
	6.97	30.16	6.65	30.91	8.98	31.52	7.13	31.21	9.09	31.66
	7.9	30.69	7.7	31.05	10.53	31.6	9.09	31.33	10.16	31.67
	8.78	30.86	8.73	31.08	12.53	31.79	11.22	31.55	11.29	31.72
	9.7	31.11	9.79	31.2			12.53	31.74	12.5	31.73
	10.57	31.2	12.6	31.32						
	11.42	31.41								
	12.24	31.41								
	AL	BLEU	AL	BLEU	AL	BLEU	AL	BLEU	AL	BLEU
En→Vi	3.21	27.87	1.29	23.06	0.89	21.89	0.81	22.3	0.75	22.26
	3.93	29.4	1.85	26.33	1.41	27.11	1.31	25.93	1.2	25.81
	4.73	30.11	2.44	28.7	1.99	29.31	1.78	28.82	1.61	28.44
	5.57	30.14	3.23	29.37	3.06	29.63	2.24	29.43	2.06	29.55
	6.43	30.08	3.76	29.5	4.6	30.15	2.9	29.51	2.8	29.9
	7.28	30.13	4.42	29.48	5.44	30.09	3.92	29.9	3.88	30.12
	8.12	30.14	5.15	29.79	6.25	30.13	4.94	30.08	4.91	30.22
	8.93	30.11	5.91	29.83	7.49	30.15	6.11	30.09	6.08	30.17
	9.7	30.1	6.7	29.94	8.08	30.2				
	10.43	30.2	7.69	29.95	8.74	30.17				
	11.13	30.16	8.67	29.84	9.61	30.01				
	11.79	30.13	9.93	29.95	10.67	30.11				
	12.41	30.16	12.58	30.01	11.69	30.1				
	13.01	30.18								

Table 2: Numerical results in Figure 5.

C Algorithm

The inference process of DrFrattn policy is summarized in Algorithm 1.

Algorithm 1: SiMT inference with the DrFrattn Policy Net

Input: streaming source tokens: $\mathbf{X}_{\leq j}$,
threshold: λ,
target idx: $i \leftarrow 1$,
source idx: $j \leftarrow 1$,
max continuous READ constraint: r_{max},
current number of continuous READ: $r_c \leftarrow 1$

Output: target tokens: $\mathbf{Y} \leftarrow \{$<BOS>$\}$

while $\mathbf{Y}_{i-1} \neq$ <EOS> **do**
  calculate the predicted cumulative attention c with $\mathbf{Y}_{i-1}$ using the DrFrattn policy net mentioned in 4.2;
  if $c \leq \lambda$ **or** $r_c \geq r_{max}$ **then**
    translate y_i with $\mathbf{X}_{\leq j}$,$\mathbf{Y}_{\leq i-1}$;
    if $y_i \neq$ <EOS> **or** $j \geq |\mathbf{X}|$ **then**
      // execute WRITE action
      $\mathbf{Y}$.Append(y_i);
      $r_c \leftarrow 0$;
      $i \leftarrow i + 1$;
    else
      // execute READ action
      $j \leftarrow j + 1$;
      $r_c \leftarrow r_c + 1$;
  else
    // execute READ action
    $j \leftarrow j + 1$;
    $r_c \leftarrow r_c + 1$;
return $\mathbf{Y}$