SimCLNMT: A Simple Contrastive Learning Method for Enhancing Neural Machine Translation Quality

Menglong Xu **Yanliang Zhang***
School of Physics & Electronic Information Engineering, Henan Polytechnic University
Henan Jiaozuo 454000, China
xml@home.hpu.edu.cn ylzhang@hpu.edu.cn

Abstract

Neural Machine Translation (NMT) models are typically trained using Maximum Likelihood Estimation (MLE). However, this approach has a limitation: while it might select the best word for the immediate context, it does not generally optimize for the entire sentence. To mitigate this issue, we propose a simple yet effective training method called **SimCLNMT**. This method is designed to select words that fit well in the immediate context and also enhance the overall translation quality over time. During training, **SimCLNMT** scores multiple system-generated (candidate) translations using the logarithm of conditional probabilities. It then employs a ranking loss function to learn and adjust these probabilities to align with the corresponding quality scores. Our experimental results demonstrate that **SimCLNMT** consistently outperforms traditional MLE training on both the NIST English-Chinese and WMT'14 English-German datasets. Further analysis also indicates that the translations generated by our model are more closely aligned with the corresponding quality scores. We release our code at https://github.com/chaos130/fairseq_SimCLNMT.

1 Introduction

Neural machine translation (NMT) (Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014; Bahdanau et al., 2015) is a pivotal task within the field of natural language processing. It focuses on translating text from one language (source) to another (target) using end-to-end neural network models. This task has garnered increasing attention in both academia and industry (Shen et al., 2016; Vaswani et al., 2017; Kreutzer et al., 2018; Yang et al., 2018). Notably, the emergence of the Transformer architecture (Vaswani et al., 2017) marks a paradigm shift in NMT. Characterized by its innovative self-attention mechanism, the Transformer enables the model to variably prioritize the significance of words within the input sentence. This results in more effectively capturing the context compared to previous models, such as the Recurrent Neural Network (RNN) (Bahdanau et al., 2015) and Convolutional Neural Network (CNN) (Gehring et al., 2017). Despite the advancements of the Transformer, it still relies on the Maximum Likelihood Estimation (MLE) for word prediction at each time step. While this practice optimizes for immediate word selection probability, it may not consistently enhance the overall translation quality. This discrepancy between probabilistic optimality and actual translation quality highlights a critical limitation in current NMT approaches. To assess the practical impact of this limitation, we conducted a preliminary study using a pre-trained NMT model (MLE baseline). Our process involves generating two candidate translations for each input and analyzing whether higher probability correlates with higher BLEU scores (Papineni et al., 2002). As depicted in Table 1, the results indicate that the accuracy is far from ideal, suggesting that there is room for improvement in how NMT models select words for translation.

As suggested by Kreutzer et al. (2018), one primary strategy to mitigate the mentioned issues involves adopting Reinforcement Learning (RL). RL training allows models to be fine-tuned with reward mechanisms that assess the overall quality of predictions. This technique focuses on the result of the translation

*Corresponding author
©2024 China National Conference on Computational Linguistics
Published under Creative Commons Attribution 4.0 International License

process rather than just the immediate predictions of the next word. However, it introduces common challenges to deep RL, including noisy gradient estimation (Greensmith et al., 2004), which can result in unstable training and high sensitivity of hyperparameter settings. Moura Ramos et al. (2023) further explores this concept by incorporating evaluation metrics learned from human quality annotations as reward models within the NMT framework. This approach enhances translation quality through data filtering, RL training, and re-ranking techniques. As an alternative strategy, Minimum Risk Training (MRT) has been adopted for language generation tasks to overcome the MLE training challenge (Shen et al., 2016). Despite its potential, the accuracy of the MRT is limited by the number of output samples. Yang et al. (2018) introduces a novel method leveraging Generative Adversarial Network (GAN) for NMT, aiming to train the model to generate sentences indistinguishable from human translations. Additionally, Liu et al. (2022) and Yuan et al. (2023) have demonstrated that contrastive learning techniques can effectively improve the quality of model output.

Dataset(dev)	Bleu	Acc(%)
NIST en-ch	45.61	58.54
WMT14 en-de	25.93	54.10

Table 1: Accuracy of pre-trained NMT models w.r.t ranking the quality of candidate translations on NIST en-ch dataset (dev) and WMT14 en-de (dev). **Acc.** stands for the frequency of the model that assigns higher probabilities to better candidate translations. The candidate translations are generated by a pre-trained model. **Bleu** are the evaluation metric scores in the development data set. The pre-trained model only achieves 58.54% and 54.10% precision on both development data sets.

In this paper, we seek to address the mentioned issues by incorporating contrastive learning into the NMT framework. Inspired by Liu et al. (2022) and Yuan et al. (2023), we introduce **SimCLNMT**, a simple yet effective approach that aligns the probabilities of various candidate translations with their corresponding quality scores via a ranking loss mechanism. Our methodology is divided into two phases: pre-training and fine-tuning. During the pre-training phase, we train a NMT model using a comprehensive data set, which serves as a strong MLE baseline for the fine-tuning stage. In the fine-tuning phase, we generate several candidate translations from the MLE baseline model. Each candidate is then evaluated using an automated metric to determine its quality score, yielding pairs of candidate translations and quality scores for every source sentence. **SimCLNMT** computes the log probability for each candidate and uses a ranking loss mechanism to align these probabilities with their corresponding quality scores, ensuring that the model predictions are consistent with the evaluations provided by the metric. In summary, we mainly make the following contributions:

- To the best of our knowledge, this work is among the first efforts to leverage contrastive learning as a strategy to mitigate the limitations inherently caused by the MLE training in the context of Neural Machine Translation.
- Our model shows a significant performance enhancement when either BLEU or COMET is used as the designated evaluation metric for ranking candidate translations. This suggests that our approach can optimize any desired target metric and serve as an alternative to RL and MRT.
- We conduct extensive experiments on NIST English-Chinese and WMT'14 English-German datasets, and the results show that our approach can achieve consistent improvements.

2 Related Work

The training of NMT models typically relies on MLE. While MLE excels at selecting contextually appropriate words, it often falls short of optimizing overall sentence performance, which can result in suboptimal translation outputs. Several previous works have aimed to mitigate this issue. Kreutzer et al. (2018) explores the use of Reinforcement Learning (RL) to improve machine translation with human feedback. This approach fine-tunes NMT models using RL techniques and rewards that assess the overall

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 1047-1058, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

quality of translations, rather than focusing solely on predicting the next word. However, it also faces the inherent challenges of deep RL, including the problem of noisy gradient estimation. Moura Ramos et al. (2023) extends the exploration of using RL for NMT by integrating evaluation metrics derived from human feedback as reward models within the NMT framework. This methodology encompasses multiple strategies that include data filtering to refine the training corpus, RL training to adjust the NMT model based on rewards reflecting translation quality, and re-ranking techniques at inference time. This comprehensive approach seeks to enhance translation quality by ensuring that the outputs of the NMT model are more closely aligned with human judgments. Shen et al. (2016) presents a significant shift in NMT training methodologies by incorporating evaluation metrics as loss functions. This allows the model to learn patterns that better align with the quality of the final translation output, rather than just the likelihood of word sequences. Yang et al. (2018) introduces a novel method that utilizes GAN for NMT. In this approach, the NMT model acts as the generator, producing human-like translations, while a discriminator is trained to distinguish between machine and human translations. The goal is to iteratively improve the NMT model based on discriminator feedback, training it to generate translations indistinguishable from human ones and directly optimizing the output quality.

In addition to the above methodologies, contrastive learning has gained traction, particularly in abstractive summarization and large language models. Liu and Liu (2021) and Liu et al. (2022) introduce contrastive learning to the domain of abstractive summarization, focusing on learning from higher-quality sentences rather than those with high model probabilities. This approach shifts the focus towards generating summaries that are statistically likely and of higher relevance and quality. Yuan et al. (2023) further refines the integration of human feedback into language model fine-tuning through a novel training paradigm called RRHF. This method evaluates sampled responses from various sources using the logarithm of conditional probabilities and employs a ranking loss mechanism to align these probabilities with human preferences.

3 SimCLNMT

This section breaks down the holistic training process into two stages: pretraining (Section 3.1) and finetuning (Section 3.2).

3.1 Pretraining

In this phase, our goal is to pre-train a MLE based model π that results in good translations. MLE is the standard training method. It aims to maximize the likelihood of the reference translation y, i.e.,

$$\theta^* = \underset{\theta}{\operatorname{argmax}} \sum_i \log p_{\pi_\theta}\left(y^{*(i)} \mid x^{(i)}; \theta\right) \tag{1}$$

where θ denotes the parameters of π and p_{π_θ} denotes the probability distribution entailed by these parameters. The summation is over the training set and $\left\{x^{(i)}, y^{*(i)}\right\}$ is the i-th training sample.

For a specific sample $\left\{x^{(i)}, y^{*(i)}\right\}$, Eq. 1 is equal to minimizing the sum of negative log-likelihoods of the tokens $\left\{y_1^*, \ldots, y_j^*, \ldots, y_l^*\right\}$ in the reference translation y^* whose length is l, which is the cross-entropy loss:

$$L_{\text{xent}} = -\sum_{j=1}^{l} \sum_{y} p_{\text{true}}\left(y \mid x, y_{<j}^*\right) \log p_{\pi_\theta}\left(y \mid x, y_{<j}^*; \theta\right) \tag{2}$$

where $y_{<j}^*$ denotes the preceding reference sequence. In practice, p_{true} is a label-smoothing distribution under the standard MLE framework by assigning probability mass β to other tokens:

$$p_{\text{true}}\left(y \mid x, y_{<j}^*\right) = \begin{cases} 1-\beta & y = y_j^* \\ \frac{\beta}{N-1} & y \neq y_j^* \end{cases} \tag{3}$$

3.2 Finetuning

In this section, we first introduce the methods of candidate generation and evaluation. Subsequently, the section provides a detailed explanation of the ultimate training objectives of the proposed model.

CCL 2024

(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Candidate Generation NMT typically uses beam search (Freitag and Al-Onaizan, 2017) or diverse beam search (Vijayakumar et al., 2016) to generate candidate translations. Both are approximate algorithms to identify the maximum a-posteriori (MAP) output, i.e., the sentence with the largest estimated probability given an input (Edunov et al., 2018a). Beam search generally finds the high probability outputs (Ott et al., 2018).

However, MAP prediction can lead to less rich translations (Ott et al., 2018) since it aims to select the output sequence with the highest probability given the input. As a result, the model may generate accurate translations, but they may lack the nuances and unique expressions present in the source text.

As an alternative, we consider sampling-based approaches for generating candidate translations. This approach allows the model to explore less frequent, potentially more interesting, or contextually appropriate translations that deterministic methods (beam search) might overlook. First, we consider exploring pure sampling, which leads to higher variability in generation but may also result in less coherent translations due to the random selection of words with lower probabilities. Second, we employ Top-k sampling, which limits the sampling pool to the k most probable words at each time step. This reduces the chance of selecting highly improbable words, leading to more coherence while maintaining some level of diversity (Graves, 2013; Ott et al., 2018; Fan et al., 2018).

Candidate Evaluation Evaluation metrics are indispensable tools in the field of machine translation. They serve two critical functions: measuring the quality of translations produced by a model and guiding the iterative improvement of the NMT system through training. Previous work predominantly use BLEU (Papineni et al., 2002) to guide the training of the NMT system (Shen et al., 2016; Yang et al., 2018). However, BLEU relies on word overlap and n-gram matching, making them ineffective for translations that have the same meaning but are substantially different from the reference. To address these shortcomings, we use robust neural metrics to evaluate candidate translations, such as reference-based COMET (Rei et al., 2022).

Training Objective After the two mentioned stages, we engage n candidate-translation and quality-score pairs (y_i, r_i) for a source sentence x, in which r_i equals the evaluation metrics score $M(x, y_i)$. In this context, n serves as a hyperparameter. As the Figure 1 shows, to align with the quality scores $\{r_i\}_k$, our pre-trained NMT model π generates the model scores p_i for every y_i, according to the following formula:

$$p_i = \frac{\sum_t \log P_\pi\left(y_{i,t} \mid x, y_{i,<t}\right)}{||y_i||^\alpha} \tag{4}$$

Here, p_i denotes the conditional logarithmic probability normalized in the length of y_i in the model π, with α as the length normalization coefficient. Our approach is straightforward: we configure the model π to assign higher probabilities to the superior candidate translations and lower probabilities to the inferior ones. To refine model performance towards this goal, we employ a ranking loss:

$$L_{\text{rank}} = \sum_{i=1}^{N}\sum_{j=1}^{N} \theta(r_i < r_j) \cdot \max(0, p_i - p_j) \tag{5}$$

This equation introduces the indicator function $\theta(r_i < r_j)$, which equals 1 when the condition $r_i < r_j$, and 0 otherwise.

Following Edunov et al. (2018b), we combine the ranking loss (Eq. 5) with the cross-entropy (Eq. 2) losses to enhance the generative ability of the pre-trained NMT model. The total loss, as defined by (Eq. 6), is the weighted sum of these two losses.

$$L = L_{\text{rank}} + \lambda L_{xent} \tag{6}$$

4 Experiments and Results

We evaluate our **SimCLNMT** on the NIST English-Chinese and the WMT'14 English-German translation tasks. We experiment **SimCLNMT** on transformer-base model.

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 1047-1058, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

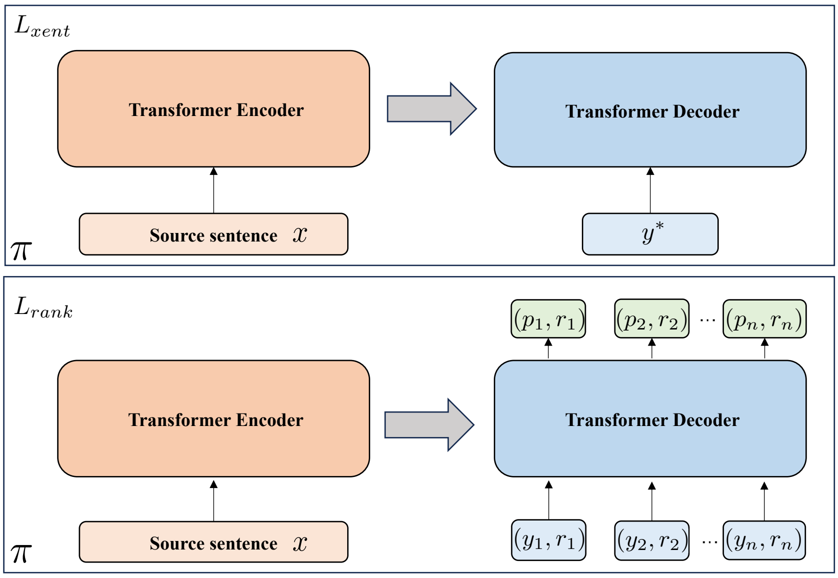

Figure 1: The π refers to the model trained in the pre-training phase. y^* denotes the reference translation of source sentence x

4.1 Data Sets and Preprocessing

English-Chinese For English-Chinese translation, our training data comprises 1.6 million sentence pairs. These are randomly extracted from LDC corpora[1]. Both the source and target sentences are encoded with byte-pair encoding (BPE) (Sennrich et al., 2016). The vocabulary for the source and target language consists of approximately 32000 tokens respectively[2]. We choose the NIST02 dataset as our development set, which includes 878 sentences. For testing, we utilize the NIST03, NIST04, NIST05, and NIST06 datasets, which contain 919, 1788, 1082, and 1664 sentences, respectively.

English-German In our English-German translation experiments, we use the WMT'14 English-German corpus. This publicly available dataset is widely recognized as a benchmark for NMT systems and contains 4.5 million sentence pairs[3]. Sentences are encoded with BPE, which employs a shared source-target vocabulary of about 37000 tokens. We report results on newstest2014. The newstest2013 is used as validation. These datasets contain 3003 and 3000 sentences, respectively.

4.2 Model Parameters and Evaluation

For the Transformer model, following the base model in Vaswani et al. (2017), we set the dimension of the word embedding as 512, the dropout rate as 0.1, and the number of attention heads to 8. Both the encoder and decoder have a stack of 6 layers. During testing, We use beam search with a beam size of 4 and a length penalty of 0.6.

We reimplement the Transformer model in Pytorch using the fairseq (Ott et al., 2019) toolkit[4] and train it synchronously across four GTX 3090 GPUs in a multi-GPU setup on a single machine. To assess the performance of each NMT system, we utilize well-established evaluation metrics, including case-sensitive detokenized BLEU with SacreBLEU [5] (Post, 2018) and ChrF (Popović, 2015).

[1] LDC2002L27, LDC2002T01, LDC2002E18, LDC2003E07, LDC2004T08, LDC2004E12, LDC2005T10

[2] When doing BPE for Chinese, we need to do word segmentation first using jieba and the following steps are the same with BPE for English.

[3] http://nlp.stanford.edu/projects/nmt

[4] Code available at https://github.com/pytorch/fairseq

[5] https://github.com/mjpost/sacrebleu

(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Model	Metric	English-Chinese					Enlish-German
		NIST03	NIST04	NIST05	NIST06	average	newstest2014
MLE	BLEU	39.05	37.53	36.18	37.31	37.52	26.21
	ChrF	33.81	32.39	31.50	32.27	32.49	57.19
SimCLNMT (diverse beam search)	BLEU	39.79	38.56	36.86	38.57	**38.45**	**26.84**
	ChrF	34.10	32.96	31.75	33.03	**32.96**	**57.85**
SimCLNMT (pure sampling)	BLEU	39.66	38.11	36.97	38.59	38.33	26.64
	ChrF	33.95	32.67	31.77	32.96	32.84	57.60
SimCLNMT (top-10 Sampling)	BLEU	39.81	38.16	37.04	38.39	38.35	26.74
	ChrF	34.08	32.63	31.87	32.83	32.85	57.61

Table 2: BLEU and ChrF score on English-Chinese and English-German translation tasks. **diverse beam search, pure sampling, top-10 sampling** represents different methods for generating candidate translations in Section 3.2.

4.3 Training Details

During the fine-tuning phase, We randomly select 1,000 examples from the NIST training set and 2000 examples from the WMT14 training set. To ensure the reliability and broad applicability of our findings, we use three different random seeds for the sample selection process and report the average results from the test set. To generate candidate translations for each source sentence, we employ various methods, including diverse beam search, pure sampling, and Top-10 sampling. Our experiments indicate that the hyper-parameter n had a negligible impact on the results. Consequently, based on the performance of the development set, we set the value of n to 32. We then fine-tune the pre-trained NMT model using the Adam optimizer, adjusting the learning rate as necessary. We train the model for fifteen epochs, selecting the best-performing model on the development set for further evaluation. For more details, please refer to Appendix A.

4.4 Main Results

Table 2 presents the main results of English-Chinese and English-German test sets. Employing diverse beam search, our model (i.e., the line of SimCLNMT (diverse beam search) in Table 2) achieves an average improvement of +0.93 points in BLEU and +0.47 points in ChrF for the English-Chinese test set. For the English-German test set, the gains are +0.63 points in BLEU and +0.66 points in ChrF. Using pure sampling, the model (the line of SimCLNMT (pure sampling)) shows improvements up to +0.81 points in BLEU and +0.35 points in ChrF on the English-Chinese test set. For the English-German test set, the gains are +0.43 points in BLEU and +0.41 points in ChrF. With top-10 sampling, the model (the line of SimCLNMT (top-10 sampling)) achieves an average improvement of +0.83 points in BLEU and +0.36 points in ChrF on the English-Chinese test set. For the English-German test set, the gains are +0.53 points in BLEU and +0.42 points in ChrF.

These findings demonstrate that our proposed model consistently outperforms MLE baseline, with optimal results observed when using diverse beam search. Notably, when employing sampling-based methods to generate candidate translations, the performance of our model does not decline significantly compared to diverse beam search. This robustness across different data distributions underscores the utility and effectiveness of our model.

5 Analysis

5.1 Ablation Study

In our investigation, we perform an ablation study to assess the effectiveness of the **SimCLNMT** approach. We train two distinct models under identical conditions, with the only difference being the loss function used. The first model is trained using the Maximum Likelihood Estimation (MLE) method, with the loss function as defined in Equation 2. In contrast, the second model integrates the **SimCLNMT**

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 1047-1058, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

technique, using the loss function introduced in Equation 5. As shown in Table 3, the **SimCLNMT(5)** model exhibits a significant improvement over the **SimCLNMT(2)** model. This comparison underscores the effectiveness of our proposed method. In the subsequent subsection, we uniformly employ diverse beam search to generate candidate translations across all our experiments.

Model	Metric	English-Chinese					English-German
		NIST03	NIST04	NIST05	NIST06	average	Newstest2014
SimCLNMT(2)	BLEU	39.00	37.57	36.08	37.37	37.51	26.20
	ChrF	33.76	32.40	31.40	32.28	32.46	57.26
SimCLNMT(5)	BLEU	39.87	38.51	36.80	38.71	**38.47**	**26.84**
	ChrF	34.15	32.90	31.69	33.14	**32.97**	**57.80**

Table 3: SimCLNMT(2) is trained using the loss function presented in Equation 2, whereas **SimCLNMT(5)** utilizes the loss function detailed in Equation 5. These results focus on the candidate translations obtained through diverse beam search.

Model	BLEU	ChrF	COMET
MLE	37.52	32.46	84.99
SimCLNMT(B)	**38.45**	**32.96**	85.18
SimCLNMT(C)	38.46	32.95	**85.50**

a

Table 4: Results on NIST English-Chinese using different evaluation metrics as M in Section 3.2. **SimCLNMT (B)** is trained with candidate translations scored by BLEU, while SimCLNMT (C) is trained with candidate translations scored by COMET. The reported results represent the average scores across the test sets

5.2 Training with Different Evaluation Metrics

In our preceding experiments, BLEU is the primary metric for assessing the quality of candidate translations during the finetuning phase. To broaden the evaluation of our methodology beyond BLEU, we introduce a reference-based neural metric, COMET[6] (Rei et al., 2022), as the evaluation metric M in Section 3.2. Subsequently, we train another version of **SimCLNMT**, leveraging the scores of candidate translations as calculated by COMET.

The results in Table 4 indicate that (1) Our model significantly enhances performance when BLEU or COMET is utilized as the target evaluation metric for ranking candidate translations. This suggests that our method can be optimized for any specific target metric, offering an alternative to reinforcement learning or minimum-risk training. (2) The model trained on one evaluation metric (e.g., COMET) also shows improvement on another metric (e.g., BLEU) compared with the MLE baseline model. This indicates that the enhancements from our model are not solely due to the potential weaknesses of individual metrics. Furthermore, it highlights a significant correlation between BLEU and COMET, reinforcing the generalization abilities of the model across different evaluation metrics.

5.3 Ranking Correlation

We evaluate the correlation between the model-calculated probabilities of candidate translations and their quality scores. We use Eq. 2 to estimate probabilities, using BLEU as the metric for the quality scores of the candidate translations. The Spearman rank correlation is calculated for each instance, with these scores then averaged to obtain the overall correlation. Using a total of 32 candidates for calculation, Table 5 shows that **SimCLNMT** exhibits a better rank correlation among candidate translations compared

[6] https://github.com/Unbabel/COMET

CCL 2024

to the MLE baseline model. This signifies that our model more accurately assigns higher probabilities to better candidate translations and lower probabilities to the less favorable ones.

Model	English-Chinese				
	NIST03	NIST04	NIST05	NIST06	average
MLE	0.0812	0.0911	0.0583	0.0855	0.0790
SimCLNMT	**0.1567**	**0.1449**	**0.1201**	**0.1240**	**0.1364**

Table 5: Rank Correlation between the model's estimated probabilities of the candidate translations and the quality scores (BLEU) of the candidate translations on NIST (English-Chinese). We calculate the Spearman rank correlation between Transformer and **SimCLNMT**.

5.4 Statistical Significance Testing

In neural machine translation models, significance tests are crucial for evaluating the statistical significance of performance differences between model variations or different models. These tests help researchers determine whether observed performance improvements are due to random fluctuations or genuine enhancements in the model. We conduct statistical significance tests using the t-test to compare **SimCLNMT** with the MLE baseline model. The experimental results in Table 6 show that the p-value of the t-test is significantly less than 0.05, indicating a statistically significant difference. These findings further confirm that the performance of **SimCLNMT** is superior to that of the MLE baseline model.

Dataset	NIST-ench	WMT14-ende
t-test	0.0002842	0.001517

Table 6: The results of statistical significance testing

6 Limitation and Future Work

Table 7 offers a comprehensive statistical comparison between the MLE baseline and **SimCLNMT** models on the English-Chinese and English-German test sets. Specifically, for the NIST03 English-Chinese test set, both models generated 919 candidate translations, with their BLEU scores subsequently measured for evaluation. Upon analysis, in 33 cases, which represent 3.59% of the total instances, the BLEU scores for the **SimCLNMT** model were found to be lower than those of the MLE baseline. This percentage, termed the "decrease rate," indicates a decline in performance for the **SimCLNMT** model in certain sentences.

We hypothesize that this performance decline is attributable to the phenomenon of "forgetting," where the model loses some of the original knowledge post full-parameter fine-tuning. This insight has prompted us to consider strategies for future work aimed at enhancing translation quality and mitigating this forgetting effect. To address the issue, we plan to explore methods that include selectively freezing certain parameters within the MLE baseline model or incorporating adapter modules (Houlsby et al., 2019). These strategies are designed to preserve the model's previously acquired knowledge and prevent its loss, thereby potentially improving the stability and quality of translations produced by the **SimCLNMT** model.

Dataset	English-Chinese				English-German
	NIST03	NIST04	NIST05	NIST06	newstest2014
decrease rate	3.59%	3.28%	3.30%	3.09%	3.21%

Table 7: The results of decrease rate.

CCL 2024

7 Conclusion

In this study, we introduce a straightforward yet effective training method termed **SimCLNMT**. The objective of this approach is to select words that not only fit well in the immediate context but also enhance the overall translation quality over time. Through extensive experimentation on translation tasks between English-Chinese and English-German, we demonstrate that our **SimCLNMT** method consistently achieves substantial improvements. These results underscore the effectiveness of our approach in the domain of machine translation.

CCL 2024

References

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*.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018a. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500.

Sergey Edunov, Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018b. Classical structured prediction losses for sequence to sequence learning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 355–364.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898.

Markus Freitag and Yaser Al-Onaizan. 2017. Beam search strategies for neural machine translation. *arXiv preprint arXiv:1702.01806*.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *International conference on machine learning*, pages 1243–1252. PMLR.

Alex Graves. 2013. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*.

Evan Greensmith, Peter L Bartlett, and Jonathan Baxter. 2004. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research*, 5(9).

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1700–1709.

Julia Kreutzer, Joshua Uyheng, and Stefan Riezler. 2018. Reliability and learnability of human bandit feedback for sequence-to-sequence reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1777–1788.

Yixin Liu and Pengfei Liu. 2021. Simcls: A simple framework for contrastive learning of abstractive summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1065–1072.

Yixin Liu, Pengfei Liu, Dragomir Radev, and Graham Neubig. 2022. Brio: Bringing order to abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2890–2903.

Miguel Moura Ramos, Patrick Fernandes, António Farinhas, and André FT Martins. 2023. Aligning neural machine translation models: Human feedback in training and inference. *arXiv e-prints*, pages arXiv–2311.

Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018. Analyzing uncertainty in neural machine translation. In *International Conference on Machine Learning*, pages 3956–3965. PMLR.

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.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Maja Popović. 2015. chrf: character n-gram f-score for automatic mt evaluation. In *Proceedings of the tenth workshop on statistical machine translation*, pages 392–395.

Matt Post. 2018. A call for clarity in reporting bleu scores. *WMT 2018*, page 186.

Ricardo Rei, José GC De Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André FT Martins. 2022. Comet-22: Unbabel-ist 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585.

CCL 2024

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 1047-1058, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers
(c) Technical Committee on Computational Linguistics, Chinese Information Processing Society of China

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725.

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

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.

Ashwin K Vijayakumar, Michael Cogswell, Ramprasath R Selvaraju, Qing Sun, Stefan Lee, David Crandall, and Dhruv Batra. 2016. Diverse beam search: Decoding diverse solutions from neural sequence models (2016). *arXiv preprint arXiv:1610.02424*.

Zhen Yang, Wei Chen, Feng Wang, and Bo Xu. 2018. Improving neural machine translation with conditional sequence generative adversarial nets. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1346–1355.

Zheng Yuan, Hongyi Yuan, Chuanqi Tan, Wei Wang, Songfang Huang, and Fei Huang. 2023. Rrhf: Rank responses to align language models with human feedback without tears. *arXiv preprint arXiv:2304.05302*.

CCL 2024

Appendix

A Training Details

During the pretraining phase, for the English-Chinese translation task, we adjust the learning rate to 5×10^{-4}, setting the warm-up steps at 4,000. We train the transformer-base model, as defined by Vaswani et al. (2017), for a total of 60,000 steps. We select the model checkpoint that achieves the best performance on the development dataset for further analysis. For the English-German translation task, we follow the configurations specified by Vaswani et al. (2017).

In the fine-tuning phase, we use FP16 to accelerate our training. The average training time for NIST and WMT14 is about 10 minutes. We use Adam optimizer with the learning rate set to 2×10^{-4}. The original development set is used for hyper-parameter optimization. Table 8 outlines the specific hyper-parameter configurations used during the fine-tuning phase.

Datasets	α (Eq. 4)	λ (Eq. 5)
NIST English-Chinese	0	0.1
WMT14 English-German	1.5	1

Table 8: Hyper-parameter Setting

CCL 2024