Multi-Hypothesis Distillation of Multilingual Neural Translation Models for Low-Resource Languages

AARÓN GALIANO-JIMÉNEZ*, Universitat d'Alacant, Spain
JUAN ANTONIO PÉREZ-ORTIZ, Universitat d'Alacant, Spain
FELIPE SÁNCHEZ-MARTÍNEZ, Universitat d'Alacant, Spain
VÍCTOR M. SÁNCHEZ-CARTAGENA, Universitat d'Alacant, Spain

This paper explores knowledge distillation (KD) of multilingual pre-trained encoder-decoder and large language models for machine translation (MT). We argue that the teacher model's output distribution holds valuable insights for the student, beyond the approximated mode obtained through beam search (the standard decoding method for MT), and present Multi-Hypothesis Distillation (MHD), a sequence-level KD method that generates multiple translations for each source sentence. This provides a larger representation of the teacher model distribution and exposes the student model to a wider range of target-side prefixes. We explore *n*-best lists from beam search to guide the student's learning and examine alternative decoding methods to address issues like low variability and the under-representation of infrequent tokens. For low-resource languages, our research shows that while sampling methods may slightly compromise translation quality compared to beam search based approaches, they enhance the generated corpora with greater variability and lexical richness. This ultimately improves student model performance and mitigates the gender bias amplification often associated with KD. Our experiments show that MHD improves over existing KD techniques both when training encoder-decoder students from scratch and when distilling into pre-trained decoder-only models.

JAIR Associate Editor: Huang Xuanjing

JAIR Reference Format:
Aarón Galiano-Jiménez, Juan Antonio Pérez-Ortiz, Felipe Sánchez-Martínez, and Víctor M. Sánchez-Cartagena. 2026. Multi-Hypothesis Distillation of Multilingual Neural Translation Models for Low-Resource Languages. *Journal of Artificial Intelligence Research* 87, Article 7 (September 2026), 33 pages. DOI: 10.1613/jair.1.22534

1 Introduction

Machine translation (MT) is an essential tool for communication and understanding among speakers of different languages. Over the past decade, the dominant architecture for MT has been the encoder-decoder Transformer (Vaswani et al. 2017). While encoder-decoder MT models excel in high-resource languages, they struggle with low-resource languages due to the limited availability of parallel training data (Goyal et al. 2020). This data scarcity problem becomes even more pronounced with large language models (LLMs), which have emerged as a powerful alternative to encoder-decoder models when enough training data is available. Consequently, although LLMs demonstrate strong translation performance in high-resource languages (Kocmi et al. 2024), they still lag behind traditional encoder-decoder models in several low-resource languages (Iyer et al. 2024; Scalvini et al. 2025;

*Corresponding Author.

Authors' Contact Information: Aarón Galiano-Jiménez, ORCID: 0000-0002-8107-1411, aaron.galiano@ua.es, Universitat d'Alacant, Spain; Juan Antonio Pérez-Ortiz, ORCID: 0000-0001-7659-8908, japerez@ua.es, Universitat d'Alacant, Spain; Felipe Sánchez-Martínez, ORCID: 0000-0002-2295-2630, fsanchez@ua.es, Universitat d'Alacant, Spain; Víctor M. Sánchez-Cartagena, ORCID: 0000-0001-9600-6885, vm.sanchez@ua.es, Universitat d'Alacant, Spain.

This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2026 Copyright held by the owner/author(s).
DOI: 10.1613/jair.1.22534

W. Zhu et al. 2024). In this context, encoder-decoder multilingual translation models like NLLB-200 (NLLB Team et al. 2022), M2M-100 (Fan, Bhosale, et al. 2021), and MADLAD-400 (Kudugunta et al. 2023) outperform bilingual models trained from scratch, primarily due to positive transfer learning from high-resource languages (Tran et al. 2021). However, despite their superior performance, state-of-the-art multilingual models remain too large and computationally demanding for widespread use, especially in resource-constrained environments like laptops or smartphones.

An approach to address this challenge is knowledge distillation (KD) (Hinton et al. 2015), which consists of transferring knowledge from a *teacher* model to a smaller *student* model. Typically, the distillation process relies on the same corpus used to train the teacher model. However, this corpus is usually unavailable for most pre-trained models, whether they are encoder-decoders or LLMs. Even if the original training data were available, the knowledge of the teacher model is also derived from transfer learning across multiple languages. As a result, extracting all relevant knowledge solely from a parallel corpus may not be feasible. In the absence of the parallel corpus used to train the teacher, one common sequence-level (Kim and Rush 2016) KD strategy is to translate a monolingual text (Lai et al. 2021; Yu et al. 2021) using beam search (Graves 2012), as this is the most widely used decoding method at inference time for MT. The resulting synthetic corpus can then be used to train a student model. However, this method has several limitations. Beam search looks for the translation with the highest probability, i.e. the mode of the probability distribution (Eikema and Aziz 2020). The mode represents a very small probability mass, and choosing a likely translation close to the mode leads to outputs with low lexical diversity (Kulikov et al. 2019), over-representation of frequent tokens, and under-representation of rare tokens (Müller and Sennrich 2021). In addition, it can amplify the biases present in the model, such as gender bias (Vamvas and Sennrich 2021).

We hypothesise that, contrary to the claim that knowledge comes from the top-1 prediction of the teacher (S. Zhang et al. 2023), sampling a broader range of the model's probability distribution can yield higher quality synthetic corpora for KD. Although word-level KD (Hinton et al. 2015) takes advantage of this distribution, it is limited to target language prefixes present in the reference translation and is therefore also affected by limited diversity and biases. To overcome these limitations, we introduce Multi-Hypothesis Distillation (MHD), a method that uses multiple translations per source sentence to provide a broader representation of the teacher's probability distribution and expose the student model to a wider range of target-side prefixes. To generate these translations, we explore different decoding methods to get the most out of the teacher model. This approach requires only monolingual corpora and supports KD via API access, even when the teacher model itself is inaccessible. This paper evaluates MHD when distilling both into encoder-decoder models trained from scratch and into pre-trained LLMs. It analyses the key characteristics of the resulting synthetic corpora and their impact on training student models. Our results demonstrate that MHD improves student model performance over sequence-level KD, even when the translations used are of lower quality than the top-1 prediction obtained with beam search, while it reduces the amplification of gender bias typically associated with KD (Ahn et al. 2022). We also show that the choice of decoding method should consider factors such as the availability of monolingual data and the translation quality of the teacher model.[1]

The rest of the paper is organised as follows. Section 2 reviews related work on knowledge distillation and decoding methods. Section 3 introduces our proposed MHD approach, formalising the training objective and detailing how multiple hypotheses are generated. Section 4 describes the experimental setup, including

[1]Part of this work was previously published as a Findings paper at the 2025 Annual Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics (NAACL) (Galiano-Jiménez, Pérez-Ortiz, et al. 2025), where we presented our initial approach. In the present work, we introduce word-level KD as a baseline, better formalise the MHD method, incorporate Minimum Bayes' Risk decoding (Kumar and Byrne 2004) (MBR) as an alternative decoding strategy, extend our analysis of hypothesis exploration with experiments using up to 100 translations per source sentence and evaluate hallucination phenomena. We also extend our experiments to include LLMs and other state-of-the-art KD strategies.

decoding methods, language pairs, corpora, and evaluation metrics. Section 5 presents and analyses the results of our experiments, covering the impact of amount of hypotheses, corpus size, decoding variability, gender bias, hallucinations and the comparison of MHD against KD techniques for LLMs.[2] Finally, Section 6 concludes the paper and outlines the main findings.

2 Related Work

There is an extensive literature on knowledge distillation and decoding methods, but the impact of decoding methods on the distillation process has been understudied. In what follows we describe the related work on KD (Sec. 2.1) and the role of decoding methods (Sec. 2.2).

2.1 Knowledge Distillation Techniques for Machine Translation

KD techniques can be classified into word level (Hinton et al. 2015) and sequence level (Kim and Rush 2016). Word-level KD mimics the teacher's probability distribution for each token, while sequence-level KD trains the student model using a synthetic corpus. This corpus is generated by the teacher by translating the source side of the original training corpus using beam search or other decoding algorithm such as Minimum Bayes' Risk decoding (Kumar and Byrne 2004) (MBR). In both cases, the same corpus used to train the teacher is used for the distillation. The difference is that sequence-level KD calculates the cross-entropy loss over the synthetic target side of the parallel corpus and word-level uses a combination of the cross-entropy loss over the real target and the Kullback-Leiber divergence (Kullback and Leibler 1951) between teacher and student probability output distributions. This means that, unlike sequence-level KD, word-level KD requires access to parallel data and greater computational resources, as both the teacher and student models must be loaded into memory simultaneously.

Regarding research on sequence-level KD with encoder-decoder multilingual translation models, some studies employ multiple teacher models, either multilingual (Do and Lee 2023) or bilingual (Tan et al. 2019), to distil knowledge into a multilingual student. In contrast, our approach aims to extract as much knowledge as possible of a language pair from a single teacher in order to train a bilingual student. Gumma et al. (2023) also use a single multilingual teacher, but they train a multilingual student model. Similarly, De Gibert et al. (2023) distil a high-resource language pair from NLLB-200 and then fine-tune the resulting student model on a set of low-resource language pairs. Some methods use high-resource languages related to low-resource ones for distillation, training the student with both languages as sources and English as the target (Song, Ezzini, et al. 2023). In contrast, our study is not limited to English as the target language. Galiano-Jiménez, Sánchez-Martínez, et al. (2023) fine-tune the teacher for specific language pairs and then train the student model with a mix of parallel and synthetic data. This method is based on parallel corpora, as well as back and forward-translation. This differs from our approach, which only uses monolingual corpora and forward-translation.

Regarding KD of LLMs for MT, several techniques typically rely on a pre-trained student rather than training from scratch. Enis and Hopkins (2024) used translations generated by Claude 3 Opus[3] to further fine-tune an NLLB-200 1.3B model that had already been fine-tuned with translations from NLLB-200 54B and a parallel corpus. However, this additional fine-tuning did not lead to significant improvements in translation quality. In an attempt to minimise the *exposure bias* (Ranzato et al. 2015), Agarwal et al. (2024) presented Generalized Knowledge Distillation (GKD), which combines word-level KD with both off-policy and on-policy training signals. In the off-policy component, the student is trained on sequences generated by the teacher and gold references. In addition, GKD introduces an on-policy loss that measures the discrepancy between teacher and student outputs when conditioning on prefixes generated by the student itself. This approach helps the student model learn

[2]The code is available at https://github.com/transducens/sampling-distillation
[3]https://www-cdn.anthropic.com/de8ba9b01c9ab7cbabf5c33b80b7bbc618857627/Model_Card_Claude_3.pdf

to generate sequences from its own predictions. Another approach based on word-level KD is Self-Evolution KD (Song, Ding, et al. 2025). This technique dynamically balances teacher distributions and ground-truth signals according to the learning difficulty of each token. While these methods generally outperform sequence-level KD, the improvements are often modest and evaluations are usually restricted to high-resource language pairs. This leaves the behaviour of these methods in truly low-resource MT largely unexplored. Additionally, both methods require the teacher and student models to be kept in memory during training, demanding significantly more resources than our approach. Following a sequence-level approach, MT-PATCHER (J. Li et al. 2024) proposes selective correction instead of full-output imitation, demonstrating that targeted supervision can outperform full sequence-level training with significantly fewer examples. This method is based on the assumption that the student model can correctly translate most source sentences and that it is more effective to focus on translations that the student generates poorly. However, this assumption rarely holds when working with low-resource languages (Zhao et al. 2025).

2.2 The Role of Decoding Methods

Neural MT models generate output tokens by producing a probability distribution over the target vocabulary at each decoding step. There are two approaches for selecting these output tokens: deterministic methods, which prioritise high-probability tokens but offer low variability (Kulikov et al. 2019), and stochastic methods, which sample from the probability distribution but can lead to incoherent text (Basu et al. 2021). For *directed generation* tasks, such as MT, beam search (Graves 2012) is commonly used because the output is closely tied to the input, and variability is less critical. In contrast, *open-ended* tasks, like conversational chatbots, require more diverse and human-like output (Holtzman et al. 2020). Although it is common to generate the output of an LLM using sampling methods for all tasks, beam search gives better results for MT (Shi et al. 2024). While several studies analyse decoding methods (DeLucia et al. 2021; Su et al. 2022; Wiher et al. 2022) and evaluate the quality of the resulting text (Pillutla et al. 2021), their focus on LLMs and open-ended tasks limits their applicability to MT with encoder-decoder models.

Nevertheless, MBR (Kumar and Byrne 2004) has recently been used to generate sequences with an LLM to fine-tune an encoder-decoder translation model (Finkelstein and Freitag 2024) and the conclusion was that the student model fine-tuned with these sequences outperforms the model fine-tuned with beam search translations. J. Wang, Briakou, et al. (2024) extended this approach by incorporating multiple translation candidates per source sentence, similar to our proposed method. They agree with our findings in that extracting multiple sentences from the teacher model better captures its probability distribution, leading to improved student models. However, our work explores a broader range of scenarios (language pairs and different decoding methods) and concludes that the choice of decoding method should depend on both the teacher model's translation quality and the size of the available corpus.

While the relationship between corpus quality and variability has been explored in open-ended tasks (H. Zhang et al. 2021), it is understudied for KD in MT. However, the variability produced by different decoding methods was investigated for back-translation, and it was concluded that top-p (Holtzman et al. 2020) results in higher final model performance (Burchell et al. 2022).

3 Approach: Multi-Hypothesis Knowledge Distillation

In this section we formalise the neural MT training objective based on Maximum Likelihood Estimation (MLE), the standard training objective for an MT model, and its adaptation to sequence-level KD. Then, we describe our proposed method, which generates multiple translation hypotheses per source sentence using different decoding strategies.

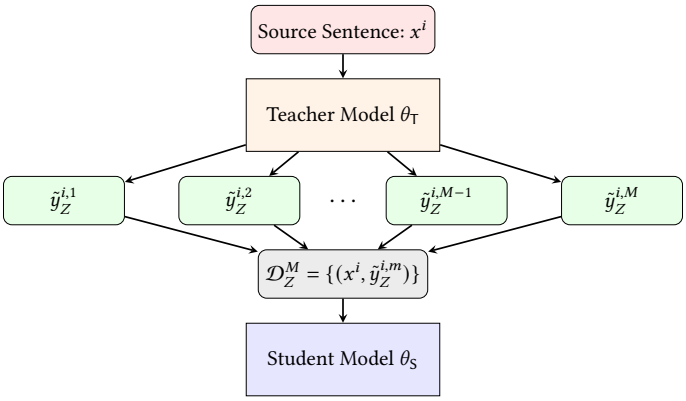

Fig. 1. Multi-Hypothesis Knowledge Distillation (MHD) approach. The teacher model generates M translations per source sentence, using Z as decoding method. The resulting dataset $\mathcal{D}_Z^M$ is then used to train the student model.

3.1 Maximum Likelihood Estimation

Given a training dataset $\mathcal{D} = \{(x^i, y^i)\}_{i=1}^N$, where x^i is a source sentence, y^i is its corresponding target translation and N corresponds to the number of sentences in the dataset, the training objective is to maximise the likelihood of the target sequence under the distribution of the model by minimising the following loss function:

$$\mathcal{L}_{\text{MLE}} = -\sum_{i=1}^{N}\sum_{t=1}^{T_i} \log P(y_t^i \mid y_{<t}^i, x^i; \theta) \tag{1}$$

Where θ represents the model parameters, T_i corresponds to the number of tokens of the i-th target sequence, $y_{<t}$ denotes the prefix of the i-th target sequence up to time step $t-1$, and y_t^i is the target token at time step t.

3.2 Sequence-Level Knowledge Distillation

In sequence-level KD, a teacher model θ_T generates a synthetic parallel corpus $\mathcal{D} = \{(x^i, \tilde{y}^i)\}$, where $\tilde{y}^i$ is a translation generated by the teacher model's distribution $P(y \mid x; \theta_T)$. The student model θ_S is then trained to mimic the teacher's outputs by minimising Equation (1) when replacing y^i by $\tilde{y}^i$.

3.3 Multi-Hypothesis Knowledge Distillation

To increase the variety of training data we propose generating multiple translation hypotheses $\tilde{\mathcal{Y}}_Z^i = \{\tilde{y}_Z^{i,1}, ..., \tilde{y}_Z^{i,M}\}$ per source sentence x^i, where Z is the decoding method used to translate. The produced dataset is:

$$\mathcal{D}_Z^M = \bigcup_{i=1}^{N}\bigcup_{m=1}^{M}\{(x^i, \tilde{y}_Z^{i,m})\} \tag{2}$$

This means that each source sentence x^i appears M times in the dataset, paired with different translations $\tilde{y}^{i,m}$ generated by the teacher model using Z as the decoding method. Appendix A details the generation of M translations with each decoding method. Training with this dataset results in the following loss function:

$$\mathcal{L}_{\text{MHD}} = -\sum_{i=1}^{N}\sum_{m=1}^{M}\sum_{t=1}^{T_i} \log P(\tilde{y}_t^{i,m} \mid \tilde{y}_{<t}^{i,m}, x^i; \theta_S). \tag{3}$$

Fig. 1 provides a graphical representation of this method. The objective of MHD is to achieve benefits similar to word-level KD and GKD regarding the exploration of the teacher's probability distribution, while restricting this exploration to a relative high-probability space to avoid degradation from poor distributions. Furthermore, by generating different target-side prefixes, MHD offers advantages similar to on-policy training without requiring the student to generate sequences during training, thus accelerating the training and reducing memory overhead as the teacher is not required online. This exploration of the teacher's probability space follows a similar spirit to MBR. However, while MBR requires generating many hypotheses to identify a single best output for inference, MHD leverages this exploration to increase the amount of training data and expose the student to multiple alternative translations for the same source sentence.[4]

4 Experimental Settings

This section describes the technical details of the experiments carried out. It starts with the decoding methods to be analysed, moves on to the language pairs, the models and the corpora used, and finally explains the features to be evaluated and how to measure each of them.

4.1 Decoding Methods

This study focuses on a selected set representing both deterministic and stochastic methods: beam search and diverse beam search as deterministic methods and top-p (also known as nucleus sampling), top-k and MBR decoding as stochastic methods.

- Beam search (BS): At each decoding step, beam search keeps the n highest probability paths (Graves 2012). This has the advantage of identifying high probability sequences that start with less likely initial tokens and would have been ignored by greedy decoding, which always chooses the most probable token. When generating data for KD, we used n=10.
- Diverse beam search (DBS): It is a variant of beam search that tries to produce more diverse results. Instead of maintaining a single list with the most likely paths, it divides the n paths into G groups and applies a penalty (λ) to prevent them from being similar to each other (Vijayakumar et al. 2018). In our experiments, as recommended by the authors of this method, we used n=G, i.e. as many groups as n, with only one sequence per group, and λ=0.5. As in the case of beam search, we used n=10.
- Top-k: The probability mass is redistributed among the k most likely tokens and the output token is then sampled for the resulting distribution (Fan, Lewis, et al. 2018). This process is repeated until the end-of-sentence token is selected.[5] For our experiments, we kept the original proposal of k=10 (Fan, Lewis, et al. 2018), which has proven to work well for generating synthetic corpora for back-translation (Y. Zhang et al. 2020).
- Top-p: The probability mass is redistributed among the smallest possible set of tokens whose cumulative probability exceeds the probability p (Holtzman et al. 2020). This way, the amount of the set of candidate tokens can dynamically increase or decrease according to the next token's probability distribution. The sequence is generated using the same iterative process as with top-k. Following Eikema and Aziz (2022), we set p to 0.7 in our experiments.
- Minimum Bayes' Risk decoding: MBR chooses the hypothesis that has the lowest risk of being incorrect evaluating its proximity to other candidate sequences, based on a distance measure. The method to calculate this distance is called *utility function* (Kumar and Byrne 2004). Following Finkelstein and Freitag (2024), we

[4]While word-level KD uses prefixes from a ground-truth translation and learns the teacher's distribution at each step, MHD exposes the student model to multiple complete sentences, allowing it to observe a much larger variety of prefixes for the same set of source sentences.
[5]The variability of the output depends on the value of k. Using the vocabulary size as k corresponds to ancestral sampling, while k=1 works as greedy decoding.

used epsilon sampling (Hewitt et al. 2022) to create 256 candidates with ϵ=0.02. We use chrF[6] (Vamvas and Sennrich 2024) as the utility function.

Generating multiple translations with each of these decoding strategies allows us to explore different regions of the teacher's probability distribution. Deterministic methods offer limited exploration but remain focused on high-probability regions. In contrast, sampling-based methods, and MBR in particular, enable broader exploration, albeit including regions with lower probability. This inherent trade-off between exploration depth and probability density is characteristic of each decoding approach; investigating how this balance affects MHD is one of the core objectives of this paper.

4.2 Models, Language Pairs and Data

Models. In order to evaluate the generalisation of MHD across different architectures and amount of network parameters, we used two configurations of teacher and student models.

First, we used NLLB-200 1.3B and NLLB-200 3.3B (NLLB Team et al. 2022) as encoder-decoder teacher models. Our primary students are encoder-decoder transformer models in the *base* configuration, as defined by Vaswani et al. (2017, Tbl. 3). With 65M parameters, our student models are notably compact, representing just 5% of the size of the NLLB-200 1.3B model and 2% of the NLLB-200 3.3B model.

Secondly, we employed the multilingual LLM EMMA-500 8B (Ji et al. 2025) as a teacher. This enabled us to explore two KD paths: (1) KD from a large decoder-only model into a compact 65M encoder-decoder students, and (2) KD into a pre-trained decoder-only student model, Llama-3.2 1B (Grattafiori et al. 2024). The latter configuration is particularly relevant for assessing our MHD approach in a transfer learning setup where the student already possesses prior linguistic knowledge. For more details on the architecture and training, see Appendix B.

Language Pairs. We selected the language pairs to be distilled from the languages that the teacher models support, based on two variables: The teacher model translation quality (Table 1) and the size of the available corpora. Our objective is to have multiple scenarios that allow us to analyse the impact of these variables at both generation and training time. The languages we have chosen are English (eng), Swahili (swh), Igbo (ibo) and Bambara (bam). Language pairs to be distilled and reasons for this selection are as follows:

- **From English:** eng-swh, eng-ibo, eng-bam. Almost unlimited monolingual source corpora and target languages with varying translation quality.
- **Into English:** swh-eng, ibo-eng, bam-eng. Limited amount of monolingual source corpora, although in some cases sufficient to experiment with different dataset sizes. As the teacher model has been exposed to a large amount of English during its training, and BS limits the vocabulary we can extract, we hypothesise that sampling methods allow us to extract more knowledge from the teacher model.
- **Zero-shot:** bam-swh. Small amount of monolingual source corpora and low translation quality. The teacher's knowledge is based on transfer learning from other translation directions and monolingual knowledge of the source and target languages.

Data. Regarding the availability of data, English and Swahili have the largest corpora, from which we selected a subset of 1 million monolingual sentences. For Igbo, we used a corpus comprising 451,789 sentences, while for Bambara we employed a corpus containing 108,187 sentences. All corpora used are freely available. Table 2 shows the origin of each corpus. Specific details about the corpora and their pre-processing can be found in Appendix C. As development and test sets we use the FLORES+[7] (NLLB Team et al. 2022) *dev* and *devtest* splits, respectively.

[6] fastChrF signature: numchars.6+beta.2.0+space.true

[7] https://huggingface.co/datasets/openlanguagedata/flores_plus

Table 1. ChrF++ scores of NLLB-200 1.3B and EMMA-500 8B on the FLORES+ devtest dataset for different decoding methods: beam search (BS), diverse beam search (DBS), top-p (average of 3 runs), top-k (average of 3 runs), and MBR. The results with BLEU are showed in Appendix D.1. The NLLB-200 3.3B results, which show the same relative order between decoding methods, can be found in Appendix D.2.

Teacher model	Method	eng-swh	eng-ibo	eng-bam	swh-eng	ibo-eng	bam-eng	bam-swh
NLLB-200 1.3B	BS	59.2	41.0	30.9	63.5	52.6	38.6	35.6
	DBS	58.5	39.0	28.6	63.0	51.7	37.6	32.6
	top-p	51.3	36.3	27.0	57.0	46.7	35.5	32.6
	top-k	49.1	34.4	27.2	52.3	44.3	34.7	32.5
	MBR	58.9	41.8	31.7	63.8	53.0	38.7	36.6
EMMA-500 8B	BS	59.8	42.8	18.0	65.3	49.3	32.1	–
	top-p	58.2	40.6	19.1	62.0	46.0	30.6	–

Table 2. Monolingual corpora used.

Language	Corpus	Sentences
English	OSCAR-3301	1,000,000
Swahili	Monolingual African Languages from ParaCrawls	1,000,000
Igbo	Monolingual African Languages from ParaCrawls	451,789
Bambara	bayelemabaga, lafand-mt, Leipzig, NLLB-Seed, xP3, MADLAD-400	108,187

4.3 Evaluation Metrics

We evaluate two key elements: the synthetic corpora and the models trained on them.

Synthetic Corpora. We assess vocabulary diversity, hipothesis variability, and translation performance of the teacher model used for its generation.

- Lexical richness: measured through Zipf's Law and by counting unique words and sentences.
- Variability among the M translations generated for each source source: evaluated using self-BLEU (Y. Zhu et al. 2018), where lower values indicate greater diversity in translations from the same source sentence.
- Translation performance of the teacher model: we evaluate translation performance on the FLORES+ dataset, generating a single translation per source. We use chrF++ (Popović 2017) to estimate the translation quality of the synthetic corpora generated by the teacher model for each language pair. The original NLLB paper reports that chrF++ scores for this model tend to be systematically higher when generating English compared to other target languages, and that these scores correlate well with human evaluations (NLLB Team et al. 2022). This observation supports the use of chrF++ as a reliable proxy for synthetic corpus quality.

All synthetic corpora are generated by translating the corresponding monolingual corpus (see Table 2) with the teacher model, using the Transformers library (Wolf et al. 2020) and the selected decoding method.

Student Models. We assess student performance based on four criteria:

- Translation performance: evaluated in the same manner as the teacher's output, using beam search (n=5) on the test set. Although recent neural evaluation metrics such as AfriCOMET (J. Wang, D. I. Adelani, et al. 2024) and SSA-COMET (S. Li et al. 2025) support some of the languages considered in this work, none of these metrics have been trained specifically to cover the full set of language pairs under study. Given this

limitation, we adopt chrF++[8] (Popović 2017) as our primary evaluation metric. To validate our findings further and provide a more comprehensive assessment, we also report BLEU[9] and, for language pairs that are supported (eng-swh and swh-eng), COMET (Rei et al. 2020) scores. We observe a consistent ranking of systems in most scenarios, indicating that the reported trends are stable and not tied to a specific evaluation method. We assessed statistical significance using paired approximate randomization (Riezler and Maxwell 2005) to compare the student models trained with different corpora $\mathcal{D}_Z^M$.

- Gender bias: measured through contrastive conditioning (Vamvas and Sennrich 2021), as detailed in Sec. 5.4, due to the lack of annotated datasets for the languages of this study.
- Hallucinations: assessed by calculating the cosine similarity between the sentence embeddings (Dale et al. 2023) (see Sec. 5.5) of the generated translations and the reference.

5 Experiments and Results

Our experiments are designed to evaluate the effectiveness of MHD and to detect the factors influencing student model performance. First, in Section 5.1, we evaluate MHD by training encoder-decoder student models from scratch using small source corpora and varying the number of translation hypotheses per sentence (M), considering both encoder-decoder and LLM architectures as teacher models. The aim is to understand how exposing the student model to multiple translations and target-side prefixes affects learning. In this setting, our baseline is the student model distilled with word-level KD using the synthetic corpus $\mathcal{D}_{\text{BS}}^1$ in place of real parallel data, allowing us to compare the effect of directly transferring the teacher's token-level distribution against our sequence-level approach.

Once the best-performing value of M is identified, we fix it and study the impact of increasing the size of the monolingual source corpus (Section 5.2). This allows us to assess how the amount of data influences the diversity of vocabulary and sentence structures available for knowledge extraction from the teacher. These experiments are conducted only for language pairs for which we have access to larger monolingual corpora. Next, in Section 5.3, we explore the trade-off between diversity and translation quality introduced by the decoding parameters p (in top-p) and k (in top-k), which control how far the sampled outputs diverge from the teacher model's most likely predictions. We then conduct a gender bias analysis (Section 5.4) and, in Section 5.5, we examine the tendency to hallucinate of our student models. Finally, in Section 5.6, we distil the LLM EMMA-500 8B into a pre-trained LLM and compare MHD with several state-of-the-art KD strategies for LLMs. Section 5.7 analyses the efficiency of these techniques and MHD in terms of training time and hardware requirements.

5.1 Impact of the Number of Translation Hypotheses and Decoding Methods

Sampling methods typically yield lower performance for MT compared to BS and DBS as shown in Table 1. Nevertheless, they are widely used in open-ended generation tasks because of their ability to produce diverse and more human-like outputs (Holtzman et al. 2020) than deterministic methods. In this section, we investigate whether, despite the drop in translation quality, sampling methods can provide more effective training data for student models.

We generated our $\mathcal{D}_Z^M$ datasets by translating 100k sentences (as this is the size of our smallest corpus) using Z={BS, DBS, top-p, top-k, MBR} and $M = \{1, 3, 5, 10\}$. As already explained in Section 4.1, while top-p and top-k rely on sampling, BS and DBS selected the M highest-probability candidates. For MBR, we selected the best M translations. Note that $\mathcal{D}_{\text{BS}}^1$ corresponds to the standard sequence-level KD. Subsequently, we trained student models on the training datasets generated with each decoding method.

[8] chrF++: "nrefs:1|case:mixed|eff:yes|nc:6|nw:2|space:no|
[9] BLEU: "nrefs:1|case:mixed|eff:no|tok:13a|smooth:exp|

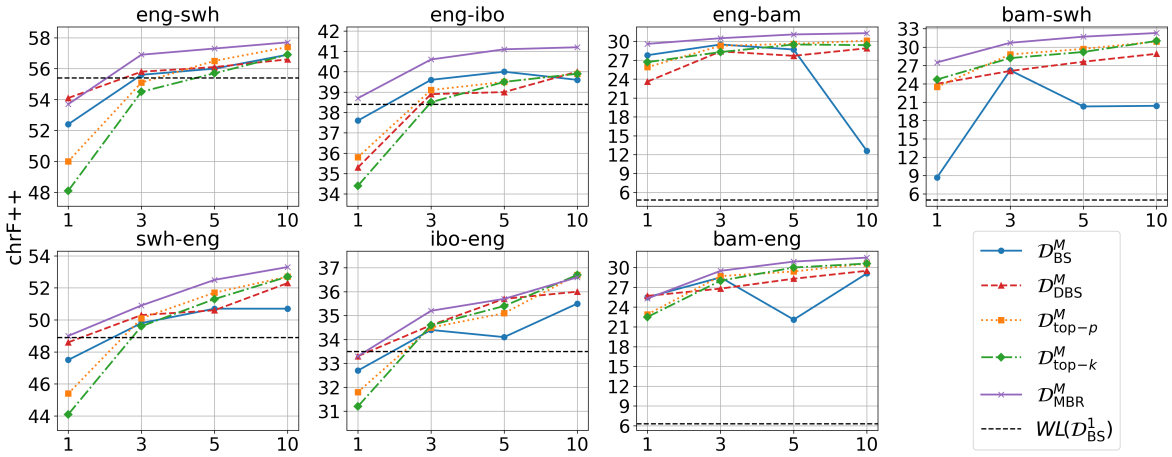

Fig. 2. Average chrF++ score obtained by encoder-decoder student models trained from scratch on M samples generated by NLLB-200 1.3B with different decoding methods and varying number of translations per source sentence (x-axis).

Results. Fig. 2 shows the performance of student models distilled from NLLB-200 1.3B, with scores reflecting the average chrF++ on three training runs.[10] Results with NLLB-200 3.3B, which shows the same relative order between decoding methods, are provided in Appendix D.3.

To examine the differences between MHD and standard sequence-level KD, we first computed the statistical significance of the difference between the student models trained on $\mathcal{D}^{10}_{Z}$ (with Z={BS, DBS, top-p, top-k, MBR}) with the student model trained on $\mathcal{D}^{1}_{\text{BS}}$. Then, to assess the impact of different decoding methods, we compared via statistical significance the models trained on each $\mathcal{D}^{10}_{Z}$ to those trained on $\mathcal{D}^{10}_{\text{BS}}$. Except for the eng-ibo models, all language pairs showed statistically significant differences compared to $\mathcal{D}^{1}_{\text{BS}}$. In contrast, when considering $\mathcal{D}^{10}_{\text{BS}}$ as the baseline, models trained with $\mathcal{D}^{10}_{\text{DBS}}$ for ibo-eng, bam-eng, and eng-swh did not show statistically-significant differences. Models trained on datasets generated by sampling methods showed statistically-significant differences for eng-bam, swh-eng, ibo-eng, bam-eng and bam-swh. The complete results of the statistical significance tests are shown in Appendix D.4, Table 10.

Word-level KD (WL($\mathcal{D}^{1}_{\text{BS}}$) in Figure 2) yields superior results compared to sequence-level KD with a single sample for several language pairs. This suggests that, when the teacher model is fitted well (Eikema and Aziz 2020), its probability distribution contains valuable information for training student models. However, if the teacher model is poorly fitted to a specific language, its probability distribution can exhibit high entropy, proving detrimental to the student model. This phenomenon is observed in language pairs involving Bambara. MHD allows for the extraction of more information regarding the teacher's probability distribution than traditional sequence-level methods. Crucially, due to the specific decoding methods employed, this distribution is constrained to a subset of tokens rather than the entire vocabulary, thereby mitigating issues associated with highly dispersed probability mass. This benefit, coupled with the generation of multiple prefixes, enables MHD to outperform word-level KD and standard sequence-level KD across all investigated language pairs.

[10]Note that, for the sampling methods, translations were sampled again from the teacher's distribution in each training run.

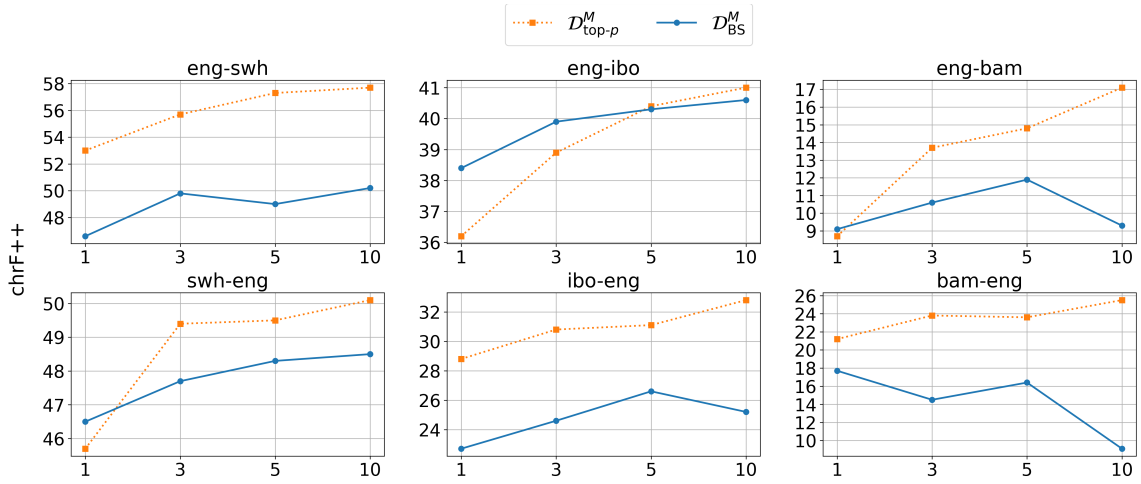

Fig. 3. ChrF++ score obtained by encoder-decoder student models trained from scratch on M samples generated by EMMA-500 8B with different decoding methods and varying number of translations per source sentence (x-axis).

As expected, student models trained on deterministic methods or MBR outputs generally performed better than top-p and top-k when only one translation per sentence was generated ($M = 1$). However, as the number of translations per sentence increased, models trained on sampled data outperformed those trained with deterministic methods.

The gap between BS and sampling methods is especially notable for bam-swh. Interestingly, students trained with $\mathcal{D}^5_{\text{BS}}$ and $\mathcal{D}^{10}_{\text{BS}}$ performed worse than those trained with $\mathcal{D}^3_{\text{BS}}$. Eikema and Aziz (2020)'s observations on the inadequacy of the mode show that, when the model is poorly fitted, the most probable translation is not the best. This can explain why traditional KD with BS ($\mathcal{D}^1_{\text{BS}}$) fails, but $\mathcal{D}^3_{\text{BS}}$ contains translations from which the student model is able to learn. The results with $\mathcal{D}^{M>3}_{\text{BS}}$ are discussed further on, together with the analysis of the generated corpora.

$\mathcal{D}^M_{\text{MBR}}$ results must be analysed carefully, as metric bias may have been introduced by using the same type of metric to rank the MBR candidates and for evaluation (Kovacs et al. 2024). $\mathcal{D}^M_{\text{MBR}}$ achieved the best results when evaluated with chrF++ but is not the most effective method when evaluated using BLEU (Appendix D.3, Fig. 15). Nevertheless, both metrics show that $\mathcal{D}^M_{\text{MBR}}$ achieved the best results for all tested values of M in language pairs involving Bambara. We hypothesise that it is for these language pairs (bam-eng, eng-bam, bam-swh) that the mode is less suitable. This is where extracting more information from the probability distribution is most beneficial; MBR helps to extract that information while filtering out possible mistranslations.

In general, with $M = 10$, the greatest advantage of sampling over deterministic methods occurs when the target language is English. This is in line with our hypothesis that, as the teacher has been trained with a vast amount of English and we are working with small corpora, the sampling methods allow us to extract more information than BS. To further explore the limits of increasing M, we conducted experiments using top-p sampling with $M = 100$. Overall, the results show a consistent improvement over $M = 10$. In several cases, these results even outperform those achieved using a larger monolingual corpus of 1M sentences and $\mathcal{D}^1_{\text{BS}}$ (as discussed in Section 5.2), demonstrating the potential of increasing M through sampling methods to better capture the teacher's distribution. These results are provided in Appendix D.5, Table 12.

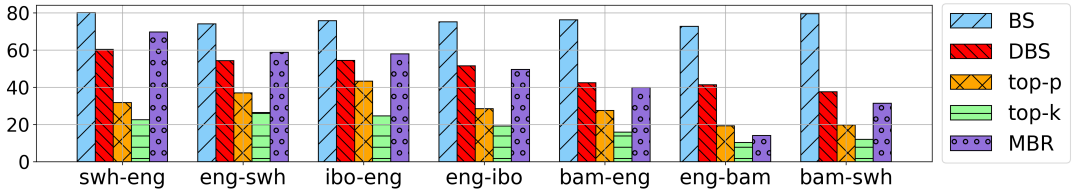

Fig. 4. Similarity among 10 generated translations per source sentence as evaluated by self-BLEU (y-axis).

To ensure that the improvements seen with sampling methods were not simply due to a particularly good translation among the multiple outputs, we conducted an additional experiment. For the eng-swh $\mathcal{D}^{10}_{\text{top-}p}$ corpus, we selected the best translation for each source sentence based on COMET without reference. We then used only these selected translations to train a student model. The resulting performance was similar to that of a model trained with $\mathcal{D}^{1}_{\text{top-}p}$, confirming that the improvements observed with sampling methods were driven by the diversity of multiple translations.

Finally, Fig. 3 illustrates the results of applying MHD to the distillation of an LLM. For this setup, we focused on BS, as it is the default decoding method for sequence-level KD; and top-p sampling, which provided the best balance between performance and computational cost in our previous experiments. It is worth noting that, while LLMs are typically associated with sampling-based generation, EMMA-500 achieves higher translation quality using BS than top-p (see Table 1). Our findings follow the same pattern observed in the NLLB-200 KD experiments: performance tends to improve as the number of generated translations increases.[11] However, BS exhibits a more pronounced degradation in language pairs where EMMA-500 performs poorly compared to NLLB (bam-eng, eng-bam, ibo-eng). This suggests that if the model is not well fitted for a specific language pair, BS fails to generate a sufficiently diverse set of correct translations for MHD.

Analysis of Generated Corpora. To explain the above results, we analyse the properties of each decoding method, as well as the synthetic corpora generated by NLLB-200 1.3B.

The variability of the generated translations indicates the amount of information extracted. Fig, 4 shows the self-BLEU (Y. Zhu et al. 2018) score of 10 translations per source generated using each decoding method. As self-BLEU measures the similarity between translations, a high score indicates low variability. As expected, deterministic methods, which focus on selecting the most likely translations, result in low variability, even when using DBS. In contrast, sampling methods, especially top-k, produce more diverse translations. This variability suggests that the translations generated using sampling methods have a greater vocabulary and are richer in terms of lexicon. This explains why they provide better training data for the student models. To support this claim, we use the Zipf distribution (Holtzman et al. 2020) of each generated corpus. Fig. 5 compares the Zipf distribution of the corpora generated by translating 100k sentences from English into Swahili, together with the distribution of a native Swahili corpus of 1M sentences (1M $\mathcal{D}$ in the plot). The figure also includes the distribution of a corpus generated by translating the English corpus of 1M sentences with BS and M=1. The analysis shows that the corpora generated with sampling and M=10 from smaller texts are more similar to native corpora than those produced with BS with a single translation from larger texts. Intuitively, the generation of multiple translations with BS might result in either very similar sentences, adding little value, or hallucinations when the model is forced into less probable paths.

[11] As in the experiments presented in Fig. 2, we computed the statistical significance of the difference between the models trained on $\mathcal{D}^{10}_{Z}$ (Z={BS, top-p}) and those trained on $\mathcal{D}^{1}_{\text{BS}}$. We also compared $\mathcal{D}^{10}_{\text{top-}p}$ with $\mathcal{D}^{10}_{\text{BS}}$. All the compared models presented statistically significant differences, with the exception of the comparison between $\mathcal{D}^{10}_{\text{top-}p}$ and $\mathcal{D}^{10}_{\text{BS}}$ for eng-ibo.

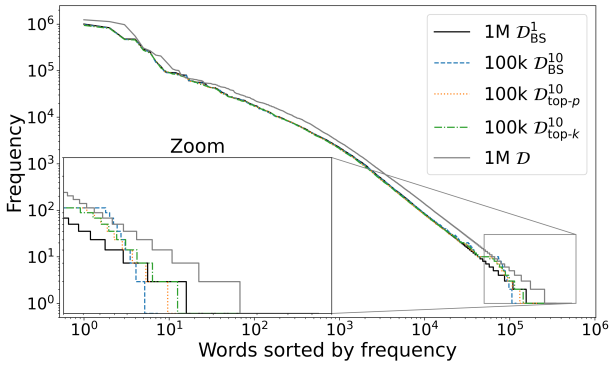

Fig. 5. Zipf's distribution over Swahili corpora. Similar patterns were observed for the rest of languages.

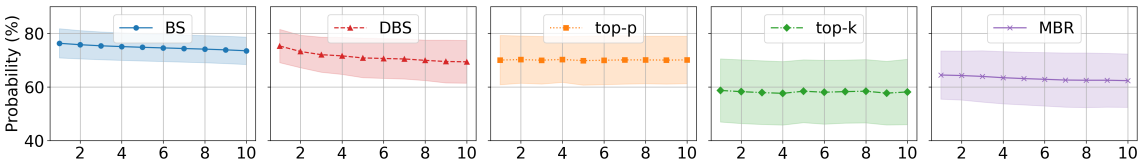

Fig. 6. Probabilities normalised by length of 10 swh-eng translation hypotheses generated with NLLB-200 1.3B for each source sentence in the FLORES+ devtest set. The shaded areas around each line represent the standard deviation.

The idea that unlikely paths lead to poorer translations is related to how each decoding method generates its translations. While sampling methods generate translations independently, deterministic methods and MBR perform a ranking of the generated hypotheses. As seen in Fig. 6, this results in the probabilities of top-p and top-k translations remaining stable, while those of deterministic methods and MBR decay. This may explain the decrease in quality observed in students trained with $\mathcal{D}^{M>1}_{\text{BS}}$ when working with languages for which the teacher is poorly fitted. If the next-token probability mass is highly concentrated in a few tokens and we require more translations, deterministic methods have to take improbable paths.

To further investigate this phenomenon, we conducted an additional experiment to approximate the expected quality of the teacher distribution. Using the 256 translations per source sentence generated with epsilon sampling, we computed the median probability of these translations as a proxy for the expected translation likelihood. Then, we filtered them by removing those with a probability lower than the previously obtained median. This allowed us to identify which beam outputs were statistically unlikely under a broader sampling of the teacher distribution. The results, illustrated in Fig. 7, show that the discarded translations were predominantly associated with short source sentences and language pairs where the teacher model performed poorly. This supports the hypothesis that deterministic methods, when forced to produce multiple outputs, are more likely to select low-probability and potentially low-quality continuations in such settings.

In contrast to deterministic methods, sampling methods can produce repeated sentences, especially top-p due to its dynamically adjusted window size. While this can limit diversity, it can also help to prevent hallucinations that could negatively affect the training of student models. This effect is visible in Fig. 4, where top-p sampling produces lower variability than top-k. The performance of MBR depends on the probability distribution of the

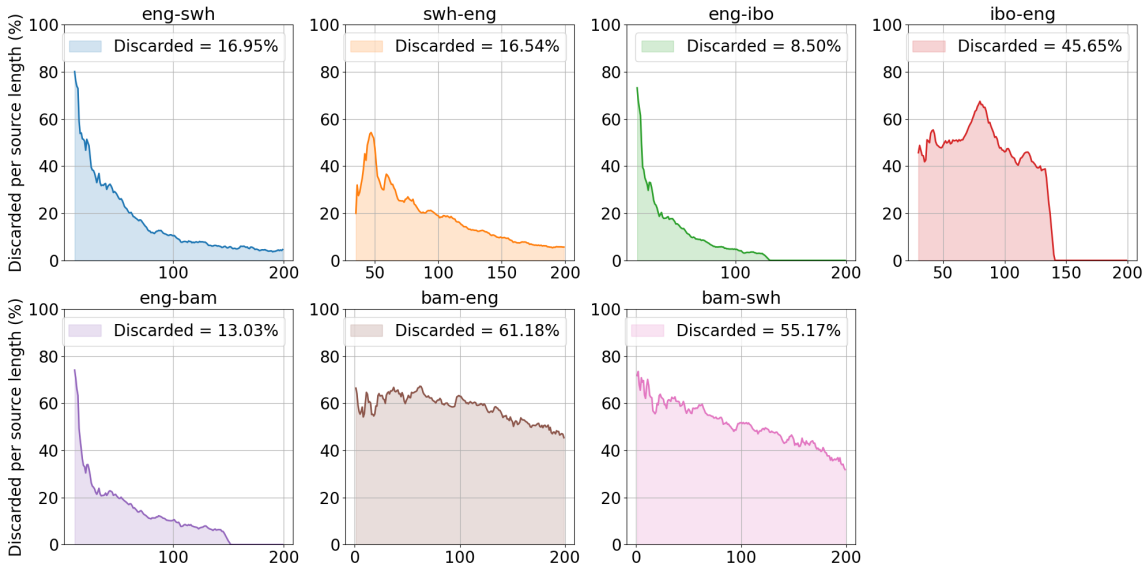

Fig. 7. Percentage of translations (y-axis) for the same source with a probability lower than the median of the 256 translations generated using epsilon sampling for the same source. Source sentences are ordered by their length in characters (limited to 200) along the x-axis. The shaded area represents the total percentage of translations that were discarded.

teacher model. When the probability mass is highly concentrated in a few tokens, MBR produces low variability (Fig. 4, swh-eng). Conversely, when the teacher's probability distribution is more spread out, MBR introduces greater diversity in its selections (Fig. 4, eng-bam).

5.2 Impact of Source Corpus Size

The size of the source corpus plays a crucial role in KD, as a larger corpus contains more vocabulary and allows for more knowledge to be extracted from the teacher. To analyse how this affects MHD, we translated 100k, 500k and 1 million sentences with NLLB-200 1.3B and M=10. We used $\mathcal{D}^1_{\text{BS}}$ obtained from the same corpus as a baseline for each size.

Results. Fig. 8 shows the performance of student models trained on each dataset. As observed, the discrepancy between different decoding methods decreases as the corpus size increases. For corpora of 500k sentences, sampling methods still outperform BS, while for corpora of 1 million sentences, sampling methods do not consistently yield superior results. However, MHD remains advantageous over $\mathcal{D}^1_{\text{BS}}$.

Analysis of Generated Corpora. To explore the importance of lexical richness in the translated corpus, we compared the number of unique words in both the source corpus and the generated corpus. Fig. 9 illustrates the relationship between the size of the source vocabulary, the vocabulary produced by each decoding method, and the chrF++ scores obtained by training student models on these corpora. The results show that sampling methods act as vocabulary amplifiers by generating multiple translations. However, it is important to know which part of this vocabulary is useful to the model. Fig. 10 shows the percentage of the devtest target vocabulary present in the training corpus. It can be seen that until a certain coverage is reached (about 87% for eng-swh and 95% for

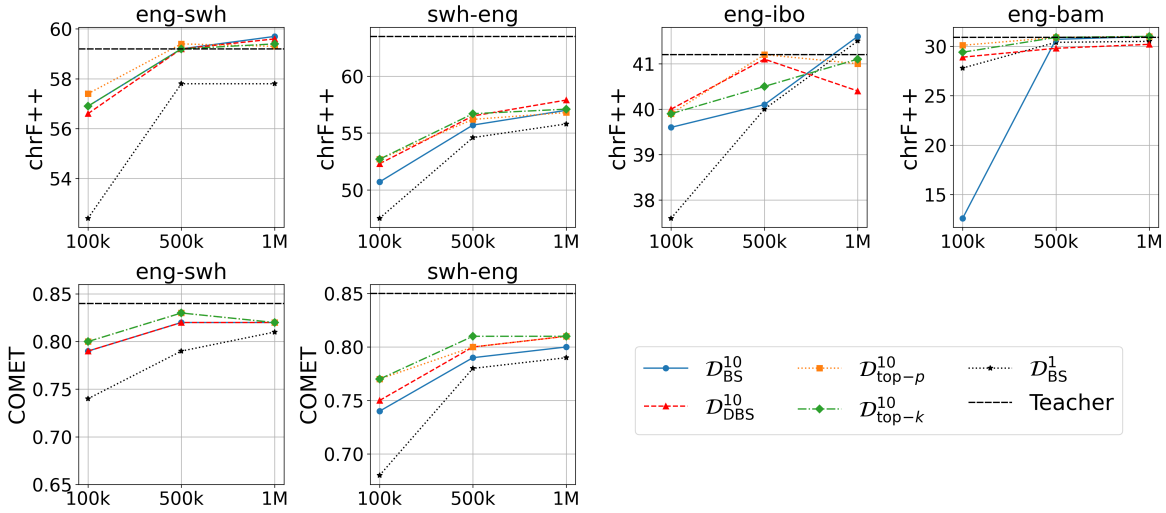

Fig. 8. Scores attained for different corpus sizes in number of sentences (x-axis). $\mathcal{D}^{1}_{\text{BS}}$ corresponds to the standard sequence-level KD. The results of $\mathcal{D}^{10}_{\text{top-}p}$ and $\mathcal{D}^{10}_{\text{top-}k}$ overlap in the COMET eng-swh graph, as well as $\mathcal{D}^{10}_{\text{BS}}$ and $\mathcal{D}^{10}_{\text{DBS}}$

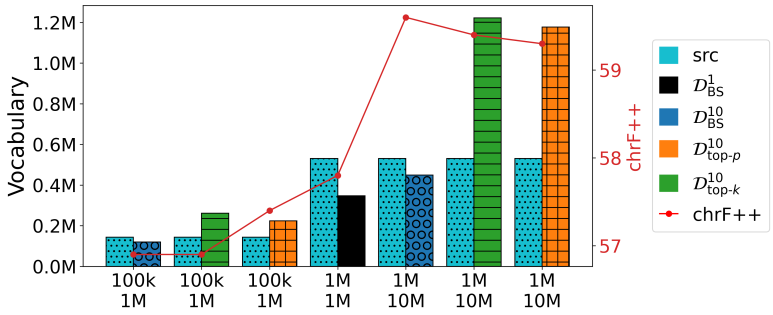

Fig. 9. Relationship between the vocabulary size of the swh-eng corpus and the chrF++ of the student models. X-axis markers indicate amount of sentences in the source corpus (first row) and sentences in the generated corpus (second row).

swh-eng), increasing the coverage produces better student models, even if the teacher translations are worse. On the other hand, once this threshold is exceeded, it is more beneficial to prioritise translation quality.

In addition to translation quality, BS can offer another benefit for KD. During training, models typically use teacher forcing, where the correct token is used as input, leading to a mismatch between training and inference. During inference, the model must rely on self-generated tokens, typically obtained with BS. This *exposure bias* (Ranzato et al. 2015) can be mitigated by training the student models with BS outputs, which are closer to the tokens generated during inference. If the source corpus is sufficiently large, BS can extract enough vocabulary, and its similarity to the inference process benefits the student model. This also explains the performance of the swh-eng model trained on a 1M source corpus using DBS (Fig. 8), which keeps the inference similarity of BS while providing greater diversity.

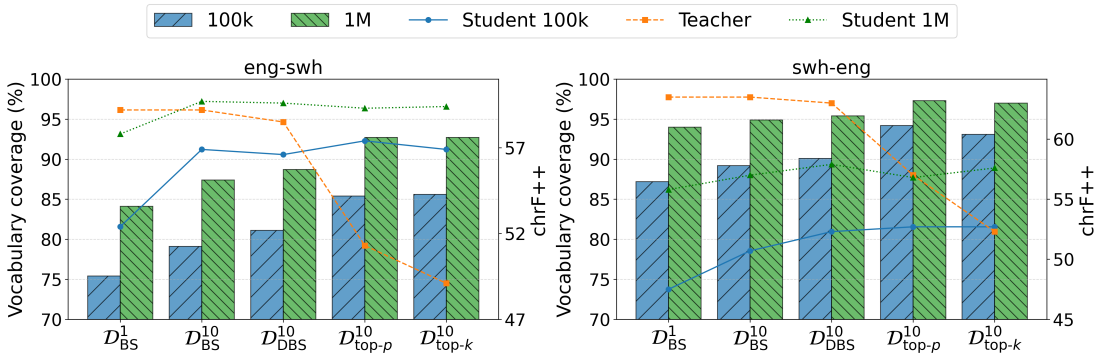

Fig. 10. Effect of vocabulary coverage and teacher translation quality. X-axis shows decoding methods ranked by variability. Columns show the percentage of devtest vocabulary (left Y-axis) present in the training corpus. Lines show the chrF++ (right Y-axis) of the models trained with each corpus and the chrF++ of the teacher generating only one translation per source sentence with the decoding method used to generate each dataset.

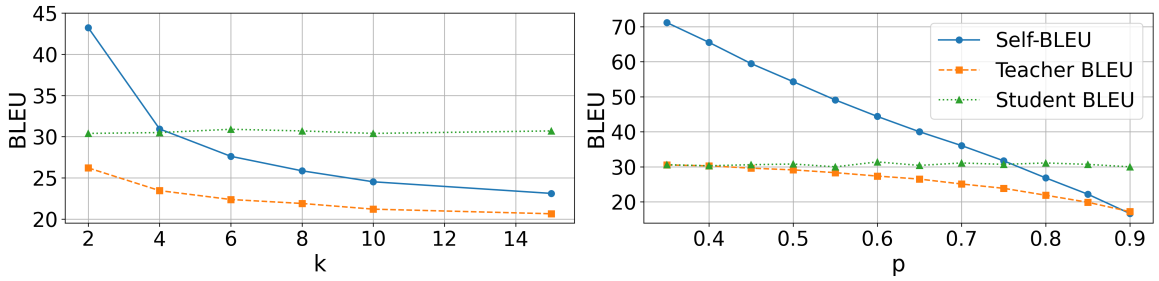

Fig. 11. Relationship between the teacher translation quality and variability and the student models score for eng-swh. Initial corpus: 100k sentences.

5.3 Divergence from the Mode vs. Translation Quality

The adjustment of the sampling parameters affects both output variability and translation quality, making the output more similar to greedy decoding or ancestral sampling. To gauge the sensitivity of MHD to the values of p and k, we conducted experiments on eng-swh, translating 100k eng sentences with M=10. Fig. 11 illustrates the impact of p and k values on both the translation performance of the student and teacher models (measured by BLEU), and the similarity of the translations (measured by self-BLEU). We use BLEU instead of chrF++ in this graph for a better comparison with self-BLEU. As observed, higher values of p and k result in more diverse translations, albeit with poorer teacher performance, while maintaining similar performance for the student models. Finally, we repeated the experiment with 1 million sentences and found that the results were consistent with our previous findings using a smaller corpus. These results suggest that the trade-off between quality and variability is independent of corpus size when using the same decoding method.

5.4 Gender Bias Analysis

Sequence-level KD typically amplifies biases present in the teacher model due to the over-representation of frequent tokens (Ahn et al. 2022). To assess whether MHD can mitigate this issue, we employed contrastive

Table 3. Contrastive conditioning for gender bias detection. The output of an unbiased model from the original source should match the output of the evaluator model from the correct disambiguation cue.

	Example
Original Source Sentence	The CEO bought a car because **she** is rich.
Correct Disambiguation Cue	The **[female]** CEO bought a car because **she** is rich.
Incorrect Disambiguation Cue	The **[male]** CEO bought a car because **she** is rich.
Correct Spanish Translation	**La Directora General** se compró un coche porque es **rica**.
Incorrect Spanish Translation	**El Director General** se compró un coche porque es **rico**.

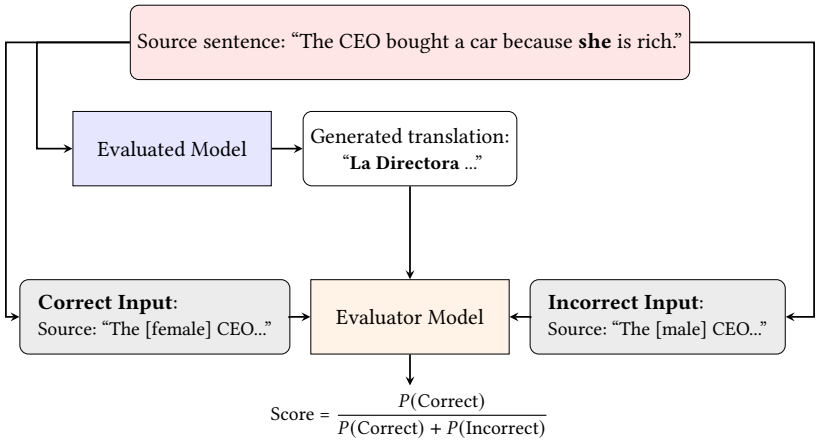

Fig. 12. Scheme of contrastive conditioning. The evaluator model calculates the probability of the generated translation for both correct and incorrect disambiguated sources. A translation aligned with the correct source will produce a score close to 1, and a translation aligned with the incorrect source will produce a score close to 0.

conditioning (Vamvas and Sennrich 2021) to evaluate gender bias in the student models from NLLB-200 1.3B, using NLLB-200 1.3B as the evaluator model and the WinoMT dataset (Stanovsky et al. 2019).

Contrastive conditioning is a method that leverages the probability assigned by a translation model to a generated translation when presented with controlled variations of the input. Specifically, it involves generating a translation from the original source sentence with the evaluated model and then calculating the probability of the generated translation with the evaluator model when the input is a disambiguated version of the source. If the evaluated model is unbiased, the probability assigned to the translation should be higher when the disambiguated input aligns with the correct gender. Table 3 shows an example of the disambiguated input and the expected output in Spanish, and Fig. 12 illustrates the evaluation protocol.

Among the languages in this study, WinoMT only contains a dataset for English, so we can only evaluate models translating from English into Swahili, Igbo and Bambara. For each language pair, we evaluated the student models trained with $\mathcal{D}^1_{\text{BS}}$ and $\mathcal{D}^{10}_Z$ ($Z \in \{\text{BS}, \text{DBS}, \text{top-}p, \text{top-}k, \text{MBR}\}$). The results in Table 4 show that generating multiple translations for training reduces gender bias compared to training with a single translation. Although all methods show improvement, sampling-based methods achieve greater bias mitigation by avoiding the over-representation of the most likely tokens inherent in BS.

Table 4. Contrastive conditioning accuracy over WinoMT dataset evaluating gender bias. Higher scores are better and the bold scores mark the best student models.

	NLLB-1.3B	$\mathcal{D}^{1}_{\text{BS}}$	$\mathcal{D}^{10}_{\text{BS}}$	$\mathcal{D}^{10}_{\text{DBS}}$	$\mathcal{D}^{10}_{\text{top-}p}$	$\text{D}^{10}_{\text{top-}k}$	$\text{D}^{10}_{\text{MBR}}$
eng-swh	52.9	49.2	51.0	50.4	**51.7**	**51.7**	50.1
eng-ibo	52.7	49.4	50.2	50.4	**50.5**	**50.5**	49.4
eng-bam	58.3	50.8	50.3	51.5	52.3	51.3	**52.5**

5.5 Hallucinations

Hallucination is a well known but atypical issue in MT (Guerreiro et al. 2023) where the model generates an output that, despite being fluent, is partially or entirely unrelated to the source (Dale et al. 2023). Hallucinations can significantly undermine users' confidence in translation models when they occur. Incorporating alternative translations and increasing the variability of the training corpus may improve the student model's fluency in the target language, but not necessarily its translation adequacy. To evaluate this, we analysed the occurrence of hallucinations in the student models from NLLB-200 1.3B.

Several studies (Dale et al. 2023; Xu et al. 2023) have shown that using cross-lingual sentence embeddings to measure the similarity between the model output and the reference yields better hallucination detection than metrics such as COMET (Guerreiro et al. 2023). Therefore, we computed sentence-level SONAR (Duquenne et al. 2023) embeddings for the system outputs and the references, and then computed cosine similarities between them. To find the values of cosine similarity of SONAR embeddings representing hallucinations, we shuffled the references and computed the cosine similarities between the shuffled and original references. Fig. 13 displays the kernel density estimation of the cosine similarity distributions for the systems. The shaded area represents the shuffled references, which cluster near zero, where hallucinations are expected to appear (Dale et al. 2023).

In most language pairs, models trained with MHD exhibit fewer hallucinations than those trained with traditional sequence-level KD. The exception is eng-bam, where the teacher model performs particularly poor, leading to more hallucinations in student models trained with $\mathcal{D}^{10}_{\text{BS}}$ compared to $\mathcal{D}^{1}_{\text{BS}}$. Fewer hallucinations occur in models producing low-resource languages when trained with sampling-based translations than for models trained with deterministic translations, while differences are minimal for models generating English. This aligns with our observations in Section 5.1 regarding the decline in quality of deterministic methods when generating multiple translations in languages where the teacher is poorly fitted. We hypothesise that this reduction in hallucinations stems from the way MHD trains the student model: by exposing it to multiple target prefixes for the same source sentence, the student learns to recover more effectively from an erroneous prediction. This makes the model less likely to let a single mistaken token cascade into a hallucination, as it has seen a wider range of valid continuations from which to resume generation.

5.6 Distilling into a Pre-Trained Large Language Model

While previous sections focused on training encoder-decoder student models from scratch, this section explores the application of MHD to a different paradigm: distilling a large decoder-only teacher model into a small and pre-trained decoder-only student model. Several state-of-the-art KD techniques for LLMs assume that the student already possesses a baseline capability for the target task. For instance, GKD (Agarwal et al. 2024) utilises outputs generated by the student itself during training, and MT-PATCHER (J. Li et al. 2024) focuses on modifying specific translations where the student fails, rather than training on the full set of teacher outputs.

To evaluate the impact of MHD on the distillation of an LLM into a decoder-only pre-trained student model, we distilled EMMA-500 8B into Llama-3.2 1B. EMMA 500 is based on an earlier version of Llama, so both models have a large amount of shared vocabulary. We compared MHD against standard KD approaches, including

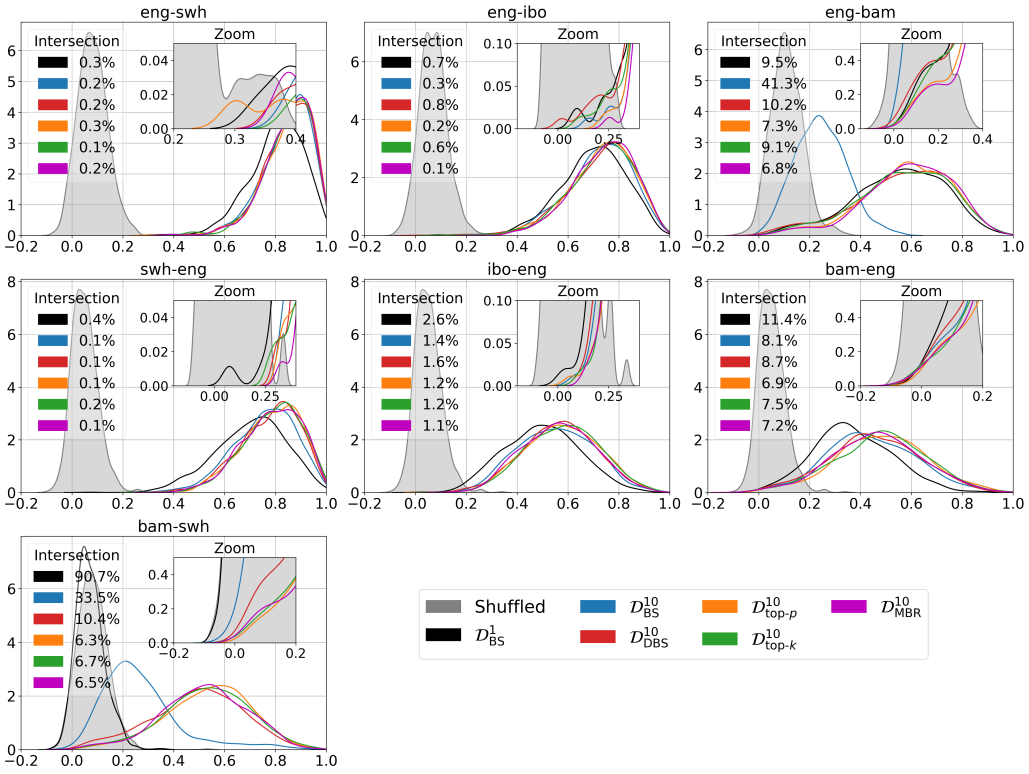

Fig. 13. Kernel density estimations (bandwidth=1.0) for SONAR-based cosine similarities between the output produced by student models trained on samples generated with different decoding methods and the reference translations. "Shuffled" denotes the reference sentences shuffled to simulate hallucinations. The intersection highlights the overlap between each model's output and the shuffled area.

sequence-level KD and word-level KD, as well as GKD, which we use as a representative method that leverages the student's prior knowledge. For MHD, the training corpus was generated with top-p sampling with $M = 10$ ($\mathcal{D}^{10}_{\text{top-}p}$), while the rest of the techniques utilised the corpus $\mathcal{D}^1_{\text{BS}}$ generated with beam search. We used the same source corpus of 100k sentences for each language pair. Detailed hyperparameter configurations are provided in Appendix B.

As shown in Fig. 14, MHD consistently outperforms the alternative methods.[12] Compared to sequence-level KD ($\mathcal{D}^1_{\text{BS}}$), the improvement stems from the same principle discussed in Section 5.1: MHD provides the student with a richer and more diverse signal from the teacher.

Notably, word-level KD (WL($\mathcal{D}^1_{\text{BS}}$)) slightly degrades the performance compared to the student model before KD. As previously observed in Section 5.1, leveraging the full probability distribution of the teacher can be detrimental when the teacher is not correctly fitted for specific language pairs (Fig. 2), leading to high-entropy distributions that confuse the student. We conducted an additional ablation experiment using only the Kullback-Leibler divergence between the teacher and student distributions, omitting the cross-entropy loss with the synthetic

[12]The difference between the models trained on $\mathcal{D}^{10}_{\text{top-}p}$ (MHD) and the other 4 methods presented in Fig. 14 is statistically signifcant for all the language pairs.

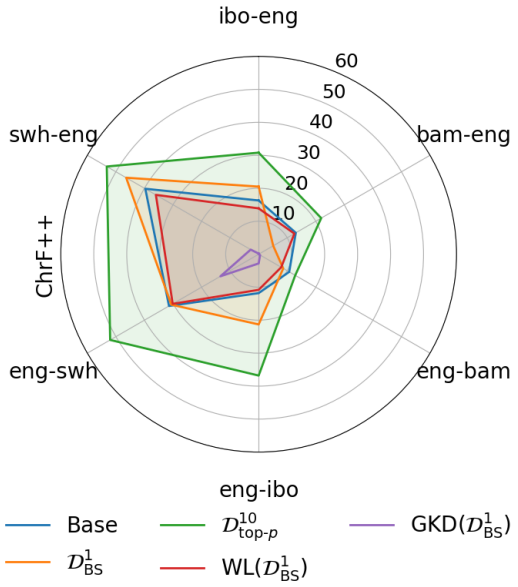

Fig. 14. ChrF++ scores of different KD techniques. Base represents the results of Llama-3.2-1B before KD.

target. In this case, the training process collapsed completely, further confirming that a raw distribution-matching approach is insufficient when the teacher's signal is noisy.

GKD (GKD($\mathcal{D}^{1}_{\text{BS}}$)) faces significant challenges in this setup due to its reliance on student-generated prefixes. For the low-resource languages in this study, small LLMs do not yet provide sufficient translation quality; consequently, the generated prefixes fail to provide a constructive learning signal. Furthermore, GKD explores the full probability distribution of the teacher, which leads to the same calibration issues as word-level KD. These results preclude the use of other KD techniques that require prior knowledge of the task from the student for many low-resource languages.

MHD provides benefits similar to word-level KD and GKD in terms of exploring the teacher's probability distribution more deeply than sequence-level KD, but it avoids their drawbacks. By restricting exploration to a high-probability subspace through multiple sampled hypotheses, MHD avoids the degradation caused by the noisy regions of the teacher distribution. Moreover, by providing diverse target-side prefixes, MHD offers the advantages of on-policy training without the computational burden of autoregressive generation during training, thereby accelerating training and enabling effective KD even into student models with insufficient prior knowledge.

5.7 Efficiency

The practical viability of KD techniques is often determined by the balance between the performance gains they offer and the computational resources they demand. In the context of low-resource languages, where researchers and organisations may have limited access to high-end infrastructure, efficiency is particularly relevant.

Notably, MHD does not increase synthetic-data generation time relative to sequence-level KD, as sampling is faster than beam search, and the latter already generates multiple hypotheses internally. While student training time naturally increases with the enlarged dataset, it remains significantly lower than that of GKD. GKD not only

requires keeping the teacher in memory during training, as word-level KD does, but it also involves expensive autoregressive generation by the student.

To illustrate this, the training time for one epoch of Llama-3.2 1B on a single A100 GPU (40GB) in the experiments of Section 5.6 is 0.99 h for sequence-level KD, 2.2 h for word-level KD, 9.5 h for MHD (scaling linearly with M relative to sequence-level), and 50 h for GKD. In terms of cost-effectiveness, MHD is the best method. It should be noted that these figures correspond to a fixed effective batch size of 32 to ensure a fair comparison. However, the varying memory requirements of each method allow for further acceleration when VRAM is available. Word-level KD and GKD require the teacher to be kept in memory, occupying 32 GB of VRAM in this setup. In contrast, sequence-level KD and MHD only require the student model, consuming only 14 GB. This lower memory footprint allows MHD to utilise significantly larger batch sizes on the same hardware, further accelerating training compared to word-level based methods. This also makes MHD feasible in resource-constrained scenarios where only small GPUs are available.

Regarding storage costs, the MHD dataset size grows by a factor of M. For example, a 100k-sentence corpus expands from approximately 10 MB to 100 MB, while a 1M-sentence corpus grows from 100 MB to 1 GB. These amounts remain negligible when compared to the size of the teacher model itself; for instance, the 1.3B-parameter NLLB-200 occupies 11 GB on disk. Consequently, the additional storage required by MHD is modest relative to the overall system footprint and the significant performance gains it provides.

6 Concluding Remarks

This study has investigated the effectiveness of MHD, a technique that generates multiple translations from the same source sentence for KD, as well as the impact of different decoding methods.

First, we show that MHD is broadly applicable across model types: it is effective when distilling both encoder-decoder models and LLMs, and it improves performance regardless of whether the student is an encoder-decoder trained from scratch or a pretrained LLM. Across all configurations, MHD consistently outperforms standard sequence-level KD and other alternatives such as word-level KD and GKD. By relying on multiple offline-generated hypotheses, MHD avoids the instability of on-policy training and the computational overhead of maintaining the teacher model in memory, making it a computationally efficient and robust method, particularly well suited for low-resource scenarios where both data and compute are limited.

Regarding the different scenarios defined by the availability of monolingual corpora (Section 4.2), MHD matches or slightly outperforms the teacher from English to low-resource languages (scenario "from English" as described in Section 4.2), where up to 1M monolingual sentences are available, but leaves a gap in translation into English. In multilingual models, it may not be possible to extract all the bilingual knowledge from the teacher model with only the synthetic parallel corpus of one language pair, since thanks to positive transfer learning, part of the translation ability comes from other language pairs. In NLLB-200, which is trained on different parallel corpora with English as the target, a relatively small monolingual corpus (containing between 100k and 1M sentences depending on the source language) translated into English by the teacher cannot capture all the English knowledge of the model. In this scenario ("into English" in Section 4.2), MHD produces better results than traditional KD with BS ($\mathcal{D}^1_{\text{BS}}$), but does not improve the performance of the teacher. A similar pattern is observed in the "zero-shot" scenario, where only 100k monolingual sentences are available, with the key difference being that in this case $\mathcal{D}^1_{\text{BS}}$ fails to train effective student models. In contrast, datasets generated using MHD allow the training of student models that achieve performance close to that of the teacher.

Second, when generating multiple hypotheses, sampling methods extract more useful information from the teacher than deterministic approaches. This advantage becomes particularly clear when the teacher is not well fitted to the language pair. Sampling methods have a natural advantage over deterministic approaches because the diversity they introduce reflects the probability distribution of the teacher model. Methods such as top-p

generate each translation independently by following the model's learned distribution. This allows the synthetic corpus to capture alternative translations that the teacher has assigned a significant probability. By contrast, deterministic methods enforce dissimilarity among hypotheses, often introducing artificial diversity by forcing exploration of regions with low probability. Consequently, increasing the number of BS hypotheses leads to quality degradation, while sampling methods can scale to a large number of translations without degradation. This allows MHD to achieve, in some cases, better performance with very small corpora than sequence-level KD trained on much larger datasets.

Third, the benefits of MHD are most pronounced in low-resource scenarios, where the amount of available monolingual data is limited. In these settings, MHD significantly improves vocabulary coverage by exposing the student to multiple alternative translations per source sentence, especially when using sampling methods. As the size of the monolingual corpus increases, this advantage becomes less pronounced, although MHD still provides improvements. In higher-resource settings, leveraging the inherent diversity of the source corpus together with high-quality deterministic translations becomes increasingly effective.

Finally, increasing hypothesis diversity has additional benefits beyond translation quality. MHD consistently reduces hallucinations and mitigates gender bias amplification across all decoding methods. Sampling-based approaches are particularly effective in this regard, as they avoid the over-representation of the most likely tokens inherent to beam search.

More generally, the choice of the decoding method plays a key role in MHD. Methods such as MBR achieve strong performance in extremely low-resource settings, but are computationally expensive. In contrast, top-p sampling provides a better trade-off between performance and efficiency, making it preferable in most scenarios. Deterministic methods can still be advantageous when the teacher model is highly reliable for the language pair and the available corpus is sufficiently large, as high-probability translations together with data diversity can compensate for the limited variability introduced by the decoding process.

It is worth noting that our study is limited to a set of African languages and English. Although MHD does not rely on any language-specific components, and our experiments include unrelated language pairs, we cannot fully guarantee that these conclusions extend to other language families, as this has not been empirically verified.

Additionally, the current formulation of MHD treats all generated hypotheses equally, regardless of the extra information or the quality that each one carries. A promising avenue for future work would be to weight each hypothesis according to the teacher model's confidence and its variability with respect to the other hypotheses generated for the same source sentence, potentially allowing the student to give more weight to more informative or reliable translations.

Ethics Statement

Knowledge distillation endeavors to produce smaller, more resource-efficient MT systems, thereby diminishing energy requirements compared to the original teacher systems and consequently aiding in the reduction of CO_2 emissions. Moreover, it lowers the entry barrier for deploying MT models, as the resulting models work on lower power hardware. Our student models are remarkably compact, operating at a mere 5% of the teacher model size. However, delving into knowledge distillation necessitates a substantial number of training iterations, each accompanied by its own energy consumption. For the experiments detailed in this paper, we trained 522 Transformer models employing NVIDIA GeForce RTX 2080 Ti GPUs and fine-tuned 18 Llama-3.2-1B models employing NVIDIA A100-SXM4 (40GB) GPUs. Furthermore, all corpora and tools utilised in this study are available under open source licenses, ensuring the complete reproducibility of the presented results.

Acknowledgments

This paper is part of the R+D+i projects PID2021-27999NB-I00 and PID2024-158157OB-C31 funded by the Spanish Ministry of Science and Innovation (MCIN), the Spanish Research Agency (AEI/10.13039/501100011033) and the European Regional Development Fund A way to make Europe. It is also part of the project GRE23-08A funded by the University of Alicante. Some of the computational resources used were funded by the Valencia Government and the European Regional Development Fund (ERDF) through project IDIFEDER/2020/003.

References

D. Adelani et al.. Dec. 2022. "Findings of the WMT'22 Shared Task on Large-Scale Machine Translation Evaluation for African Languages." In: *Proceedings of the Seventh Conference on Machine Translation (WMT)*. Ed. by P. Koehn et al. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates (Hybrid), (Dec. 2022), 773–800. https://aclanthology.org/2022.wmt-1.72.

R. Agarwal, N. Vieillard, Y. Zhou, P. Stanczyk, S. R. Garea, M. Geist, and O. Bachem. 2024. "On-Policy Distillation of Language Models: Learning from Self-Generated Mistakes." In: *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=3zKtaqxLhW.

J. Ahn, H. Lee, J. Kim, and A. Oh. July 2022. "Why Knowledge Distillation Amplifies Gender Bias and How to Mitigate from the Perspective of DistilBERT." In: *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*. Association for Computational Linguistics, Seattle, Washington, (July 2022), 266–272. doi:10.18653/v1/2022.gebnlp-1.27.

S. Basu, G. S. Ramachandran, N. S. Keskar, and L. R. Varshney. 2021. *Mirostat: A Neural Text Decoding Algorithm that Directly Controls Perplexity*. (2021). arXiv: 2007.14966 (cs.CL).

N. Boizard, K. E. Haddad, C. Hudelot, and P. Colombo. 2025. *Towards Cross-Tokenizer Distillation: the Universal Logit Distillation Loss for LLMs*. (2025). https://arxiv.org/abs/2402.12030 arXiv: 2402.12030 (cs.CL).

L. Burchell, A. Birch, and K. Heafield. July 2022. "Exploring diversity in back translation for low-resource machine translation." In: *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*. Association for Computational Linguistics, Hybrid, (July 2022), 67–79. doi:10.18653/v1/2022.deeplo-1.8.

D. Dale, E. Voita, L. Barrault, and M. R. Costa-jussà. July 2023. "Detecting and Mitigating Hallucinations in Machine Translation: Model Internal Workings Alone Do Well, Sentence Similarity Even Better." In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by A. Rogers, J. Boyd-Graber, and N. Okazaki. Association for Computational Linguistics, Toronto, Canada, (July 2023), 36–50. doi:10.18653/v1/2023.acl-long.3.

O. De Gibert, R. Vázquez, M. Aulamo, Y. Scherrer, S. Virpioja, and J. Tiedemann. July 2023. "Four Approaches to Low-Resource Multilingual NMT: The Helsinki Submission to the AmericasNLP 2023 Shared Task." In: *Proceedings of the Workshop on Natural Language Processing for Indigenous Languages of the Americas (AmericasNLP)*. Ed. by M. Mager, A. Ebrahimi, A. Oncevay, E. Rice, S. Rijhwani, A. Palmer, and K. Kann. Association for Computational Linguistics, Toronto, Canada, (July 2023), 177–191. doi:10.18653/v1/2023.americasnlp-1.20.

A. DeLucia, A. Mueller, X. L. Li, and J. Sedoc. Aug. 2021. "Decoding Methods for Neural Narrative Generation." In: *Proceedings of the 1st Workshop on Natural Language Generation, Evaluation, and Metrics (GEM 2021)*. Association for Computational Linguistics, Online, (Aug. 2021), 166–185. doi:10.18653/v1/2021.gem-1.16.

H. Do and G. G. Lee. Mar. 2023. "Target-Oriented Knowledge Distillation with Language-Family-Based Grouping for Multilingual NMT." *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 22, 2, Article 42, (Mar. 2023), 18 pages. doi:10.1145/3546067.

P.-A. Duquenne, H. Schwenk, and B. Sagot. 2023. *SONAR: Sentence-Level Multimodal and Language-Agnostic Representations*. (2023). https://arxiv.org/abs/2308.11466 arXiv: 2308.11466 (cs.CL).

B. Eikema and W. Aziz. Dec. 2020. "Is MAP Decoding All You Need? The Inadequacy of the Mode in Neural Machine Translation." In: *Proceedings of the 28th International Conference on Computational Linguistics*. International Committee on Computational Linguistics, Barcelona, Spain (Online), (Dec. 2020), 4506–4520. doi:10.18653/v1/2020.coling-main.398.

B. Eikema and W. Aziz. Dec. 2022. "Sampling-Based Approximations to Minimum Bayes Risk Decoding for Neural Machine Translation." In: *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, (Dec. 2022), 10978–10993. doi:10.18653/v1/2022.emnlp-main.754.

M. Enis and M. Hopkins. 2024. *From LLM to NMT: Advancing Low-Resource Machine Translation with Claude*. (2024). https://arxiv.org/abs/2404.13813 arXiv: 2404.13813 (cs.CL).

A. Fan, S. Bhosale, et al.. 2021. "Beyond English-Centric Multilingual Machine Translation." *Journal of Machine Learning Research*, 22, 107, 1–48. http://jmlr.org/papers/v22/20-1307.html.

A. Fan, M. Lewis, and Y. Dauphin. 2018. *Hierarchical Neural Story Generation*. (2018). arXiv: 1805.04833 (cs.CL).

M. Finkelstein and M. Freitag. 2024. "MBR and QE Finetuning: Training-time Distillation of the Best and Most Expensive Decoding Methods." In: *The Twelfth International Conference on Learning Representations*. https://openreview.net/forum?id=bkNx3O0sND.

A. Galiano-Jiménez, J. A. Pérez-Ortiz, F. Sánchez-Martínez, and V. M. Sánchez-Cartagena. Apr. 2025. "Beyond the Mode: Sequence-Level Distillation of Multilingual Translation Models for Low-Resource Language Pairs." In: *Findings of the Association for Computational Linguistics: NAACL 2025*. Ed. by L. Chiruzzo, A. Ritter, and L. Wang. Association for Computational Linguistics, Albuquerque, New Mexico, (Apr. 2025), 6661–6676. ISBN: 979-8-89176-195-7. doi:10.18653/v1/2025.findings-naacl.372.

A. Galiano-Jiménez, F. Sánchez-Martínez, V. M. Sánchez-Cartagena, and J. A. Pérez-Ortiz. June 2023. "Exploiting large pre-trained models for low-resource neural machine translation." In: *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*. Ed. by M. Nurminen et al. European Association for Machine Translation, Tampere, Finland, (June 2023), 59–68. https://aclanthology.org/2023.eamt-1.7.

V. Goyal, S. Kumar, and D. M. Sharma. July 2020. "Efficient Neural Machine Translation for Low-Resource Languages via Exploiting Related Languages." In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*. Association for Computational Linguistics, Online, (July 2020), 162–168. doi:10.18653/v1/2020.acl-srw.22.

A. Grattafiori et al.. 2024. *The Llama 3 Herd of Models*. (2024). https://arxiv.org/abs/2407.21783 arXiv: 2407.21783 (cs.AI).

A. Graves. 2012. *Sequence Transduction with Recurrent Neural Networks*. (2012). arXiv: 1211.3711 (cs.NE).

N. M. Guerreiro, E. Voita, and A. Martins. May 2023. "Looking for a Needle in a Haystack: A Comprehensive Study of Hallucinations in Neural Machine Translation." In: *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. Ed. by A. Vlachos and I. Augenstein. Association for Computational Linguistics, Dubrovnik, Croatia, (May 2023), 1059–1075. doi:10.18653/v1/2023.eacl-main.75.

V. Gumma, R. Dabre, and P. Kumar. June 2023. "An Empirical Study of Leveraging Knowledge Distillation for Compressing Multilingual Neural Machine Translation Models." In: *Proceedings of the 24th Annual Conference of the European Association for Machine Translation*. Ed. by M. Nurminen et al. European Association for Machine Translation, Tampere, Finland, (June 2023), 103–114. https://aclanthology.org/2023.eamt-1.11.

J. Hewitt, C. Manning, and P. Liang. Dec. 2022. "Truncation Sampling as Language Model Desmoothing." In: *Findings of the Association for Computational Linguistics: EMNLP 2022*. Ed. by Y. Goldberg, Z. Kozareva, and Y. Zhang. Association for Computational Linguistics, Abu Dhabi, United Arab Emirates, (Dec. 2022), 3414–3427. doi:10.18653/v1/2022.findings-emnlp.249.

G. Hinton, O. Vinyals, and J. Dean. 2015. *Distilling the Knowledge in a Neural Network*. (2015). arXiv: 1503.02531 (stat.ML).

A. Holtzman, J. Buys, L. Du, M. Forbes, and Y. Choi. 2020. *The Curious Case of Neural Text Degeneration*. (2020). arXiv: 1904.09751 (cs.CL).

V. Iyer, B. Malik, P. Stepachev, P. Chen, B. Haddow, and A. Birch. Nov. 2024. "Quality or Quantity? On Data Scale and Diversity in Adapting Large Language Models for Low-Resource Translation." In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by B. Haddow, T. Kocmi, P. Koehn, and C. Monz. Association for Computational Linguistics, Miami, Florida, USA, (Nov. 2024), 1393–1409. doi:10.18653/v1/2024.wmt-1.128.

S. Ji, Z. Li, J. Paavola, H. Luo, and J. Tiedemann. 2025. *Massively Multilingual Adaptation of Large Language Models Using Bilingual Translation Data*. (2025). https://arxiv.org/abs/2506.00469 arXiv: 2506.00469 (cs.CL).

Y. Kim and A. M. Rush. Nov. 2016. "Sequence-Level Knowledge Distillation." In: *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, (Nov. 2016), 1317–1327. doi:10.18653/v1/D16-1139.

D. P. Kingma and J. Ba. 2015. "Adam: A Method for Stochastic Optimization." In: *3rd International Conference on Learning Representations, ICLR 2015, Conference Track Proc*. http://arxiv.org/abs/1412.6980.

T. Kocmi et al.. Nov. 2024. "Findings of the WMT24 General Machine Translation Shared Task: The LLM Era Is Here but MT Is Not Solved Yet." In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by B. Haddow, T. Kocmi, P. Koehn, and C. Monz. Association for Computational Linguistics, Miami, Florida, USA, (Nov. 2024), 1–46. doi:10.18653/v1/2024.wmt-1.1.

G. Kovacs, D. Deutsch, and M. Freitag. Nov. 2024. "Mitigating Metric Bias in Minimum Bayes Risk Decoding." In: *Proceedings of the Ninth Conference on Machine Translation*. Ed. by B. Haddow, T. Kocmi, P. Koehn, and C. Monz. Association for Computational Linguistics, Miami, Florida, USA, (Nov. 2024), 1063–1094. doi:10.18653/v1/2024.wmt-1.109.

T. Kudo and J. Richardson. Nov. 2018. "SentencePiece: A simple and language independent subword tokenizer and detokenizer for Neural Text Processing." In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Brussels, Belgium, (Nov. 2018), 66–71. doi:10.18653/v1/D18-2012.

S. Kudugunta et al.. 2023. *MADLAD-400: A Multilingual And Document-Level Large Audited Dataset*. (2023). arXiv: 2309.04662 (cs.CL).

I. Kulikov, A. Miller, K. Cho, and J. Weston. Oct. 2019. "Importance of Search and Evaluation Strategies in Neural Dialogue Modeling." In: *Proceedings of the 12th International Conference on Natural Language Generation*. Association for Computational Linguistics, Tokyo, Japan, (Oct. 2019), 76–87. doi:10.18653/v1/W19-8609.

S. Kullback and R. A. Leibler. 1951. "On Information and Sufficiency." *The Annals of Mathematical Statistics*, 22, 1, 79–86.

S. Kumar and W. Byrne. May 2004. "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*. Association for Computational Linguistics, Boston, Massachusetts, USA, (May 2004), 169–176. https://aclanthology.org/N04-1022.

W. Lai, J. Libovický, and A. Fraser. Nov. 2021. "The LMU Munich System for the WMT 2021 Large-Scale Multilingual Machine Translation Shared Task." In: *Proceedings of the Sixth Conference on Machine Translation*. Association for Computational Linguistics, Online, (Nov. 2021), 412–417. https://aclanthology.org/2021.wmt-1.49.

J. Li, S. Cheng, S. Huang, and J. Chen. June 2024. "MT-PATCHER: Selective and Extendable Knowledge Distillation from Large Language Models for Machine Translation." In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Ed. by K. Duh, H. Gomez, and S. Bethard. Association for Computational Linguistics, Mexico City, Mexico, (June 2024), 6445–6459. doi:10.18653/v1/2024.naacl-long.358.

S. Li et al.. 2025. *SSA-COMET: Do LLMs Outperform Learned Metrics in Evaluating MT for Under-Resourced African Languages?* (2025). https://arxiv.org/abs/2506.04557 arXiv: 2506.04557 (cs.CL).

M. Müller and R. Sennrich. Aug. 2021. "Understanding the Properties of Minimum Bayes Risk Decoding in Neural Machine Translation." In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, Online, (Aug. 2021), 259–272. doi:10.18653/v1/2021.acl-long.22.

NLLB Team et al.. 2022. *No Language Left Behind: Scaling Human-Centered Machine Translation*. (2022). doi:10.48550/ARXIV.2207.04672.

K. Pillutla, S. Swayamdipta, R. Zellers, J. Thickstun, S. Welleck, Y. Choi, and Z. Harchaoui. 2021. *MAUVE: Measuring the Gap Between Neural Text and Human Text using Divergence Frontiers*. (2021). arXiv: 2102.01454 (cs.CL).

M. Popović. Sept. 2017. "chrF++: words helping character n-grams." In: *Proceedings of the Second Conference on Machine Translation*. Association for Computational Linguistics, Copenhagen, Denmark, (Sept. 2017), 612–618. doi:10.18653/v1/W17-4770.

M. Ranzato, S. Chopra, M. Auli, and W. Zaremba. 2015. "Sequence Level Training with Recurrent Neural Networks." *CoRR*, abs/1511.06732. https://api.semanticscholar.org/CorpusID:7147309.

R. Rei, C. Stewart, A. C. Farinha, and A. Lavie. Nov. 2020. "COMET: A Neural Framework for MT Evaluation." In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, (Nov. 2020), 2685–2702. doi:10.18653/v1/2020.emnlp-main.213.

S. Riezler and J. T. Maxwell. June 2005. "On Some Pitfalls in Automatic Evaluation and Significance Testing for MT." In: *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*. Association for Computational Linguistics, Ann Arbor, Michigan, (June 2005), 57–64. https://aclanthology.org/W05-0908.

V. M. Sánchez-Cartagena, M. Bañón, S. Ortiz-Rojas, and G. Ramírez-Sánchez. Oct. 2018. "Prompsit's submission to WMT 2018 Parallel Corpus Filtering shared task." In: *Proceedings of the Third Conference on Machine Translation, Volume 2: Shared Task Papers*. Association for Computational Linguistics, Brussels, Belgium, (Oct. 2018).

B. Scalvini, I. N. Debess, A. Simonsen, and H. Einarsson. Mar. 2025. "Rethinking Low-Resource MT: The Surprising Effectiveness of Fine-Tuned Multilingual Models in the LLM Age." In: *Proceedings of the Joint 25th Nordic Conference on Computational Linguistics and 11th Baltic Conference on Human Language Technologies (NoDaLiDa/Baltic-HLT 2025)*. Ed. by R. Johansson and S. Stymne. University of Tartu Library, Tallinn, Estonia, (Mar. 2025), 609–621. ISBN: 978-9908-53-109-0. https://aclanthology.org/2025.nodalida-1.62/.

C. Shi, H. Yang, D. Cai, Z. Zhang, Y. Wang, Y. Yang, and W. Lam. Nov. 2024. "A Thorough Examination of Decoding Methods in the Era of LLMs." In: *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*. Ed. by Y. Al-Onaizan, M. Bansal, and Y.-N. Chen. Association for Computational Linguistics, Miami, Florida, USA, (Nov. 2024), 8601–8629. doi:10.18653/v1/2024.emnlp-main.489.

Y. Song, S. Ezzini, J. Klein, T. Bissyande, C. Lefebvre, and A. Goujon. 2023. *Letz Translate: Low-Resource Machine Translation for Luxembourgish*. (2023). arXiv: 2303.01347 (cs.CL).

Y. Song, L. Ding, C. Zan, and S. Huang. Jan. 2025. "Self-Evolution Knowledge Distillation for LLM-based Machine Translation." In: *Proceedings of the 31st International Conference on Computational Linguistics*. Ed. by O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert. Association for Computational Linguistics, Abu Dhabi, UAE, (Jan. 2025), 10298–10308. https://aclanthology.org/2025.coling-main.686/.

G. Stanovsky, N. A. Smith, and L. Zettlemoyer. July 2019. "Evaluating Gender Bias in Machine Translation." In: *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, Florence, Italy, (July 2019), 1679–1684. doi:10.18653/v1/P19-1164.

Y. Su, T. Lan, Y. Wang, D. Yogatama, L. Kong, and N. Collier. 2022. *A Contrastive Framework for Neural Text Generation*. (2022). arXiv: 2202.06417 (cs.CL).

X. Tan, Y. Ren, D. He, T. Qin, and T.-Y. Liu. 2019. "Multilingual Neural Machine Translation with Knowledge Distillation." In: *Seventh International Conference on Learning Representations*. https://openreview.net/forum?id=S1gUsoR9YX.

C. Tran, S. Bhosale, J. Cross, P. Koehn, S. Edunov, and A. Fan. 2021. "Facebook AI WMT21 News Translation Task Submission." In: *Proc. of the Sixth Conference on Machine Translation (WMT)*, 205–215.

J. Vamvas and R. Sennrich. Nov. 2021. "Contrastive Conditioning for Assessing Disambiguation in MT: A Case Study of Distilled Bias." In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, (Nov. 2021), 10246–10265. doi:10.18653/v1/2021.emnlp-main.803.

J. Vamvas and R. Sennrich. Aug. 2024. "Linear-time Minimum Bayes Risk Decoding with Reference Aggregation." In: *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Ed. by L.-W. Ku, A. Martins, and V. Srikumar. Association for Computational Linguistics, Bangkok, Thailand, (Aug. 2024), 790–801. doi:10.18653/v1/2024.acl-short.71.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin. 2017. "Attention is all you need." In: *Proceedings of the 31st International Conference on Neural Information Processing Systems* (NIPS'17). Curran Associates Inc., Long Beach, California, USA, 6000–6010. ISBN: 9781510860964.

A. Vijayakumar, M. Cogswell, R. Selvaraju, Q. Sun, S. Lee, D. Crandall, and D. Batra. Apr. 2018. "Diverse Beam Search for Improved Description of Complex Scenes." *Proceedings of the AAAI Conference on Artificial Intelligence*, 32, 1, (Apr. 2018). doi:10.1609/aaai.v32i1.12340.

J. Wang, D. I. Adelani, et al.. June 2024. "AfriMTE and AfriCOMET: Enhancing COMET to Embrace Under-resourced African Languages." In: *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*. Ed. by K. Duh, H. Gomez, and S. Bethard. Association for Computational Linguistics, Mexico City, Mexico, (June 2024), 5997–6023. doi:10.18653/v1/2024.naacl-long.334.

J. Wang, E. Briakou, H. Dadkhahi, R. Agarwal, C. Cherry, and T. Cohn. 2024. *Don't Throw Away Data: Better Sequence Knowledge Distillation*. (2024). https://arxiv.org/abs/2407.10456 arXiv: 2407.10456 (cs.CL).

G. Wiher, C. Meister, and R. Cotterell. 2022. *On Decoding Strategies for Neural Text Generators*. (2022). arXiv: 2203.15721 (cs.CL).

T. Wolf et al.. Oct. 2020. "Transformers: State-of-the-Art Natural Language Processing." In: *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online, (Oct. 2020), 38–45. doi:10.18653/v1/2020.emnlp-demos.6.

W. Xu, S. Agrawal, E. Briakou, M. J. Martindale, and M. Carpuat. 2023. "Understanding and Detecting Hallucinations in Neural Machine Translation via Model Introspection." *Transactions of the Association for Computational Linguistics*, 11, 546–564. doi:10.1162/tacl_a_00563.

Z. Yu et al.. Nov. 2021. "HW-TSC's Participation in the WMT 2021 Large-Scale Multilingual Translation Task." In: *Proceedings of the Sixth Conference on Machine Translation*. Association for Computational Linguistics, Online, (Nov. 2021), 456–463. https://aclanthology.org/2021.wmt-1.55.

H. Zhang, D. Duckworth, D. Ippolito, and A. Neelakantan. Apr. 2021. "Trading Off Diversity and Quality in Natural Language Generation." In: *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*. Association for Computational Linguistics, Online, (Apr. 2021), 25–33. https://aclanthology.org/2021.humeval-1.3.

S. Zhang, Y. Liang, S. Wang, Y. Chen, W. Han, J. Liu, and J. Xu. July 2023. "Towards Understanding and Improving Knowledge Distillation for Neural Machine Translation." In: *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Ed. by A. Rogers, J. Boyd-Graber, and N. Okazaki. Association for Computational Linguistics, Toronto, Canada, (July 2023), 8062–8079. doi:10.18653/v1/2023.acl-long.448.

Y. Zhang et al.. 2020. "The niutrans machine translation systems for wmt20." In: *Proceedings of the Fifth Conference on Machine Translation*, 338–345.

Y. Zhao, Y. Zhang, L. Xiang, J. Zhang, Y. Zhou, and C. Zong. 2025. "Imitate, Reward and Introspect: Enhancing Neural Machine Translation by Utilizing Large Language Models as Environments." *IEEE Transactions on Audio, Speech and Language Processing*, 33, 4736–4747. doi:10.1109/TASLPRO.2025.3623904.

W. Zhu, H. Liu, Q. Dong, J. Xu, S. Huang, L. Kong, J. Chen, and L. Li. June 2024. "Multilingual Machine Translation with Large Language Models: Empirical Results and Analysis." In: *Findings of the Association for Computational Linguistics: NAACL 2024*. Ed. by K. Duh, H. Gomez, and S. Bethard. Association for Computational Linguistics, Mexico City, Mexico, (June 2024), 2765–2781. doi:10.18653/v1/2024.findings-naacl.176.

Y. Zhu, S. Lu, L. Zheng, J. Guo, W. Zhang, J. Wang, and Y. Yu. 2018. "Texygen: A Benchmarking Platform for Text Generation Models." In: *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (SIGIR '18). Association for Computing Machinery, Ann Arbor, MI, USA, 1097–1100. ISBN: 9781450356572. doi:10.1145/3209978.3210080.

A Dataset Formalisation

This appendix formalises the generation of M hypotheses with each decoding method Z and their integration into the datasets used for MHD. We define each set of M translations per source sentence x^i as $\tilde{\mathcal{Y}}_Z^i = \{\tilde{y}^{i,1}, ..., \tilde{y}^{i,M}\}$. Therefore, each dataset containing N parallel sentences is defined as:

$$\mathcal{D}_Z^M = \bigcup_{i=1}^{N} \bigcup_{m=1}^{M} \{(x^i, \tilde{y}^{i,m})\} \tag{4}$$

where $\tilde{y}^{i,m} \in \tilde{\mathcal{Y}}_Z^i$.

Beam Search (BS). Beam Search maintains a set of n partial hypotheses $\mathcal{H}_t$ at each time step t, expanding them by selecting the most n probable tokens over the vocabulary $\mathcal{V}$:

$$\mathcal{H}_{t+1} = \max_n \left(\mathcal{H}_t \times \mathcal{V}\right), \tag{5}$$

Where $\max_n$ selects the n hypotheses with the highest probability:

$$P(y_{1:t} \mid x^i) = P(y_{1:t-1} \mid x^i) \cdot P(y_t \mid y_{<t}, x^i). \tag{6}$$

The final set of M translations is:

$$\tilde{\mathcal{Y}}^i_{\text{BS}} = \max_M \left(\mathcal{H}_T\right). \tag{7}$$

Note that n must be equal to or greater than M.

Diverse Beam Search (DBS). Diverse Beam Search divides the beam into G groups and applies a penalty $\lambda(y_{1:t})$. Each group g is defined as:

$$\mathcal{H}^g_{t+1} = \text{top-}n_g \left(\mathcal{H}^g_t \times \mathcal{V}\right), \tag{8}$$

Where the probability of each hypothesis is adjusted with the diversity penalty:

$$P(y_{1:t} \mid x^i) = P(y_{1:t-1} \mid x^i) \cdot P(y_t \mid y_{<t}, x^i) - \lambda(y_{1:t}). \tag{9}$$

The final set of M translations is:

$$\tilde{\mathcal{Y}}^i_{\text{DBS}} = \text{top-}M \left(\bigcup_{g=1}^{G} \mathcal{H}^g_T\right). \tag{10}$$

Top-k Sampling. At each time step t, the set of the k most probable tokens is defined as:

$$\mathcal{V}_t = \max_k \left(P(y_t \mid y_{<t}, x^i)\right). \tag{11}$$

A translation is generated by sampling from the renormalised distribution $P_{\mathcal{V}_t}$ over $\mathcal{V}_t$:

$$y_t \sim P_{\mathcal{V}_t}(y_t \mid y_{<t}, x^i), \tag{12}$$

The set of M generated translations is:

$$\tilde{\mathcal{Y}}^i_{\text{top-}k} = \left\{\tilde{y}^{i,m} \mid \tilde{y}^{i,m} = \{y_t \sim \mathcal{V}_t\}_{t=1}^{T}\right\}_{m=1}^{M}. \tag{13}$$

Top-p. The set of tokens whose cumulative probability mass reaches p is defined as:

$$\mathcal{V}_t = \left\{ y_t \in \text{argmin}_{\mathcal{V}} \sum_{y \in \mathcal{V}} P(y \mid y_{<t}, x^i) \geq p \right\}. \tag{14}$$

Where argmin returns the smallest set of tokens with a probability mass of p. A translation is generated by sampling from the renormalised distribution $\mathcal{V}_t$:

$$y_t \sim P_{\mathcal{V}_t}(y_t \mid y_{<t}, x^i), \tag{15}$$

The set of translations is:

$$\tilde{\mathcal{Y}}^i_{\text{top-}p} = \left\{\tilde{y}^{i,m} \mid \tilde{y}^{i,m} = \{y_t \sim \mathcal{V}_t\}_{t=1}^{T}\right\}_{m=1}^{M}. \tag{16}$$

Minimum Bayes Risk (MBR). We first generate a set of n hypotheses $\mathcal{H}(x^i) = \{h_1, h_2, \ldots, h_n\}$ using epsilon sampling (Hewitt et al. 2022). The utility function $U(h, c)$ measures the similarity between a hypotheses h and a candidate c. In our experiments, we computed the utility using fastChrF (Vamvas and Sennrich 2024). The expected utility for a hypothesis h is defined as:

$$U(h) = \sum_{c \in \mathcal{H}(x^{(i)})} P(c \mid x^i) \cdot U(h, c), \tag{17}$$

where $P(c \mid x^i)$ is the probability assigned to candidate c by the teacher model. The optimal translation is the one that maximises the expected utility:

$$\tilde{y}^i = \text{argmax}_{h \in \mathcal{H}(x^i)} U(h). \tag{18}$$

To generate M diverse hypotheses, we select the top M hypotheses with the highest expected utility:

$$\tilde{\mathcal{Y}}^i_{\text{MBR}} = \max_M \left(U(h) \mid h \in \mathcal{H}(x^i) \right). \tag{19}$$

B Training Details

Each encoder-decoder student model consist of a transformer (Vaswani et al. 2017) with 6 layers for both the encoder and the decoder, embedding dimension of 512, feed-forward inner-layer dimension of 2048, and 8 attention heads. All our models were trained using the `Fairseq` toolkit[13] and a different joint bilingual SentencePiece (Kudo and Richardson 2018) model for each language pair, trained on the training samples generated from the teacher with a vocabulary of 10,000 tokens. For training we used a learning rate of 0.0007 with the Adam (Kingma and Ba 2015) optimizer (β_1=0.9, β_2=0.98), 8,000 warm-up updates and 8,000 max tokens. We used dropout of 0.1 and updated the model after 2 training steps. The cross-entropy loss with label smoothing was computed on the development set after every epoch and the best checkpoint was selected after 6 validation steps with no improvement.

To distil EMMA-500-8B into Llama-3.2-1B, we used the SFT Trainer[14] and GOLD Trainer[15] classes from the TRL library.[16] The SFT Trainer was used to implement both standard sequence-level KD and MHD. For GKD, we utilised the GOLD Trainer implementation, as this enables distillation between models with different tokenizers (Boizard et al. 2025). In the GKD experiments, the interpolation parameter was set to $\lambda = 0.5$, providing a balanced supervision between the student's predictions and the reference targets. The GOLD Trainer also allows the use of word-level KD with the following hyperparameters: $\lambda = 0.0$, $\beta = 0.0$, and *uld_crossentropy=0.5*. Table 5 summarises the core hyperparameters used during the fine-tuning process of the student models.

The training was conducted on NVIDIA A100-SXM4 (40GB). While the *per_device_train_batch_size* and *gradient_accumulation_steps* were dynamically adjusted based on the available VRAM and the specific KD method, we maintained a constant effective batch size of 32 for all experiments to ensure a fair comparison.

C Corpora

The largest corpora correspond to English and Swahili. The English corpus is a fragment of OSCAR-3301 dataset[17] and for Swahili we used Monolingual African Languages from ParaCrawls, a collection of corpora available for

[13] https://github.com/facebookresearch/fairseq
[14] https://huggingface.co/docs/trl/sft_trainer
[15] https://huggingface.co/docs/trl/gold_trainer
[16] https://huggingface.co/docs/trl/index
[17] https://huggingface.co/datasets/oscar-corpus/OSCAR-2301

Table 5. Summary of hyperparameters for student pre-trained LLM (Llama-3.2) KD.

Hyperparameter	Value
Effective Batch Size	32
Learning Rate	2×10^{-5}
Training Epochs	1
Optimiser	`paged_adamw_8bit`
LR Scheduler Type	Cosine
Warmup Ratio	0.03

Table 6. BLEU scores for NLLB-200 1.3B on the FLORES+ devtest dataset for different decoding methods: beam search (BS), diverse beam search (DBS), top-p (average of 3 runs), top-k (average of 3 runs), and MBR.

Method	eng-swh	eng-ibo	eng-bam	swh-eng	ibo-eng	bam-eng	bam-swh
BS	33.1	16.1	6.8	42.9	30.3	17.8	11.6
DBS	32.2	13.7	6.1	42.4	30.3	16.2	8.3
top-p	25.1	12.4	4.9	34.8	24.3	14.4	8.9
top-k	21.5	10.9	4.6	28.8	20.9	12.6	8.3
MBR	30.3	15.1	6.3	42.2	29.8	14.9	10.1

the joint task Large-Scale Machine Translation Evaluation for African Languages"at WMT22 (D. Adelani et al. 2022). The Igbo corpus was obtained from the same collection.

To clean these three corpora, we used `monocleaner` (Sánchez-Cartagena et al. 2018). We used the available ready-to-use language packages for English and Swahili and trained a model for Igbo using the Igbo part of the wmt22_african dataset.[18] We removed all sentences with a `monocleaner` score lower than 0.5 and, for English and Swahili, we then randomly picked one million sentences. For Igbo, our final corpus comprises 451,789 sentences.

For Bambara we collected all available corpora in Hugging Face.[19] For the MADLAD-400 (Kudugunta et al. 2023) corpus we used only the clean part. After concatenating these corpora, we removed duplicated sentences and the result was 108,187 sentences.

D Additional Results

This sections reports additional results to measure the effect of decoding methods in sequence-level KD.

D.1 NLLB-200 1.3 Translation Quality

Table 6 shows the impact of each decoding method on the translation quality of NLLB-200 1.3B, evaluated with BLEU.

D.2 Teacher Models Translation Quality

Tables 7 and 8 show the impact of each decoding method on the translation quality of NLLB-200 3.3B, as evaluated using chrF++ and BLEU, respectively.

Table 9 shows the impact of each decoding method on the translation quality of EMMA-500 8B, as evaluated using BLEU.

[18] https://huggingface.co/datasets/allenai/wmt22_african

[19] https://huggingface.co/datasets/RobotsMaliAI/bayelemabaga, https://github.com/masakhane-io/lafand-mt, https://wortschatz.uni-leipzig.de/en/download/Bambara, https://github.com/facebookresearch/flores/tree/main/nllb_seed, https://huggingface.co/datasets/bigscience/xP3, https://huggingface.co/datasets/allenai/MADLAD-400

Table 7. chrF++ scores of NLLB-200 3.3B on the FLORES+ devtest dataset when decoding with beam search, diverse beam search, top-p (average of 3 runs) and top-k (average of 3 runs).

	eng-swh	**eng-ibo**	**eng-bam**	**swh-eng**	**ibo-eng**	**bam-eng**	**bam-swh**
BS	59.2	41.2	31.1	65.0	53.3	39.0	36.0
DBS	59.0	40.6	29.4	64.6	52.7	38.9	35.2
top-p	55.9	38.6	28.5	61.1	49.7	36.8	33.7
top-k	50.4	35.5	27.0	54.8	45.3	34.6	32.0

Table 8. BLEU scores of NLLB-200 3.3B on the FLORES+ devtest dataset when decoding with beam search, diverse beam search, top-p (average of 3 runs) and top-k (average of 3 runs).

	eng-swh	**eng-ibo**	**eng-bam**	**swh-eng**	**ibo-eng**	**bam-eng**	**bam-swh**
BS	33.9	16.2	7.0	44.8	32.0	17.5	10.8
DBS	32.7	15.9	6.0	44.2	31.1	17.1	10.7
top-p	28.9	14.0	5.9	39.7	28.1	15.1	9.3
top-k	22.1	10.8	4.7	30.9	21.9	12.1	7.0

Table 9. BLEU scores of EMMA-500 8B on the FLORES+ devtest dataset when decoding with beam search and top-p.

	eng-swh	**eng-ibo**	**eng-bam**	**swh-eng**	**ibo-eng**	**bam-eng**
BS	35.0	18.4	3.0	45.7	29.0	13.3
top-p	32.5	16.6	3.1	41.2	25.1	11.3

D.3 Experiments with 100k Sentences

Figure 15 shows the BLEU scores obtained by student models trained on corpus generated by NLLB-200 1.3B.

The results obtained using NLLB-200 3.3B are shown in figures 16 and 17.

D.4 Experiments with 500k and 1 Million Sentences

Tables 10 and 11 show the BLEU and chrF++ scores of the trained student models together with the teacher score. The results in Table 11 correspond to those in Fig. 8, together with the directions for which the available corpus does not reach one million sentences.

D.5 Experiments with 100 Samples

Table 12 show the chrF++ scores of student models trained with corpus generated by beam search and top-p sampling with varying numbers of hypotheses, under two source corpus sizes (100k and 1M sentences).

Received 15 April 2026; accepted 21 August 2026

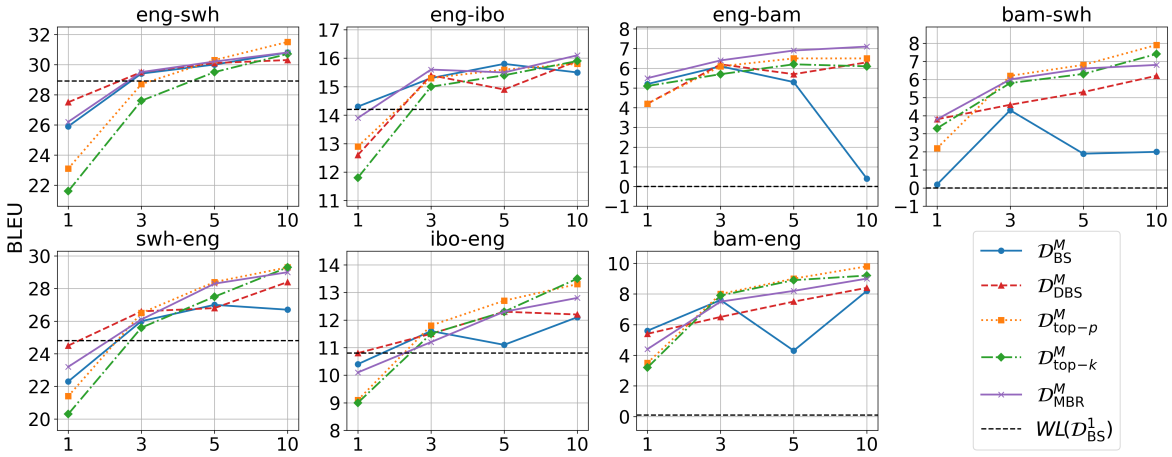

Fig. 15. Average BLEU score obtained by student models trained on M samples generated with different decoding methods (x-axis).

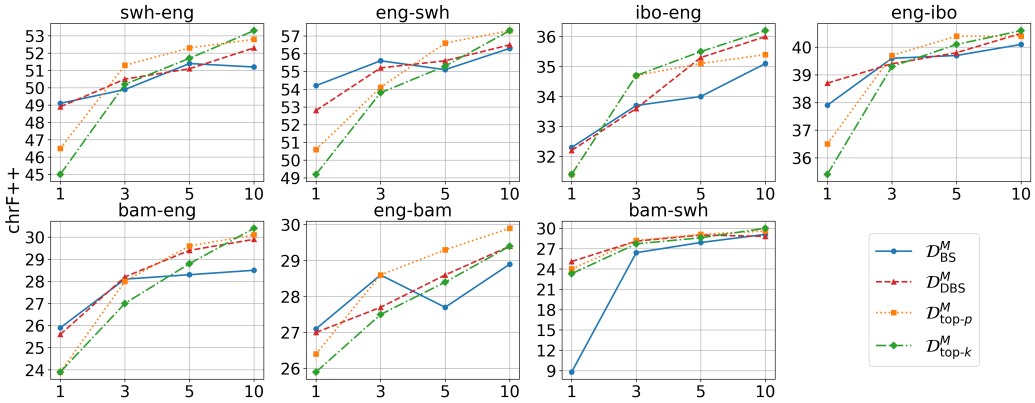

Fig. 16. Average chrF++ score obtained by student models trained on M samples generated by NLLB-200 3.3B with different decoding methods (x-axis).

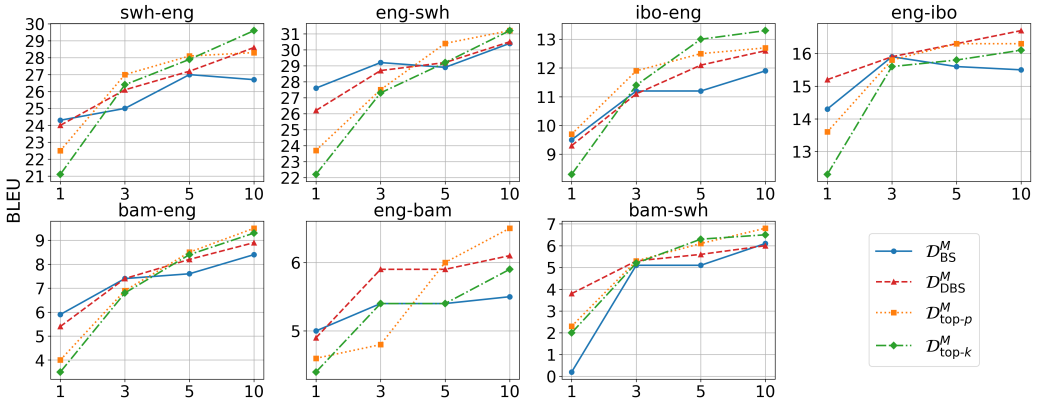

Fig. 17. Average BLEU score obtained by student models trained on M samples generated by NLLB-200 3.3B with different decoding methods (x-axis).

Table 10. BLEU scores on the FLORES+ devtest for several student models and the teacher. Underlined results are those that show no statistically significant difference compared to $\mathcal{D}^1_{\text{BS}}$. Bolded results are those that show no statistically significant difference compared to $\mathcal{D}^{10}_{\text{BS}}$.

	eng-swh	eng-ibo	eng-bam	swh-eng	ibo-eng	bam-eng	bam-swh
NLLB 1.3B	33.1	16.1	6.8	42.9	30.7	17.8	11.6
Word-level $\mathcal{D}^1_{\text{BS}}$	28.9	14.2	0.0	24.8	10.8	0.1	0.0
100k $\mathcal{D}^1_{\text{BS}}$	26.2	14.1	4.7	22.9	10.0	5.8	2.1
100k $\mathcal{D}^{10}_{\text{BS}}$	30.4	**<u>15.6</u>**	0.3	27.9	12.1	8.7	1.2
100k $\mathcal{D}^{10}_{\text{DBS}}$	**30.2**	**<u>15.9</u>**	6.3	28.4	**12.4**	**8.3**	6.1
100k $\mathcal{D}^{10}_{\text{top-}p}$	31.1	**<u>16.0</u>**	6.6	29.4	13.4	9.9	7.7
100k $\mathcal{D}^{10}_{\text{top-}k}$	**30.9**	**<u>15.8</u>**	6.3	29.1	13.4	9.7	7.5
100k $\mathcal{D}^{10}_{\text{MBR}}$	30.8	16.1	7.1	28.9	12.8	9.0	6.8
500k $\mathcal{D}^1_{\text{BS}}$	31.5	15.4	6.3	31.3	14.2	–	–
500k $\mathcal{D}^{10}_{\text{BS}}$	33.5	15.1	6.4	32.7	15.5	–	–
500k $\mathcal{D}^{10}_{\text{DBS}}$	33.2	16.9	6.6	34.0	16.9	–	–
500k $\mathcal{D}^{10}_{\text{top-}p}$	33.9	16.9	6.9	33.3	16.2	–	–
500k $\mathcal{D}^{10}_{\text{top-}k}$	33.4	16.0	6.7	33.6	17.1	–	–
1M $\mathcal{D}^1_{\text{BS}}$	33.5	16.5	6.5	34.0	–	–	–
1M $\mathcal{D}^{10}_{\text{BS}}$	34.1	17.0	6.8	34.5	–	–	–
1M $\mathcal{D}^{10}_{\text{DBS}}$	34.0	15.7	6.6	35.6	–	–	–
1M $\mathcal{D}^{10}_{\text{top-}p}$	33.8	16.6	7.1	34.1	–	–	–
1M $\mathcal{D}^{10}_{\text{top-}k}$	33.7	16.7	7.1	34.4	–	–	–

Table 11. chrF++ scores on the FLORES+ devtest for several student models and the teacher model.

	eng-swh	eng-ibo	eng-bam	swh-eng	ibo-eng	bam-eng	bam-swh
NLLB 1.3B	59.2	41.0	30.9	63.5	52.6	38.6	35.6
Word-level $\mathcal{D}^1_{\text{BS}}$	55.4	38.4	4.8	48.9	33.5	6.3	5.1
100k $\mathcal{D}^1_{\text{BS}}$	52.4	37.6	27.8	47.5	32.7	25.5	8.7
100k $\mathcal{D}^{10}_{\text{BS}}$	56.9	39.6	12.6	50.7	35.5	29.1	20.4
100k $\mathcal{D}^{10}_{\text{DBS}}$	56.6	40.0	28.9	52.3	36.0	29.5	28.9
100k $\mathcal{D}^{10}_{\text{top-}p}$	57.4	39.9	30.1	52.7	36.7	30.6	30.9
100k $\mathcal{D}^{10}_{\text{top-}k}$	56.9	39.9	29.4	52.7	36.7	30.6	31.0
100k $\mathcal{D}^{10}_{\text{MBR}}$	57.7	41.2	31.3	53.3	36.6	31.5	32.3
500k $\mathcal{D}^1_{\text{BS}}$	57.8	40.0	30.4	54.6	37.3	-	-
500k $\mathcal{D}^{10}_{\text{BS}}$	59.2	40.1	30.7	55.7	39.2	-	-
500k $\mathcal{D}^{10}_{\text{DBS}}$	59.2	41.1	29.8	56.5	40.7	-	-
500k $\mathcal{D}^{10}_{\textbf{top-}p}$	59.4	41.2	30.9	56.2	39.7	-	-
500k $\mathcal{D}^{10}_{\textbf{top-}k}$	59.2	40.5	30.9	56.7	39.8	-	-
1M $\mathcal{D}^1_{\text{BS}}$	57.8	41.5	30.5	55.8	-	-	-
1M $\mathcal{D}^{10}_{\text{BS}}$	59.7	41.6	31.0	57.0	-	-	-
1M $\mathcal{D}^{10}_{\text{DBS}}$	59.6	40.4	30.2	57.9	-	-	-
1M $\mathcal{D}^{10}_{\textbf{top-}p}$	59.3	41.0	31.0	56.8	-	-	-
1M $\mathcal{D}^{10}_{\textbf{top-}k}$	59.4	41.1	31.0	57.1	-	-	-

Table 12. chrF++ scores on the FLORES+ devtest for student models trained with beam search and top-p corpora with different numbers of hypotheses across data scales.

	source: 100k			source: 1M	
	$\mathcal{D}^1_{\text{BS}}$	$\mathcal{D}^{10}_{\text{top-}p}$	$\mathcal{D}^{100}_{\text{top-}p}$	$\mathcal{D}^1_{\text{BS}}$	$\mathcal{D}^{10}_{\text{top-}p}$
eng-swh	52.4	57.4	58.7	57.8	59.3
eng-ibo	37.6	39.9	42.17	41.5	41.0
eng-bam	27.8	30.1	31.6	30.5	31.0
swh-eng	47.5	52.7	51.34	55.8	56.8
ibo-eng	32.7	36.7	36.27	–	–
bam-eng	25.5	30.6	30.46	–	–