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

Anonymous ACL submission

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-to-Persian translation using an mBART model fine-tuned on the CCMatrix dataset demonstrate that our method achieves 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 iid methods.

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; Kumar 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.

Furthermore, we demonstrate that the batch-selection method is more effective than the individual sample-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. Our experiments focus on English-to-Persian translation, leveraging an mBART fine-tuned on the CCMatrix dataset.

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

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

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, with supple-

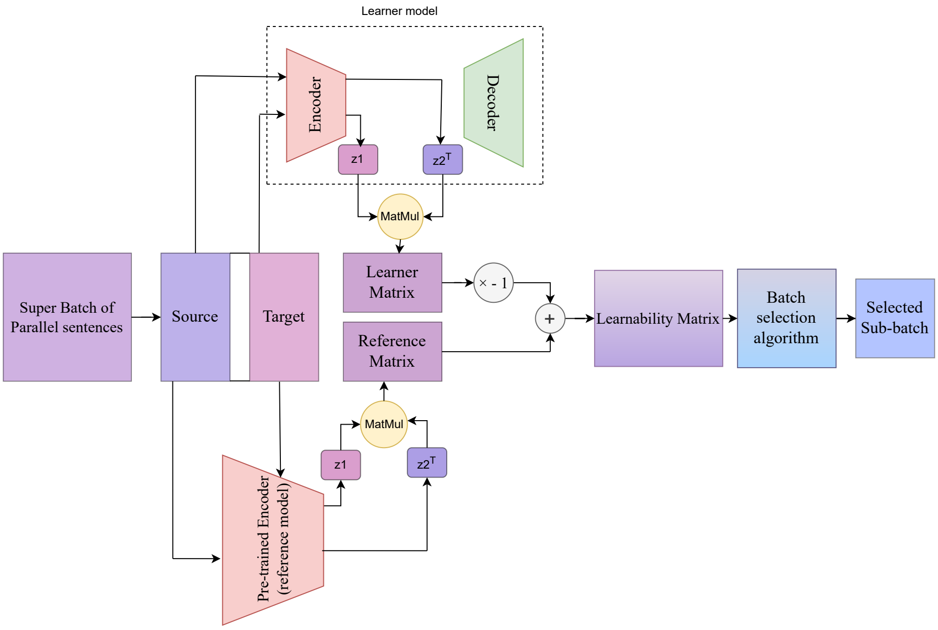

Figure 1: Diagram illustrating our proposed method for data selection in machine translation

mentary materials in Appendix A and Appendix B.

2 Related Work

Offline data selection: Traditional methods focus on selecting parallel data subsets to enhance translation quality and reduce resource consumption. Several studies highlight the role of data filtering in improving NMT, such as using influence functions to remove harmful examples (Lam et al., 2022) and filtering low-quality synthetic data to boost accuracy (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; Kumar 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:

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

where θ denotes learner model parameters, B is the batch, M represents matrix multiplication, 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 overfit-

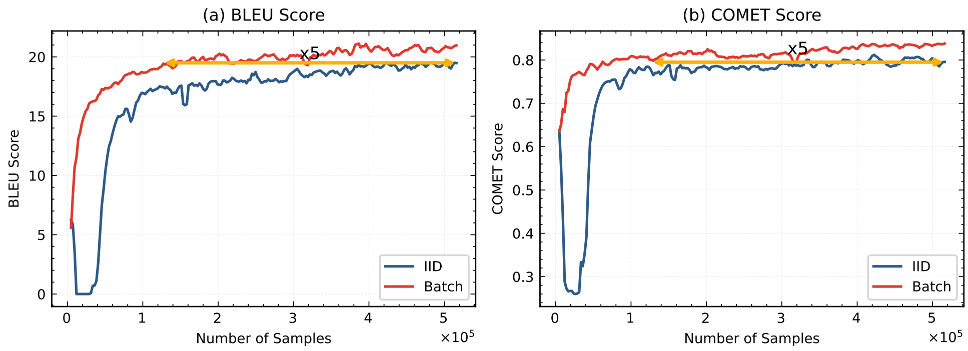

Figure 2: Comparison between our approach and independent and identically distributed (iid) training on BLEU and COMET-22 metrics on the filtered dataset.

Algorithm 1 Joint example selection

Input: learnability_matrix, n_{chunks}, filter_ratio, M (a large constant)
Output: sampled indices $inds$

$superb\_s \leftarrow$ NUM_ROWS(learnability_matrix)
$n_{draws} \leftarrow \lfloor superb\_s \times (1 - filter\_ratio)/n_{chunks} \rfloor$
$pos\_ii \leftarrow$ DIAGONAL(learnability_matrix)
$inds \leftarrow$ RANDOM_SAMPLE(pos_ii, n_{draws})
for $i = 1$ **to** $n_{chunks} - 1$ **do**
  $is\_sampled \leftarrow$ LEARNABILITY_EYE($inds$)
  $pos\_ij \leftarrow$ SUM_ROWS($is\_sampled$)
  $pos\_ji \leftarrow$ SUM_COLUMNS($is\_sampled$)
  $pos \leftarrow pos\_ii + pos\_ij + pos\_ji$
  $pos \leftarrow pos - is\_sampled \times M$
  $new\_inds \leftarrow$ SAMPLE_WITH_PROBS(pos, n_{draws})
  $inds \leftarrow$ CONCATENATE($inds$, new_inds)

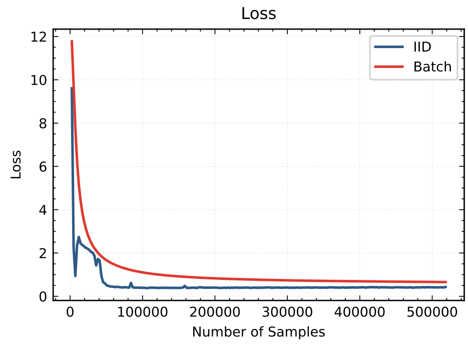

Figure 3: Batch-selection using learnability score has a smoother learning loss and better generalization.

ting. The easy reference score is defined as:

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

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 prioritizes unlearned data (high s^{hard}) while filtering noise (i.e. high 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 dimensions $[2048, 2048]$. 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 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 the English-Persian subset of the noisy CCMatrix dataset (Nikolova-Stoupak et al., 2022). We considered two settings: (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.

Our evaluation used the Persian-English subset of FLoRes-200 (Guzmán et al., 2019), with all experiments conducted on its test set. As shown in Figure 2, our approach achieves comparable BLEU and COMET-22 scores to that of the iid training while using five times less data, demonstrating its

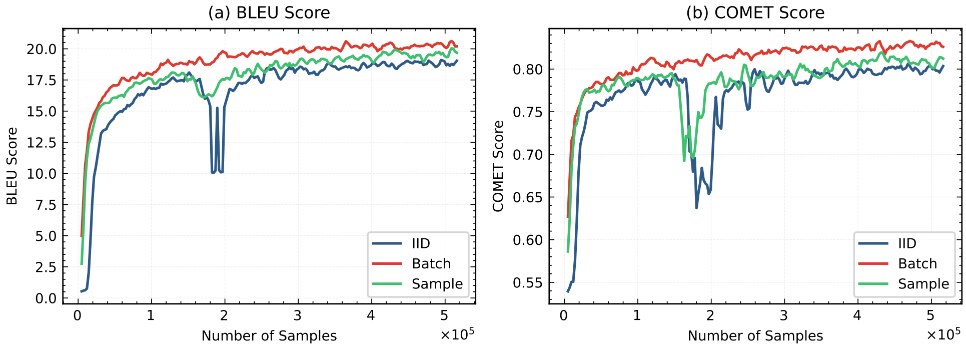

Figure 4: Comparison of our approach with independent and identically distributed (iid) and individual sample training methods based on BLEU and COMET-22 metrics on the unfiltered dataset.

data efficiency.

Method/Metric	BLEU	COMET-22
Batch Selection	**20.86**	**0.84**
iid	19.26	0.78

Table 1: Final metric for iid and batch selection after training on 518000 data points.

For Figure 2, we set a filtering ratio of 0.9 and set the number of chunks to 4, used a super-batch size of 4000 with a sub-batch size of 400. From these, 400 samples were selected for model updates. The learnability score was computed by assigning a weight of 0.2 to the learner model's similarity matrix and 0.8 to the reference model's similarity matrix. We observed reduced effectiveness for smaller super-batches, with performance approaching that of the iid training. The final results after training on 0.5M data points are depicted in Table 1.

As depicted in Figure 3, our batch selection method ensures smoother training loss and improved generalization. By dynamically selecting batches based on learnability, the model avoids overfitting noisy data while maintaining a balanced dataset representation.

We further evaluated our approach on unfiltered datasets to assess its robustness. As seen in Figure 4, joint batch selection outperforms iid and individual sample selection in stability and data efficiency, emphasizing the advantage of using learnability-based batch selection.

Although our method requires more computation than iid training due to additional forward passes, fewer samples are needed to achieve comparable performance, leading to overall efficiency gains when caching the reference model embeddings instead of recalculating them (Table 2). Experiments were conducted on an NVIDIA RTX 3090 GPU with 24GB VRAM. Due to memory constraints, we processed sub-batches in chunks of 32 samples, though processing the full sub-batch at once could yield further improvements.

Method/Metric	Samples	Relative FLOPS
Batch Selection	360,000	29.86
Batch Selection (Cached)	360,000	**0.76**
iid	1,159,200	1

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 test set.

5 Conclusion

We proposed a method for online data selection for fine-tuning machine translation, employing a batch selection algorithm to identify learnable data points—data points that the model has not yet learned but are not noise. Using an mBART model, we fine-tuned it on the English-to-Persian section of CCMatrix, demonstrating improved data efficiency compared to traditional iid training and individual sample selection methods. Our approach proved effective on both uncurated and curated datasets, showcasing its versatility.

Our learnability-based batch selection approach improved robustness against overfitting, especially in early training, and produced a smoother loss curve. This demonstrates its potential to enhance data and computation efficiency in machine translation fine-tuning while ensuring robust performance across diverse datasets.

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.

Additionally, we need to conduct experiments on more language pairs to verify the effectiveness of our method across different languages.

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*, pages 524–531. 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):831–847.

Talfan Evans, Nikhil Parthasarathy, Hamza Merzic, and Olivier J Henaff. 2024. Data curation via joint example selection further accelerates multimodal learning. *arXiv preprint arXiv:2406.17711*.

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*, pages 264–280. Springer.

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

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*, 106:107366.

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*, pages 4040–4041.

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)*, pages 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. *arXiv preprint arXiv:2104.08718*.

VB Kumar, Ben Harwood, Gustavo Carneiro, Ian Reid, and Tom Drummond. 2017. Smart mining for deep metric learning. *arXiv preprint arXiv:1704.01285*, 2.

Tsz Kin Lam, Eva Hasler, and Felix Hieber. 2022. Analyzing the use of influence functions for instance-specific data filtering in neural machine translation. *arXiv preprint arXiv:2210.13281*.

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*, pages 15630–15649. 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)*, pages 39–48.

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*, 34:20596–20607.

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*, 35:25278–25294.

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*, pages 118–126.

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)*, pages 10043–10054.

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*, pages 2840–2848.

Guanghao Xu, Youngjoong Ko, and Jungyun Seo. 2019. Improving neural machine translation by filtering synthetic parallel data. *Entropy*, 21(12):1213.

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*, pages 126–142. 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.

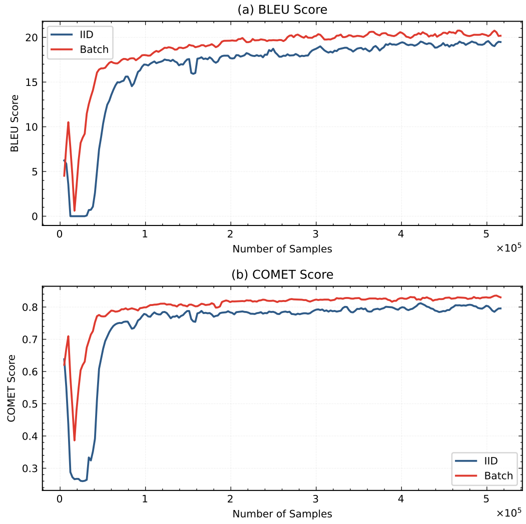

Figure 5: We utilize a smaller model as a reference model, apply quantization to it, and demonstrate superior performance compared to iid.

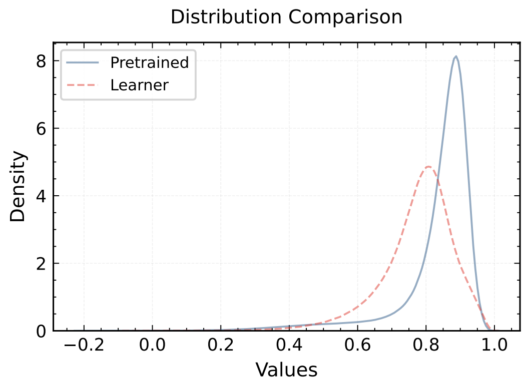

Figure 6: Distribution of dot products between the embeddings of source and target sentences.