Dynamic Jointly Batch Selection for Data Efficient Machine Translation Fine-Tuning

Mohammad Amin Ghanizadeh and Mohammad Javad Dousti
Department of Electrical and Computer Engineering,
College of Engineering, University of Tehran, Tehran, Iran
{ghanizadeh.amin,mjdousti}@ut.ac.ir

Abstract

Data quality and its effective selection are fundamental to improving the performance of machine translation models, serving as cornerstones for achieving robust and reliable translation systems. This paper presents a data selection methodology specifically designed for fine-tuning machine translation systems, which leverages the synergy between a learner model and a pre-trained reference model to enhance overall training effectiveness. By defining a learnability score, our approach systematically evaluates the utility of data points for training, ensuring that only the most relevant and impactful examples contribute to the fine-tuning process. Furthermore, our method employs a batch selection strategy which considers interdependencies among data points, optimizing the efficiency of the training process while maintaining a focus on data relevance. Experiments on English ↔ Persian and several other language pairs using an mBART model fine-tuned on the CCMatrix dataset demonstrate that our method can achieve up to a fivefold improvement in data efficiency compared to an iid baseline. Experimental results indicate that our approach improves computational efficiency by 24% when utilizing cached embeddings, as it requires fewer training data points. Additionally, it enhances generalization, resulting in superior translation performance compared to random selection method.

1 Introduction

Machine translation is a fundamental task in natural language processing. As with any data-driven learning task, the effectiveness of training heavily depends on the quality of the data. (Fenza et al., 2021; Gupta et al., 2021; Chen et al., 2021) In particular, parallel datasets may contain irrelevant sentence pairs or poorly translated documents, which negatively impact the performance of the final model.

Beyond the quality of data, the state of the learner model itself plays a crucial role in selecting beneficial training data. For instance, studies have shown that data points associated with high loss on the learner model are typically those the model struggles to learn. (Bucher et al., 2016; Harwood et al., 2017) Allocating more computational resources to such data points, rather than to those the model has already mastered, can lead to more effective training.

Training can be made more data-efficient by employing selection methods during the training process, such as those based on the loss of data points on the learner model, a pre-trained model, or a combination of both.

We demonstrate that the batch-selection method is more effective than both the individual sample-selection and random selection method. More specifically, selecting data points within a batch, where the points are interdependent, is more effective than independently selecting high-scoring data points. Similar findings have also been reported in previous studies for multimodal learning (Evans et al., 2024). Our experiments focus on 12 different directions, namely, Persian ↔ English, German ↔ English, French ↔ English, Finnish ↔ English, Arabic ↔ English and Hindi ↔ English.

An mBART model (Liu, 2020) is used as the learner and a pre-trained LaBSE model (Feng et al., 2022) as the reference model. The pre-trained model is called *reference model*, while the model undergoing fine-tuning is called the *learner model*.

We use features extracted from both the learner model and the pre-trained model for selecting the data during the training. We employ the learnability score (Mindermann et al., 2022) to select data points for fine-tuning.

As demonstrated in our experiments, the use of the learnability score as a selection metric enables the model to generalize more effectively to the data, rather than overfitting. As a result, we achieved up to 5 times the data efficiency of random selection for English→Persian.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 26617–26625

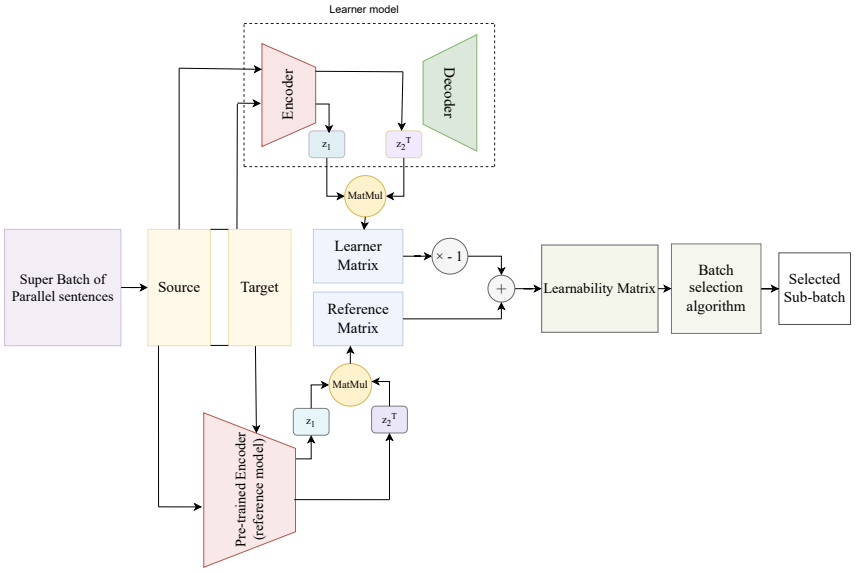


Figure 1: Our proposed method diagram for data selection in machine translation

For the remainder of this paper, we refer to training with randomly selected data as iid training. The paper is organized as follows: Section 2 reviews related work, Section 3 presents our methodology, Section 4 details results, and Section 5 concludes. Section 6 discusses limitations. Appendix A, Appendix B and Appendix C contains complementary material.

2 Related Work

Offline data selection: Traditional methods improve translation and efficiency by selecting parallel data subsets. Studies show that filtering harmful or low-quality data enhances NMT performance (Lam et al., 2022; Xu et al., 2019).

Online Data Selection: Fixed curation strategies may not adapt to evolving training needs. Online methods dynamically identify challenging examples, improving NMT by varying selected data across training epochs (Van Der Wees et al., 2017).

Hard Negative Mining: This technique enhances learning by focusing on difficult negative examples, widely used in computer vision and contrastive learning (Bucher et al., 2016; Harwood et al., 2017; Mishchuk et al., 2017; Simo-Serra et al., 2015; Wu et al., 2017; Xuan et al., 2020; Robinson et al., 2020; Tian et al., 2021). However, its application in machine translation remains underexplored.

Batch selection. Unlike sample selection, batch selection considers inter-data relationships. Evans et al. (2024) proposed an iterative batch selection method using learnability scores in multimodal datasets. Our work extends this concept to machine translation.

3 Methodology

3.1 Selection criteria

Our primary selection criterion is the learnability metric proposed by Mindermann et al. (2022), consisting of a hard learner score and an easy reference score. The hard learner score is assigned by the learner model, while the easy reference score is assigned by the reference model. We first sample a super-batch of data, ensuring equal selection probability, then choose a sub-batch based on the learnability metric and perform backpropagation.

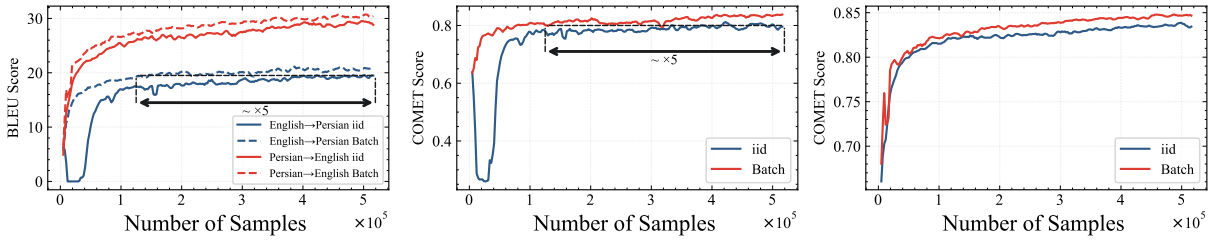
Effective parallel sentences exhibit closer embeddings in latent space, making similarity between embeddings a key selection factor. A low similarity on the learner model indicates unlearned data points, which should be prioritized. We define the hard learner score as

$$s^{hard}(B,\theta) = -H_\theta(B_{src})H_\theta(B_{trg}), \quad (1)$$

where θ denotes learner model parameters, B is the batch and $H_\theta(.)$ is the embedding matrix from the learner model. While effective for clean datasets (Paul et al., 2021), this heuristic can amplify noise in less curated datasets (Evans et al., 2025).

Data points with high similarity on a pre-trained model are typically learnable and high quality (Hessel et al., 2021; Schuhmann et al., 2022). Leveraging this, we filter noisy samples to mitigate overfitting. The easy reference score is defined as

$$s^{easy}(B,\theta^*) = H_{\theta^*}(B_{src})H_{\theta^*}(B_{trg}), \quad (2)$$



(a) English↔Persian BLEU Score (b) English→Persian COMET Score (c) Persian→English COMET Score

Figure 2: Comparison between our approach and independent and identically distributed (iid) training using BLEU and COMET-22 metrics on the filtered dataset.

Algorithm 1 Joint example selection

Input: M_l (learnability matrix), n_{chunks}, and $filter\_ratio$
$C \leftarrow 10^6$ // A large constant
$n_{rows} \leftarrow$ NUM_ROWS(M_l)
$n_{draws} \leftarrow \lfloor n_{rows} \times (1 - filter\_ratio)/n_{chunks} \rfloor$
$diag \leftarrow$ DIAGONAL(M_l)
$inds \leftarrow$ RANDOM_SAMPLE($diag, n_{draws}$)
for $z = 1$ **to** $n_{chunks} - 1$ **do**
  $is\_sampled \leftarrow$ LEARNABILITY_EYE($inds$)
  $s_{rows} \leftarrow$ SUM_ROWS($M_l \times is\_sampled$)
  $s_{cols} \leftarrow$ SUM_COLUMNS($M_l \times is\_sampled$)
  $probs \leftarrow diag + s_{rows} + s_{cols}$
  $probs \leftarrow probs - is\_sampled \times C$
  $inds' \leftarrow$ SAMPLE_WITH_PROBS($probs, n_{draws}$)
  $inds \leftarrow$ CONCATENATE($inds, inds'$)
return $inds$

where θ^* represents the reference model parameters. Combining both scores, learnability is defined as

$$s^{learn}(B|\theta, \theta^*) = s^{hard}(B, \theta) + s^{easy}(B, \theta^*). \quad (3)$$

This formulation proritizes unlearned data (high s^{hard}) while downweighting noise (low s^{easy}).

Similarity is computed as the dot product of sentence embedding from the learner and the reference model, forming matrices. Assuming a super-batch size of 2048 and embedding dimension of 1024, this results in $[2048, 1024]$ matrices for both source and target languages. The final similarity matrix, obtained by multiplying these matrices, has a dimension of $[2048, 2048]$. Using this matrix, we compute similarities and derive the learnability matrix via Equation (3).

After computing the learnability matrix, we employ the iterative batch selection algorithm (Algorithm 1) for obtaining the next sub-batch. The algorithm takes the learnability matrix, n_{chunks} (number of data points appended to final mini-batch in each iteration), and a filter ratio as input, outputting selected indices from the super-batch. This approach samples batches that are both learnable and previously unlearned by the model, improving data efficiency compared to individual sample selection, as demonstrated in our experiments.

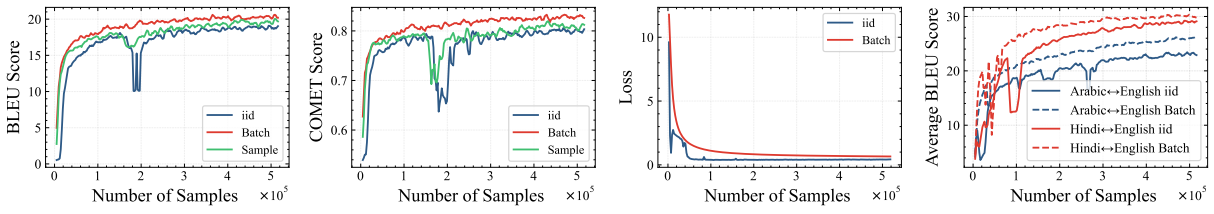
4 Experiments

To evaluate our method, we fine-tuned an mBART model on Persian ↔ English along with German ↔ English, French ↔ English, Finnish ↔ English, Arabic ↔ English and Hindi ↔ English subsets of the noisy CCMatrix dataset (Nikolova-Stoupak et al., 2022). We considered two settings for Persian ↔ English: (1) *raw dataset fine-tuning*, where mBART was trained on the unprocessed dataset, and (2) *curated dataset fine-tuning*, where CCMatrix was first filtered using LaBSE before applying our method. For other language pairs, we experiment with unfiltered dataset.

Our evaluation uses FLORES-200 (Guzmán et al., 2019), with all experiments conducted on its test set. We used a filtering ratio of 0.9, four chunks, a super-batch size of 4000, and a sub-batch size of 400, selecting 400 samples for updates. Learnability scores of 0.8 and 0.2 were used for the reference and learner similarity matrices, respectively. Smaller super-batches reduced effectiveness, nearing iid performance. Final results after training on about 0.5 million data points are shown in Table 1. As results demonstrate, batch selection enhances BLEU scores by 1.94 points for Persian→English direction, whereas it improves English→Persian BLEU score by 1.6 points.

As shown in Figure 2, our approach achieves comparable BLEU and COMET-22 scores to that of the iid training, while using approximately five times less data on English→Persian, demonstrating its data efficiency.

As depicted in Figure 3 (c), our batch selection method ensures smoother training loss and improved generalization. By dynamically selecting batches based on learnability, the model avoids



(a) English→Persian BLEU Score (b) English→Persian COMET Score (c) Loss between Batch selection and iid (d) Average BLEU score

Figure 3: (a, b) Comparison of our approach with iid training and individual sample training methods based using BLEU and COMET-22 metrics on the unfiltered dataset. (c) Batch selection is robust to overfitting on noisy data, especially in early stages of the training. (d) Comparison of Batch selection and iid on Arabic ↔ English and Hindi ↔ English. Each line represents the average of both to and from English directions for each language.

Method/Metric	English→Persian		Persian→English	
	BLEU	COMET-22	BLEU	COMET-22
Batch selection	**20.86**	**0.84**	**30.32**	**0.84**
iid	19.26	0.78	28.38	0.83

Table 1: Final metric for iid and batch selection after training on about 0.5 million data points for English ↔ Persian. Results are averaged over two seeds.

overfitting noisy data while maintaining a balanced dataset representation.

We evaluated our approach on unfiltered dataset to test robustness. As shown in Figure 3 (a) and (b), joint batch selection is more data-efficient than iid and individual selection, highlighting the benefit of learnability-based batching.

While our method involves more computation than iid training due to extra forward passes, it requires fewer samples to achieve similar performance, resulting in overall efficiency gains when caching reference embeddings (Table 2). Experiments were run on an NVIDIA 3090 GPU, using sub-batch chunks of 32 samples due to memory limits, though larger sub-batches may improve results further.

Method/Metric	Samples	Relative FLOPS
Batch selection	360,000	29.86
Batch selection (cached)	360,000	**0.76**
iid	1,159,200	1.00

Table 2: Relative floating-point operations with respect to iid training and the number of training samples required to achieve a BLEU score of 21 on the English→Persian test set.

4.1 Further experiments in other language pairs

We further evaluate the effectiveness of our method on additional language pairs and translation directions. As shown in Figure 4 and Figure 3 (d),

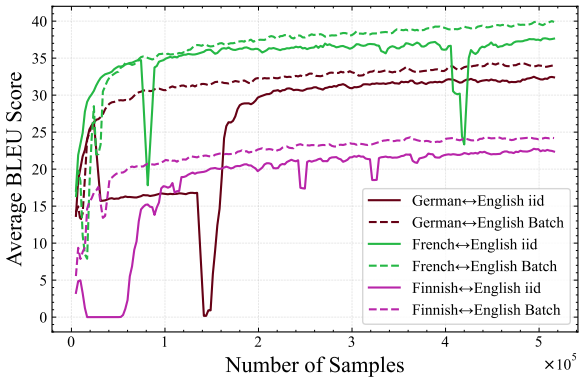


Figure 4: Comparison of our approach against iid training on German ↔ English, French ↔ English and Finnish ↔ Englsih. Each line represents the average of both to and from English directions for each language.

the results demonstrate the robustness of our approach across different languages.

Further experiments are presented in Appendix C.

5 Conclusion

We propose a novel online data selection method to improve machine translation fine-tuning. Using a learnability-based batch selection algorithm, our approach identifies data points that are informative yet not fully learned, enhancing training efficiency. Fine-tuning an mBART model on multiple language pairs, we observe improved performance over iid and individual selection strategies.

Our method shows greater resistance to overfitting and more stable loss trends, particularly in early training. By focusing on optimal learning samples, it boosts data and computational efficiency while ensuring stable parameter updates. This demonstrates the value of data selection in low-resource or noisy settings.

6 Limitations

A key limitation of any data selection method, including ours, is the additional computational overhead required to calculate the utility of individual data points. Our method requires greater computational resources compared to iid when training the model on an equivalent number of data points, particularly when embeddings are not cached. However, the key advantage of our approach lies in its data efficiency; it enables the learner model to achieve comparable performance with fewer data points than the iid training.

Nonetheless, our method may not be optimal in scenarios where a fixed, small, and carefully curated dataset is available. In such cases, iid training could be a more practical choice, as it eliminates the need for utility calculations and avoids the associated computational costs. This trade-off highlights the context-dependent applicability of our method, emphasizing its strengths in situations where data efficiency outweighs computational concerns.

7 Acknowledgements

The ChatGPT-4o Mini model was utilized exclusively for editing purposes in this study.

References

Maxime Bucher, Stéphane Herbin, and Frédéric Jurie. 2016. Hard negative mining for metric learning based zero-shot classification. In *Computer Vision–ECCV 2016 Workshops: Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III 14*. Springer.

Haihua Chen, Jiangping Chen, and Junhua Ding. 2021. Data evaluation and enhancement for quality improvement of machine learning. *IEEE Transactions on Reliability*, 70(2).

Talfan Evans, Nikhil Parthasarathy, Hamza Merzić, and Olivier J. Hénaff. 2024. Data curation via joint example selection further accelerates multimodal learning. In *Advances in Neural Information Processing Systems*, volume 37. Curran Associates, Inc.

Talfan Evans, Shreya Pathak, Hamza Merzic, Jonathan Schwarz, Ryutaro Tanno, and Olivier J Henaff. 2025. Bad students make great teachers: Active learning accelerates large-scale visual understanding. In *European Conference on Computer Vision*. Springer.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Dublin, Ireland. Association for Computational Linguistics.

Giuseppe Fenza, Mariacristina Gallo, Vincenzo Loia, Francesco Orciuoli, and Enrique Herrera-Viedma. 2021. Data set quality in machine learning: consistency measure based on group decision making. *Applied Soft Computing*, 106.

Nitin Gupta, Shashank Mujumdar, Hima Patel, Satoshi Masuda, Naveen Panwar, Sambaran Bandyopadhyay, Sameep Mehta, Shanmukha Guttula, Shazia Afzal, Ruhi Sharma Mittal, et al. 2021. Data quality for machine learning tasks. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. 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)*, Hong Kong, China. Association for Computational Linguistics.

Ben Harwood, Vijay Kumar BG, Gustavo Carneiro, Ian Reid, and Tom Drummond. 2017. Smart mining for deep metric learning. In *Proceedings of the IEEE international conference on computer vision*.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. CLIPScore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tsz Kin Lam, Eva Hasler, Amazon AI Translate, and Felix Hieber. 2022. Analyzing the use of influence functions for instance-specific data filtering in neural machine translation. *WMT 2022*.

Y Liu. 2020. Multilingual denoising pre-training for neural machine translation. *arXiv preprint arXiv:2001.08210*.

Dheeraj Mekala, Alex Nguyen, and Jingbo Shang. 2024. Smaller language models are capable of selecting instruction-tuning training data for larger language models. *arXiv preprint arXiv:2402.10430*.

Sören Mindermann, Jan M Brauner, Muhammed T Razzak, Mrinank Sharma, Andreas Kirsch, Winnie Xu, Benedikt Höltgen, Aidan N Gomez, Adrien Morisot, Sebastian Farquhar, et al. 2022. Prioritized training on points that are learnable, worth learning, and not yet learnt. In *International Conference on Machine Learning*. PMLR.

Anastasiia Mishchuk, Dmytro Mishkin, Filip Radenovic, and Jiri Matas. 2017. Working hard to know your neighbor's margins: Local descriptor learning loss. *Advances in neural information processing systems*, 30.

Iglika Nikolova-Stoupak, Shuichiro Shimizu, Chenhui Chu, and Sadao Kurohashi. 2022. Filtering of noisy web-crawled parallel corpus: the japanese-bulgarian language pair. In *Proceedings of the 5th International Conference on Computational Linguistics in Bulgaria (CLIB 2022)*.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. 2021. Deep learning on a data diet: Finding important examples early in training. *Advances in neural information processing systems*, 34.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Joshua Robinson, Ching-Yao Chuang, Suvrit Sra, and Stefanie Jegelka. 2020. Contrastive learning with hard negative samples. *arXiv preprint arXiv:2010.04592*.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. 2022. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35.

Edgar Simo-Serra, Eduard Trulls, Luis Ferraz, Iasonas Kokkinos, Pascal Fua, and Francesc Moreno-Noguer. 2015. Discriminative learning of deep convolutional feature point descriptors. In *Proceedings of the IEEE international conference on computer vision*.

Yonglong Tian, Olivier J. Hénaff, and Aäron van den Oord. 2021. Divide and contrast: Self-supervised learning from uncurated data. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*.

Marlies Van Der Wees, Arianna Bisazza, and Christof Monz. 2017. Dynamic data selection for neural machine translation. *arXiv preprint arXiv:1708.00712*.

Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. 2017. Sampling matters in deep embedding learning. In *Proceedings of the IEEE international conference on computer vision*.

Guanghao Xu, Youngjoong Ko, and Jungyun Seo. 2019. Improving neural machine translation by filtering synthetic parallel data. *Entropy*, 21(12).

Hong Xuan, Abby Stylianou, Xiaotong Liu, and Robert Pless. 2020. Hard negative examples are hard, but useful. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*. Springer.

A Appendix A: Using smaller models as reference model

To explore computational efficiency, we replaced LaBSE with Distiluse (Reimers and Gurevych, 2019) as the reference model. Although Distiluse is significantly smaller, it remained effective for data selection, as shown in Figure 5. Furthermore, we applied 4-bit quantization to this model to reduce inference resource requirements. These modifications enabled us to maintain performance while significantly lowering the computational overhead.

This experiment demonstrates that small models are capable of effectively selecting data points for training larger models, as shown in Mekala et al. (2024). This finding highlights the potential of lightweight models in reducing computational costs while maintaining the quality of data selection.

Although smaller models exhibit slight instability at the beginning of training, this issue may be mitigated by adjusting the weights assigned to the learner and reference matrices.

B Appendix B: Examining learner and reference scores

As stated in the earlier sections, we use dot products between embeddings of the source and target languages as a measure of similarity, where values range between -1 and 1. These scores are then utilized for data selection. For instance, suppose a parallel sentence receives a score of -1 from the learner model. According to Section 3, we multiply this value by -1, yielding a score of 1. This implies that such a sentence is assigned high priority, despite having an opposite meaning to its counterpart. This scenario could arise if the dataset contained a significant number of parallel sentences with reversed meanings. However, in our case, an analysis of the score distribution demonstrates that this is not the case. Specifically, by measuring and plotting the distribution of dot product values, we observe that very few data points fall below 0, while the majority of dot product values exceed 0.8 for both models, as illustrated in Figure 6.

Furthermore, as depicted in Figure 6, the distribution of dot product values for the learner model exhibits a lower mean and higher variance compared to the reference model. This suggests that the learner model remains weaker in its ability to generate aligned embeddings. Ideally, a perfect dataset, when evaluated with a perfect model, would produce a sharp peak at 1, representing an impulse function, indicating that all parallel sentences align perfectly.

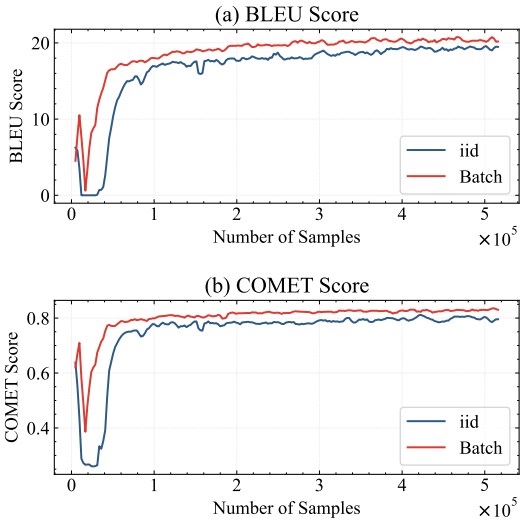


Figure 5: We utilize a smaller model as a reference model, apply quantization to it, and demonstrate superior performance compared to iid.

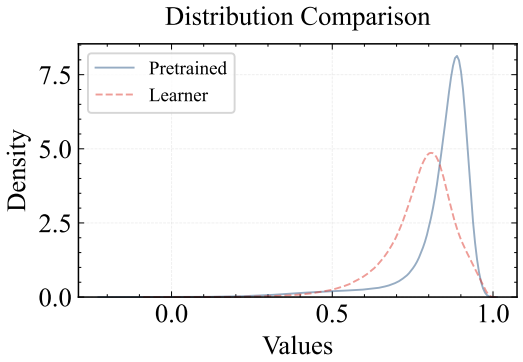


Figure 6: Distribution of dot products between the embeddings of source and target sentences.

C Appendix C: Experiment details

In this section, we present a comprehensive analysis of the experimental results obtained using our proposed method. We provide a detailed comparison of the performance across various language pairs to highlight the effectiveness and robustness of our approach in multilingual settings.

Figure 7 illustrates the outcomes for translation tasks from English to German, French, and Finnish. We include BLEU score and COMET score to provide a clear view of the model's strengths.

On the other hand, Figure 8 reports the same set of metrics, but this time for the reverse direction $xx \rightarrow$English. This comparison is particularly important for understanding whether the model exhibits any directional bias or asymmetry in trans-

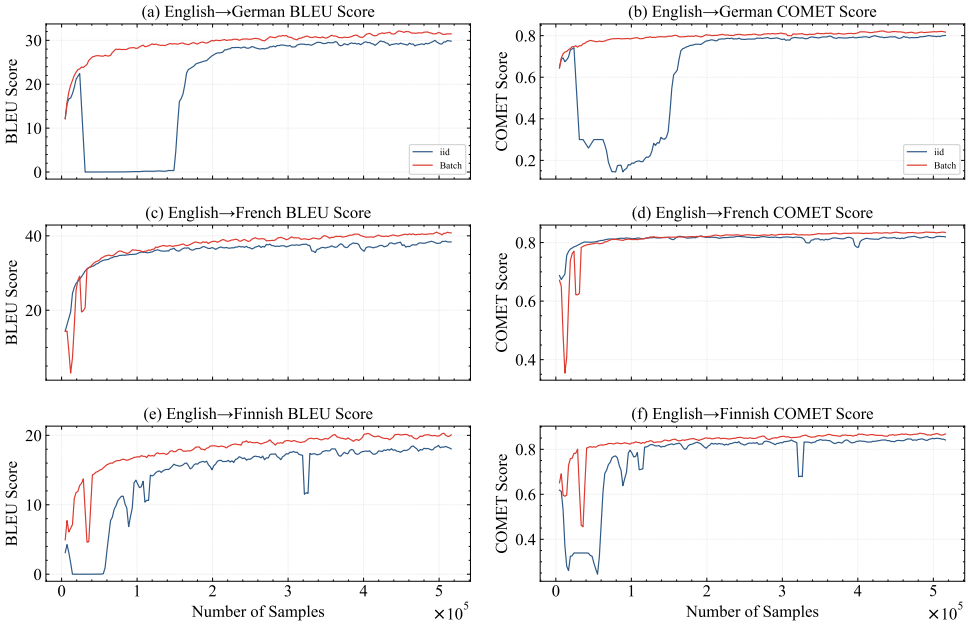


Figure 7: Comparison of our approach with iid training on English→German, English→French and English→Finnish.

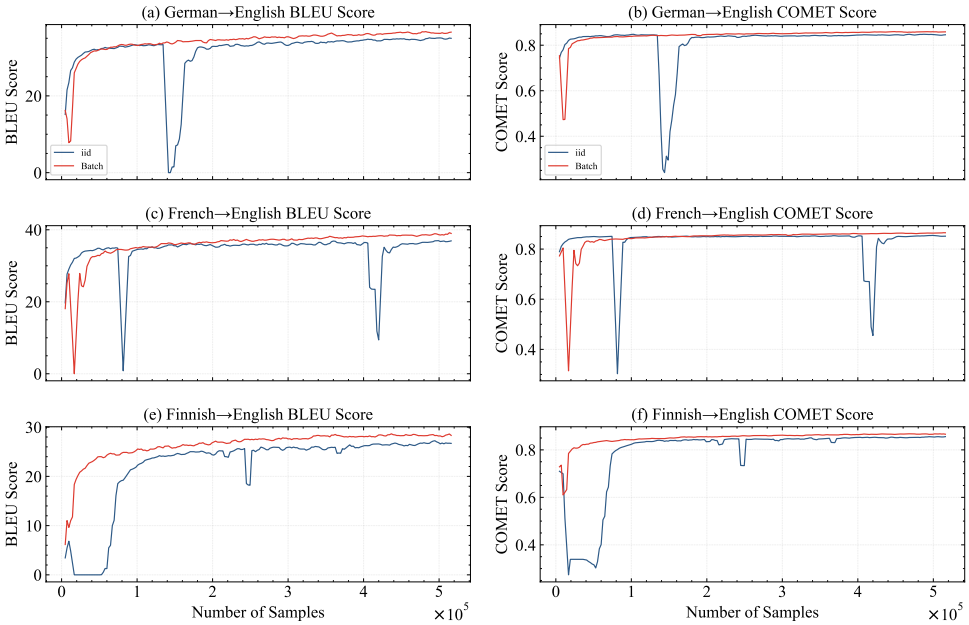


Figure 8: Comparison of our approach with iid training on German→English,French→English and Finnish→English.

lation quality. Notably, the performance in this direction provides insights into the model's ability to decode diverse linguistic structures back into English.

Furthermore, Figure 9 presents our experimental findings for low-resource languages, specifically Arabic and Hindi. For these languages, we evaluate the model's performance in both translation directions—into and out of English. This helps us assess the model's generalization capability on typologically distinct languages with limited training data.

Finally, Table 3 summarizes the final results of the model for the Arabic and Hindi translation task after the completion of training.

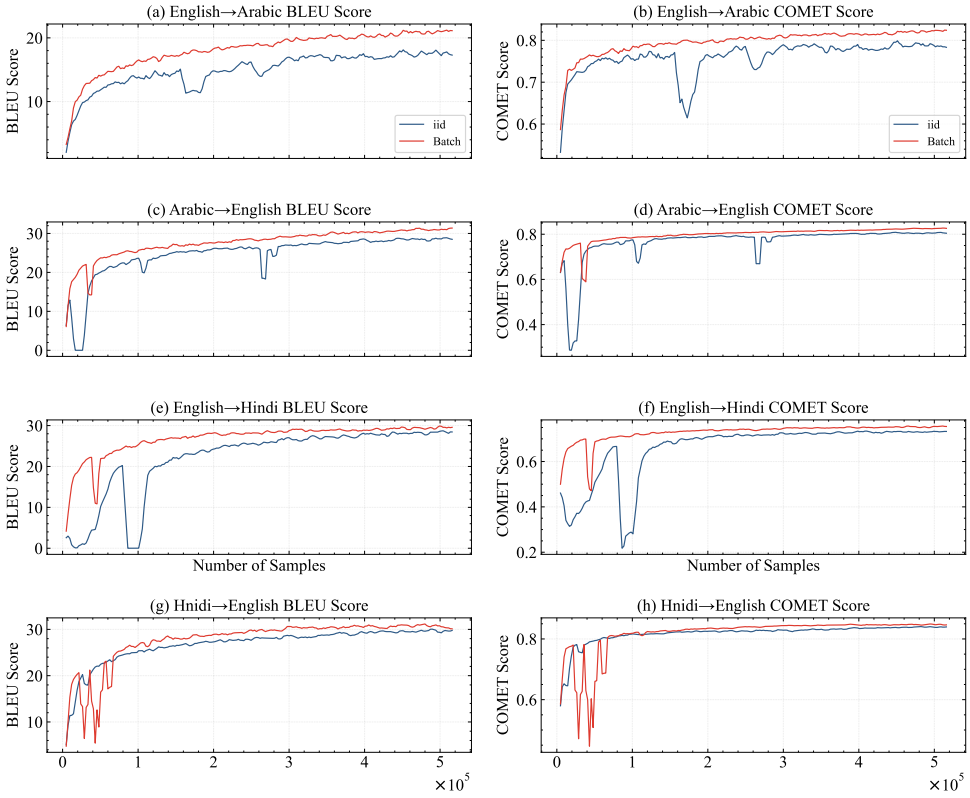


Figure 9: Comparison of our approach with iid training on Arabic and Hindi.

Language	Method	BLEU	COMET-22
English→Arabic	Batch selection	**21.02**	**0.82**
	iid	17.43	0.78
Arabic→English	Batch selection	**31.34**	**0.82**
	iid	28.59	0.80
English→Hindi	Batch selection	**29.52**	**0.75**
	iid	28.53	0.73
Hindi→English	Batch selection	**30.10**	**0.84**
	iid	29.94	**0.84**

Table 3: Final metric for iid and batch selection after training on about 0.5 million data points for Arabic and Hindi.