SCIR-MT's Submission for WMT24 General Machine Translation Task

Baohang Li, Zekai Ye, Yichong Huang, Xiaocheng Feng, Bing Qin
Harbin Institute of Technology
{baohangli,zkye,ychuang,xcfeng,qinb}@ir.hit.edu.cn

Abstract

This paper introduces the submission of SCIR research center of Harbin Institute of Technology participating in the WMT24 machine translation evaluation task of constrained track for English to Czech. Our approach involved a rigorous process of cleaning and deduplicating both monolingual and bilingual data, followed by a three-stage model training recipe. During the testing phase, we used the beam serach decoding method to generate a large number of candidate translations. Furthermore, we employed COMET-MBR decoding to identify optimal translations.

1 Introduction

This paper presents the submission from the SCIR-MT in the WMT24 machine translation evaluation task, focusing on the **constrained** track of English to Czech translation. In the field of machine translation, the quality of translation systems has been improved with the development of large language models and the increase in data volume. However, achieving high-quality translation outputs under limited conditions remains a challenging task due to resource and computational constraints (Freitag and Al-Onaizan, 2017).

Our team has adopted a series of innovative methods to address this challenge. Initially, we conducted a rigorous cleaning and deduplication process for both monolingual and bilingual data to ensure the quality of the training dataset. Subsequently, we implemented a three-stage model training strategy, including monolingual continual pre-training, bilingual continual pre-training, and translation-specific supervised intruction tuning. During the testing phase, we utilized the beam search decoding method to generate a multitude of candidate translations and applied the COMET-MBR (Fernandes et al., 2022) decoding strategy to identify the optimal translations.

The structure of this paper is as follows: we first provide a detailed description of the data preprocessing steps and strategies; then, we outline our foundational model selection and training strategy; next, we introduce the decoding algorithms used in the testing phase; and finally, we present the COMET-MBR decoding method and report our experimental results on the wmttest2023 dataset. These methods have led to excellent performance in terms of both BLEU (Post, 2018) and COMET (Rei et al., 2020) scores, demonstrating the effectiveness of our approach.

2 Data Preprocessing

2.1 Provided Data

Bilingual Corpus We used all the provided bitext corpora: Europarl v10, ParaCrawl v9 (Bañón et al., 2020), Common Crawl, News Commentary v18.1, Wiki Titles v3, WikiMatrix, Tilde MODEL corpus, and TED Talks (Cettolo et al., 2012).

Monolingual Corpus We also used the following provided monolingual data: News Crawl, Europarl v10, News Commentary, Common Crawl, and Leipzig Corpora (Biemann et al., 2007).

2.2 Data Cleaning

Data cleaning played a pivotal role in improving the quality of our training dataset. During this stage, we implemented several key steps to ensure the quality of the bilingual data and monolingual data, respctively.

2.2.1 Bilingual Corpus

Given that a significant portion of the training dataset is synthetically-aligned, we need to use a comprehensive data preprocessing pipeline to ensure good translation quality. In particular, we sequentially performed heuristic-based, statistics-based, and embedding-based methods to filter our data.

Proceedings of the Ninth Conference on Machine Translation (WMT), pages 280–285
November 15–16, 2024.

Heuristic-based The following heuristic-based filters are used before applying the others:

- **Language Detection** We use fasttext[1] (Joulin et al., 2016) to filter out sentence pairs mismatching the English-Czech direction.
- **Numerical Matching** If one sentence in a pair has a number (ordinal, date, etc.), we also checked the other sentence if a matching number is present. If a match is not detected, the pair is removed.

Statistics-based We employed statistics-based filters on sentence pairs following (Cruz, 2023). We first tokenized then applied the following statistics-based filters:

- **Length Filter** We removed pairs containing sentences with more than 50 characters.
- **Pair Length Ratio** We removed pairs where the ratio of the string lengths between the source and target sentences is greater than 1.2.
- **Symbol Token Ratio** We removed any sentence pairs in which either the source or target sentence appears more than 5 times.
- **Messy Token Ratio** We removed pairs where the number of messy characters in the sentences exceeds 2.
- **Most Frequent Words Gap** We measured the symmetry of bilingual text pairs by calculating the difference in the occurrence counts of the most frequent words in each text, and removed pairs where this difference exceeded 5.

Embedding-based Finally, we experimented with the use of sentence embedding models to compute the cross-lingual embedding similarity between the sentence pair. We used LaBSE (Feng et al., 2020) models to embed both the source and target sentences then computed a cosine similarity score between the two. The pair must have a similarity score $0.95 < s \leq 1$ to be kept.

After rigorous data cleaning, we filtered the bilingual training data from 56,288,239 pairs to 2,725,848 pairs, retaining only 4.8% of the highest quality data for continual pre-training.

[1]https://fasttext.cc

2.2.2 Monolingual Corpus

For incremental pre-training of large language models(Wu et al., 2024), we employed the Data-Juicer[2] (Chen et al., 2024b) to filter monolingual data in English and Czech. The filtering part includes the following filters: 1) Number of words, 2) Character repetition ratio, 3) Word repetition ratio, 4) Special character ratio, 5) Stop word ratio, 6) Flagged word ratio, 7) Language identification confidence, 8) Perplexity score, 9) Document length (number of characters), 10) Number of lines, 11) Short line length ratio, 12) Short line ratio.

To address the challenge of assessing the quality of the Czech data, we assumed that the Czech data provided by the competition organizers was of generally acceptable quality, reflecting a reasonable approximation of Czech syntax and expression. To further enhance data quality and improve model performance, we applied the Interquartile Range (IQR) (Whaley III, 2005) statistical method to establish a threshold for data filtering. The IQR method is particularly advantageous because it allows for the objective identification of outliers samples without making specific assumptions about the data distribution.

We calculated the IQR for the Czech dataset to define a reasonable range for data quality. Any samples falling outside this range were deemed potential outliers and excluded from the training data. By evaluating each data pair against these quality filtering criteria, we ensured that only samples within the acceptable range contributed to the training process. This approach enabled us to retain the most representative, high-quality samples, thereby enhancing the overall performance of the translation model. Table 1 presents the number of rows in each dataset with/without filtering.

Corpus	w/o. Filtering	w. Filtering
Common Crawl corpus	333 M	37 M
News Crawl	12M	4.6M
Leipzig Corpora	4 M	1.9 M
Europarl v10	669 K	391K
News Commentary v18.1	283 K	138 K

Table 1: Czech Corpus Statistics. Line counts are listed before and after filtering.

[2]https://github.com/modelscope/data-juicer

3 Translation Model Training

This Section describes our foundation model selection and model training strategy.

3.1 Model Configuration

We adopted LLaMA-2-13B as our foundation model considering its impressive performance on most English benchmarks after pre-training on 1.4T tokens (Touvron et al., 2023). Specifically, our Translation Model was initialized from the LLaMA-2-13B model to reduce the computational cost and continues to train on massive Czech and parallel corpus.

3.2 Training Strategy

In pre-trained models such as LLaMA-2, which are primarily trained on English data, integrating monolingual data during continual pre-trainingalongside parallel data has been shown to substantially enhance performance (Guo et al., 2024; Alves et al., 2024). Leveraging this approach, we improved our translation model by first incorporating monolingual data during the continual pre-training phase of models initially trained in English. This was followed by further continual pre-training using parallel data. Finally, we conducted instruction fine-tuning with a limited amount of bilingual data. Our models were developed using the LLaMA-Factory framework (Zheng et al., 2024), which facilitated this comprehensive training process.

3.2.1 Stage-1: Monolingual Continual Pre-training

In the initial phase of our training approach, we conducted secondary pre-training on the large language model (LLM) utilizing the carefully-curated monolingual dataset (shown in 2.2.2). The core objective of this stage is to enrich the LLM's understanding and generation capabilities in non-English languages.

We aimed to strengthen the LLM's multilingual capabilities by exposing it to a diverse monolingual corpus. Although this step was related to machine translation, it was designed to lay a solid foundation for the model's language proficiency, which was critical for the subsequent stages focusing on translation tasks.

Hyperparameters We used the AdamW optimizer, with $\beta_1 = 0.9$, $\beta_2 = 0.95$, and $\epsilon = 1.0 \times 10^{-8}$. The context length is 2048, and training is conducted for 1 epoch. We performed validation every 100 training steps. We used a cosine learning rate schedule with a warmup ratio of 1% and a peak learning rate of 2×10^{-5}. We applied a weight decay of 0.1 and gradient clipping of 1.0. We utilized eight NVIDIA RTX A800 GPUs, processing 1 batch on each GPU with a gradient accumulation step of 32, achieving an effective batch size of 256. During training, Flash-Attenion(Dao et al., 2022), bfloat16 precision, gradient checkpointing, and DeepSpeed ZeRO Stage 2(Rasley et al., 2020) were employed. With these configurations, the training process was completed in 5 days, which accelerates the overall training duration.

3.2.2 Stage-2: Bilingual Continual Pre-training

Bilingual Continual Pre-training is a methodology that involves ongoing training on bilingual datasets to improve the model's alignment between languages. This approach facilitates the model's ability to capture detailed syntactic and semantic correspondences across languages. Such fine-grained alignment is helpful for machine translation, as it enhances the accuracy of encoding source language information and improves the quality of the generated translations, thereby producing more precise and fluent translation outcomes.

Hyperparameters We performed continual pre-training on the model that achieves the minimum validation. We adopted the AdamW optimizers parameter used in Section 3.2.1. Weight decay and gradient clipping remained the same as in Section 3.2.1. We used a cosine learning rate schedule without warmup and a peak learning rate of 1×10^{-5}. We conduct validation every 10% of the total training steps for Continual Pre-training with Sentence-aligned Parallel Data, with 1 epoch and a batch size of 256.

3.2.3 Stage-3: Translation-specific Supervised Fine-Tuning

During the instruction fine-tuning stage, we constructed bilingual translation data in a question and answer format, where the instruction language was the source language for translation. We also employed full-scale parameter training. As highlighted in previous research (Xu et al., 2023), instruction fine-tuning of large language models

(LLMs) benefits from limited yet high-quality datasets. To ensure the optimal quality of data during fine-tuning, we followed previous research practices and used translation fine-tuning datasets constructed from the WMT validation data. These datasets, which underwent rigorous quality control measures, were ideal for fine-tuning purposes.

Hyperparameters We adjusted the AdamW optimizers parameters as used in Section 3.2.1. Weight decay and gradient clipping remained the same as in Section 3.2.1. The peak learning rate was set to 9.0×10^{-6} for full fine-tuning, without warmup, using an inverse square schedule. We conducted validation every 10% of the total training steps for SFT, with 3 epochs and a batch size of 64.

4 Decoding algorithms

In the test stage, we first generated multiple candidate translations for the given source sentence. Then, we performed MBR to determine the final translation.

4.1 Candidate Generation

During the testing phase, we produced 42 high-quality candidate translations. To enhance the diversity of these results, we employed In-Context Learning (ICL) techniques alongside the beam search algorithm. Specifically, we began by sampling various translation examples to serve as demonstrations, which contributed to greater result diversity. We then applied beam search with a beam width of 5 to generate the final set of 42 top hypotheses. This approach effectively integrates context learning and diversity sampling, thereby optimizing both the coverage and quality of the translations.

4.2 COMET-MBR

COMET-MBR (Fernandes et al., 2022) employs Minimum Bayes Risk (MBR) decoding (Kumar and Byrne, 2004; Eikema and Aziz, 2020) with a COMET model (Rei et al., 2020) that has been trained on direct assessments. Typically, a translation $\hat{y}^{MAP} \in \mathcal{V}_Y^{|y^*|}$ is generated using Maximum-A-Posteriori (MAP) decoding, defined as:

$$\hat{y}^{MAP} = \underset{y \in \mathcal{Y}}{argmax} \log p(y|x), \tag{1}$$

where $\mathcal{Y} \subseteq \bigcup_{i=1}^{\infty} \mathcal{V}_Y^i$ represents the search space of target sentences. Unlike MAP decoding, MBR decoding aims to identify the translation that minimizes the Bayes risk:

$$\hat{y}^{MBR} = \underset{h \in \mathcal{Y}}{argmax} \underbrace{E_{y' \sim p(y|x)}[u(y', h)]}_{\approx \frac{1}{m} \sum_{j=1}^{m} u(y^{(j)}, h)}, \tag{2}$$

where $\bar{\mathcal{Y}} \subseteq \mathcal{Y}$ denotes a set of translation hypotheses, and $u : \mathcal{Y} \times \mathcal{Y} \rightarrow R$ is the utility function. In our study, we utilize COMET[3] (Rei et al., 2020) as the utility function u. It is important to note that the hypotheses set $\bar{\mathcal{Y}}$ and the sample set used for expectation estimation, $\{y^{(1)}, \ldots, y^{(m)}\}$, are shared, except for h, i.e., $\{y^{(1)}, \ldots, y^{(m)}\} = \bar{\mathcal{Y}} \setminus \{h\}$. Consequently, given a candidate set, the computational complexity of MBR decoding is on the order of $\mathcal{O}(m^2)$, which leads to slower inference speeds as m increases.

5 Experimental Results

We evaluated the translation performance of our system on the WMTTest2023 dataset (Tom et al., 2023) and the Flores-200 benchmark (Costa-jussà et al., 2022). To assess translation quality, we employed both BLEU and COMET scores, utilizing the COMET model Unbabel/wmt22-comet-da[3]. Table 2 provides a comparative analysis with two existing commercial translation systems, Baidu[4] and Google[5]. In this table, "Stage1" "Stage2" and "Stage3" refer to the respective stages of our model training process. The performance labeled as "COMET-MBR" corresponds to the results of applying our MBR decoding approach to the candidate translations.

6 Conclusion

In this paper, we describe the materials we submitted for the general translation task at WMT2024. We participated in a constrained track: En→Cs. We trained a machine translation model based on LLaMA, utilizing a comprehensive data pipeline for filtering and curation. This pipeline integrates embedding-based, heuristic-based, and statistics-based filters. Subsequently, we employed a three-stage training method to enhance the translation capabilities of the model. Additionally, we utilized minimum Bayes risk decoding to refine the translation candidates. On two benchmark

[3]https://huggingface.co/Unbabel/wmt22-comet-da
[4]https://fanyi.baidu.com
[5]https://translate.google.com

Methods	Flores		WMT23	
	BLEU	COMET	BLEU	COMET
	Existing Systems			
Baidu	31.43	89.26	35.34	86.04
Google	36.81	91.51	50.25	89.90
	Ours(Based on LLaMA2-13B)			
Baseline	23.74	86.44	22.08	79.71
+Stage1	25.75	88.84	26.72	84.18
+Stage2	31.60	89.83	33.29	85.09
+Stage3	32.95	89.51	35.60	87.76
+COMET-MBR	33.44	92.27	36.61	89.09

Table 2: Comparison of translation performance using BLEU and COMET scores. We use LLaMA-2-13B as our base model.

datasets, our system outperformed Baidu and exhibited performance comparable to Google, both of which are unconstrained business systems with significantly more training data.

Future Directions. In the future, we aim to investigate how to prevent the catastrophic forgetting problem in the general capabilities of LLMs caused by continual pre-training on non-English data, which will help models benefit from effective translation-specific prompting techniques (Huang et al., 2024a; He et al., 2024; Chen et al., 2024a). Additionally, it is promising to train multiple translation systems based on different pre-training language models and combine their outputs with the ensemble learning strategies (Huang et al., 2024b; Jiang et al., 2023).

References

Duarte M Alves, José Pombal, Nuno M Guerreiro, Pedro H Martins, João Alves, Amin Farajian, Ben Peters, Ricardo Rei, Patrick Fernandes, Sweta Agrawal, et al. Tower: An open multilingual large language model for translation-related tasks. *arXiv preprint arXiv:2402.17733*, 2024.

Marta Bañón, Pinzhen Chen, Barry Haddow, Kenneth Heafield, Hieu Hoang, Miquel Esplà-Gomis, Mikel Forcada, Amir Kamran, Faheem Kirefu, Philipp Koehn, et al. Paracrawl: Web-scale acquisition of parallel corpora. Association for Computational Linguistics (ACL), 2020.

Chris Biemann, Gerhard Heyer, Uwe Quasthoff, and Matthias Richter. The leipzig corpora collection-monolingual corpora of standard size. *Proceedings of Corpus Linguistic*, 2007, 2007.

Mauro Cettolo, Christian Girardi, and Marcello Federico. Wit3: Web inventory of transcribed and translated talks. In *Proceedings of the Conference of European Association for Machine Translation (EAMT)*, pages 261–268, 2012.

Andong Chen, Lianzhang Lou, Kehai Chen, Xuefeng Bai, Yang Xiang, Muyun Yang, Tiejun Zhao, and Min Zhang. DUAL-REFLECT: Enhancing large language models for reflective translation through dual learning feedback mechanisms. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 693–704, Bangkok, Thailand, August 2024a. Association for Computational Linguistics. URL `https://aclanthology.org/2024.acl-short.64`.

Daoyuan Chen, Yilun Huang, Zhijian Ma, Hesen Chen, Xuchen Pan, Ce Ge, Dawei Gao, Yuexiang Xie, Zhaoyang Liu, Jinyang Gao, et al. Data-juicer: A one-stop data processing system for large language models. In *Companion of the 2024 International Conference on Management of Data*, pages 120–134, 2024b.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*, 2022.

Jan Christian Blaise Cruz. Samsung r&d institute philippines at wmt 2023. *arXiv preprint arXiv:2310.16322*, 2023.

Tri Dao, Dan Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. Flashattention: Fast and memory-efficient exact attention with io-awareness. *Advances in Neural Information Processing Systems*, 35:16344–16359, 2022.

Bryan Eikema and Wilker Aziz. Is map decoding all you need? the inadequacy of the mode in neural machine translation. *arXiv preprint arXiv:2005.10283*, 2020.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*, 2020.

Patrick Fernandes, António Farinhas, Ricardo Rei, José GC de Souza, Perez Ogayo, Graham Neubig, and André FT Martins. Quality-aware decoding for neural machine translation. *arXiv preprint arXiv:2205.00978*, 2022.

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

Jiaxin Guo, Hao Yang, Zongyao Li, Daimeng Wei, Hengchao Shang, and Xiaoyu Chen. A novel paradigm boosting translation capabilities of large language models. *arXiv preprint arXiv:2403.11430*, 2024.

Zhiwei He, Tian Liang, Wenxiang Jiao, Zhuosheng Zhang, Yujiu Yang, Rui Wang, Zhaopeng Tu, Shuming Shi, and Xing Wang. Exploring human-like translation strategy with large language models. *Transactions of the Association for Computational Linguistics*, 12:229–246, 2024. doi: 10.1162/tacl_a_00642. URL https://aclanthology.org/2024.tacl-1.13.

Yichong Huang, Xiaocheng Feng, Baohang Li, Chengpeng Fu, Wenshuai Huo, Ting Liu, and Bing Qin. Aligning translation-specific understanding to general understanding in large language models, 2024a. URL https://arxiv.org/abs/2401.05072.

Yichong Huang, Xiaocheng Feng, Baohang Li, Yang Xiang, Hui Wang, Bing Qin, and Ting Liu. Ensemble learning for heterogeneous large language models with deep parallel collaboration, 2024b. URL https://arxiv.org/abs/2404.12715.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-blender: Ensembling large language models with pairwise ranking and generative fusion. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14165–14178, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.792. URL https://aclanthology.org/2023.acl-long.792.

Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hérve Jégou, and Tomas Mikolov. Fasttext.zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.

Shankar Kumar and Bill Byrne. Minimum bayes-risk decoding for statistical machine translation. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 169–176, 2004.

Matt Post. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*, 2018.

Jeff Rasley, Samyam Rajbhandari, Olatunji Ruwase, and Yuxiong He. Deepspeed: System optimizations enable training deep learning models with over 100 billion parameters. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 3505–3506, 2020.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*, 2020.

Kocmi Tom, Eleftherios Avramidis, Rachel Bawden, Ondřej Bojar, Anton Dvorkovich, Christian Federmann, Mark Fishel, Markus Freitag, Thamme Gowda, Roman Grundkiewicz, et al. Findings of the 2023 conference on machine translation (wmt23): Llms are here but not quite there yet. In *WMT23-Eighth Conference on Machine Translation*, pages 198–216, 2023.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Dewey Lonzo Whaley III. The interquartile range: Theory and estimation. Master's thesis, East Tennessee State University, 2005.

Tongtong Wu, Linhao Luo, Yuan-Fang Li, Shirui Pan, Thuy-Trang Vu, and Gholamreza Haffari. Continual learning for large language models: A survey. *arXiv preprint arXiv:2402.01364*, 2024.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*, 2023.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*, 2024.