UvA-MT's Participation in the WMT23 General Translation Shared Task

Di Wu* **Shaomu Tan*** **David Stap** **Ali Araabi** **Christof Monz**
Language Technology Lab
University of Amsterdam
{d.wu, s.tan, d.stap, a.araabi, c.monz}@uva.nl

Abstract

This paper describes the UvA-MT's submission to the WMT 2023 shared task on general machine translation. We participate in the constrained track in two directions: English ↔ Hebrew. In this competition, we show that by using one model to handle bidirectional tasks, as a minimal setting of Multilingual Machine Translation (MMT), it is possible to achieve comparable results with that of traditional bilingual translation for both directions. By including effective strategies, like back-translation, re-parameterized embedding table, and task-oriented fine-tuning, we obtained competitive final results in the automatic evaluation for both English → Hebrew and Hebrew → English directions.

1 Introduction

Multilingual Machine Translation (MMT) (Johnson et al., 2017) has attracted a lot of attention in recent years because of 1) its high-level efficiency (multiple translation directions within a single model) and 2) the potential for knowledge transfer, especially for low-resource or even unseen directions. In MMT systems, only one additional tag is introduced to indicate the translation direction, compared to the conventional encoder-decoder architecture. In this competition, we explore MMT with a minimal setting, i.e., using one model for bidirectional translations simultaneously.

We leverage all the official parallel data and a substantial portion of the monolingual data generously provided by the WMT23 organizer, as elaborated in Section 2. To elevate the quality of our parallel data, we implemented a comprehensive three-step cleaning procedure. Additionally, for monolingual data, we further trained an n-gram language model to filter out low-quality sentences, with the goal of generating synthetic data as elaborated in Section 4.1.

The backbone of our system is based on a standard transformer (Vaswani et al., 2017). Additionally, we build a re-parameterized embedding table (Wu and Monz, 2023) (see Section 3.2) to enhance the representational word similarity between English and Hebrew, targeting better knowledge transfer for multilingual translation.

The final system involves three stages of training: 1) **Pretraining with synthetic data** (see Section-4.1), where we leverage back-translation (Sennrich et al., 2016) to produce synthetic data, and finally add them as additional data within new translation directions in our MMT system to conduct pretraining. 2) **Training without synthetic data** (see Section 4.2), where we discard the additional synthetic data and further train our system using real bitext only. 3) **Fine-tuning with task-related data** (see Section 4.3), where we copy and fine-tune our system using English → Hebrew and Hebrew → English data for each track respectively. We observe evident improvements for stage 1 and stage 2, while surprisingly performance drops for stage 3.

We report our results, including the offline evaluation and the final online evaluation, in Section 5. Our *constrained* system showed comparable performance to *unconstrained* systems, and outperformed the second-place constrained submission with +10 BLEU.

2 Data

In this section, we provide an overview of our data sources and the data cleaning procedures applied to our English-Hebrew translation task. We utilize both parallel and monolingual data sets provided by the organizers for training our translation systems.

2.1 Parallel Data

We make use of all the available data from the constrained track of the shared task for English-

[*] Equal contribution.

Proceedings of the Eighth Conference on Machine Translation (WMT), pages 175–180
December 6–7, 2023. ©2023 Association for Computational Linguistics

Hebrew translation. To enhance the quality of our parallel data, we undergo a thorough preprocessing phase involving three key steps, as outlined below. All steps in the cleaning step 1 are executed using the Moses toolkit[*] (Koehn et al., 2007). Consequently, we reduced the size of the raw bitext data from 70 million to 34 million sentences after completing the three steps of the cleaning process:

- Cleaning Step 1
  - Deescaping special characters in XML.
  - Removing non-printable characters.
  - Normalizing punctuation and tokenizing sentences using Moses.
- Cleaning Step 2
  - Filtering out sentences longer than 256 tokens.
  - Eliminating sentences where over 75% of the words on both the source and target sides are identical.
  - Removing sentences with a source-to-target token ratio exceeding 1.5.
  - Eliminating duplicate sentences.
- Cleaning Step 3
  - Removing off-target sentences using the FastText Language identification tool (Joulin et al., 2016).
  - Excluding sentences exhibiting one-to-many or many-to-one mappings, for example, a single source sentence having multiple different target sentences.

Furthermore, we sampled 10 million parallel sentences to learn a 32k joint unigram (Kudo, 2018) model-based subword vocabulary using SentencePiece (Kudo and Richardson, 2018), which we then utilized across all our models, including the n-gram KenLM model discussed in the next section. However, we encountered a situation where certain emoji tokens were not included in our vocabulary. As a result, we integrated an additional post-processing step in Section 4.4 to address this issue.

2.2 Monolingual Data

To enhance our translation systems further, we incorporate monolingual data to produce synthetic data through back-translation. For our monolingual data, we primarily rely on the official English data provided by the organizers. Note that we did not use any Hebrew monolingual data since it is limited (only 1 million sentences). We combine three official English monolingual datasets: News Discussions 2019, Leipzig News Corpora 2020, and News Crawl (2007-2022) to construct our raw monolingual dataset. Following this, we apply the same Cleaning Step 1 procedure as detailed in the Parallel Data section to preprocess the monolingual data, and this results in 373 million sentences.

Considering the low quality of monolingual data, we additionally filter them by training an n-gram language model, i.e., KenLM (Heafield, 2011), and eliminate the sentences below an LM score threshold. The training data of KenLM is all of the test data in English, including our offline test data Flores, and the official test dataset. We train KenLM at the subword level, where we use the same unigram model (trained upon original bilingual data) to split the training data of KenLM. Then, we use it to score all of the monolingual data. To establish a filtering threshold, we randomly selected 1,000 sentences and labeled them as positive or negative based on criteria such as fluency, naturalness (e.g., avoiding strings of numbers), and relevance to the domain mentioned for WMT23 test datasets. Finally, we chose a threshold that could filter 70% bad cases within the 1,000 sentences, with the cost of monolingual 30% data, resulting in around 250M total sentences.

Lastly, considering the limitation of the computational resource, we sample 32M monolingual sentences (at the same level as the bilingual dataset) from the filtered dataset.

3 Systems

3.1 Backbone and Baseline

In this section, we outline the foundational architecture and adjustments made to our baseline systems. Our baseline model leverages English ↔ Hebrew translation directions by incorporating the target language token at the beginning of the encoder, denoted as "2he" and "2en". Our implementations are grounded in the Transformer architecture (Vaswani et al., 2017), leveraging the Fairseq toolkit (Ott et al., 2019).

For our baseline model, we utilize a 12-layer Transformer architecture (mT-large) with specific modifications, including pre-norm for both the en-

[*] https://github.com/moses-smt/mosesdecoder/

coder and decoder, and layer-norm for embedding. To enhance stability and performance, we tie the parameters of encoder embedding, decoder embedding, and decoder output. We also introduce dropout and attention dropout with a probability of 0.1, along with label smoothing at a rate of 0.1.

Similar to the approach described by Vaswani et al. (2017), we employ the Adam optimizer with a learning rate of 5e-4, implementing an inverse square root learning rate schedule with 4,000 warmup steps. We set the maximum number of tokens to 10,240, with gradient accumulation every 21 steps to facilitate large-batch training in Tang et al. (2021). We train all of our systems with 4 NVIDIA A6000 Gpus, and to expedite the training process, we conducted all experiments using half-precision training (FP16). Additionally, we save checkpoints every 2000 steps and implement early stopping based on perplexity, with a patience of 5 epochs.

3.2 Re-parameterized Embedding Table

Using a vocabulary that is shared across languages is common practice in MMT. In addition to its simple design, shared tokens play an important role in positive knowledge transfer, assuming that shared tokens refer to similar meanings across languages. This point has been demonstrated by previous works (Pires et al., 2019; Sun et al., 2022; Stap et al., 2023; Wu and Monz, 2023). To enhance word-level knowledge transfer, we follow (Wu and Monz, 2023) to implement a re-parameterized shared embedding table and equipped it with our backbone.

We leverage eflomal (Östling and Tiedemann, 2016) to train and extract subword-level alignments based on all of the bilingual data we used. Then, we build the priors of word equivalence (word alignments) into a graph and leverage GNN (Welling and Kipf, 2016) to re-parameterize the embedding table.

More specifically, For two words v_i and v_j in V, we define an alignment probability from v_j to v_i in corpus D as corresponding transfer ratios $g_{i,j}$ as follows:

$$g_{i,j} = \frac{c_{i,j}}{\sum_{k=1}^{|V|} c_{i,k}}, \tag{1}$$

where $c_{i,j}$ is the number of times both words are aligned with each other across D. The corresponding bilingual equivalence graph G can be induced by filling an adjacency matrix using $g_{i,j}$, G is applied within graph networks to re-parameterize the original embedding table as follows:

$$E' = \rho(EW_1 + GEW_2 + B). \tag{2}$$

To allow the message to pass over multiple hops, we stack multiple graph networks and calculate representations recursively as follows:

$$E^{h+1} = \rho(E^h W_1^h + GE^h W_2^h + B^h), \tag{3}$$

where h is the layer index, i.e., *hop*, and E^0 is equal to the original embedding table E. The last layer representation E^H is the final re-parameterized embedding table, for the maximum number of hops H, which is then used by the system just like any vanilla embedding table.

4 Experiments

We describe the training process of our system in three stages.

4.1 Pretraining with Synthetic Data

Back-translation plays an important role in leveraging monolingual data in machine translation. In this competition, we also apply it to produce synthetic data and include it in our first-stage training.

Specifically, we first train a base MMT model (backbone with re-parameterized embedding tables) using bilingual data. Then, we feed our monolingual English data to produce EN-HE synthetic bitext. Finally, we merge the original bilingual data with the synthetic data together to pre-train our MMT system. We follow Fan et al. (2021) and add an additional language tag "2syn" to differentiate between synthetic and original Hebrew data. Note that, although normally original data (here, it is EN) is used as target side data after back translation, we use synthetic data for both directions.

4.2 Training without Synthetic Data

Considering that the synthetic data may differ from the original bilingual data in terms of data quality, domain difference, and diversity, in the second-stage training, we encourage our system to skew towards the original bilingual distribution. We achieve this by discarding the synthetic data directly and continuing training upon the first-stage system as a kind of full parameter finetuning.

Strategy	Sampled Data		Full Data	
	EN→HE	HE→EN	EN→HE	HE→EN
Bilingual Baseline	24.6	31.1	34.1	46.0
MMT Baseline	24.7	31.6	34.1	45.8
MMT + GM 1-hop	**26.2**	32.3	**34.3**	**46.2**
MMT + GM 2-hop	25.5	**32.7**	-	-

Table 1: Offline evaluation results on sampled and full training data. For sampled data (2M), the backbone is Transformer Base, while for full data (34M) the backbone is Transformer Large as we describe in Section 3.1. MMT + GM means that we equip graph-based re-parameterized embedding tables for our MMT baseline, and hop means how many graph network layers are involved. The best BLEU scores in each column are written in bold.

Strategy	Offline		Online	
	EN→HE	HE→EN	EN→HE	HE→EN
MMT Baseline	34.1	45.8	33.3	50.3
MMT + GM 1-hop	34.3	46.2	33.6	50.7
MMT + GM 1-hop + Stage-1	**35.4**	46.8	**35.0**	50.1
MMT + GM 1-hop + Stage-1,2	34.1	**47.4**	**35.0**	**51.0**
MMT + GM 1-hop + Stage-1,2,3	33.3	44.3	33.3	48.0

Table 2: Final results of three stages training. The best BLEU scores in each column are written in bold.

4.3 Finetuning on Task-Specific Data

Lastly, to encourage the system to focus on one certain language direction, we further fine-tune direction-specific data on the second-stage system. Note, the direction-specific data here, i.e., EN → HE and HE → EN are both from the original bilingual data. The effectiveness is also demonstrated by Ding et al. (2021); Zan et al. (2022) for bilingual translation.

In short, in this three-stage training process, we gradually narrow down the data distribution to focus on task-specific real data.

4.4 Post-Processing

We noticed that some emoji tokens in the official test set were not included in our vocabulary. Thus, we integrated an additional post-processing step to process them. Specifically, we escaped the emoji tokens to their Unicode string[*] before tokenizing and feeding them to our system to conduct inference, and then convert the Unicode string back for generated predictions.

4.5 Offline Evaluation

We used Flores-200 (Costa-jussà et al., 2022) to evaluate our strategies offline before submissions and Ntrex-128 (Federmann et al., 2022) as the validation set. We show the results in Table 1. Due to resource limitations, we sample 2M of bilingual data to verify whether there is a big performance gap between MMT and bilingual baseline. Meanwhile, we also chose the best hyperparameter for our re-parameterized embedding table, i.e., the hop number, based on the sampled dataset.

As shown in Table 1, on both sampled and full data, the MMT baseline achieves comparable results with bilingual counterparts. Especially for sampled data, it even outperforms 0.5 BLEU for into-English translation.

The model-equipped 1-hop re-parameterized embedding table demonstrates a notable improvement, yielding a 1.5 BLEU gain for the out-of-English direction and a 0.7 BLEU gain for the out-of- and into-English directions, on 2M datasets. It shows that the embedding re-parameterized method (Wu and Monz, 2023) also works for bilingual settings, which is not explored in the original paper. We did not observe evident gains for 2-hop compared with 1-hop on sampled data, hence, we only apply the 1-hop graph networks for the full data training. As shown in the table, the results are consistent with that of small data, where MMT with 1-hop graph networks achieve better performance than MMT baseline.

As above, we chose MMT with the 1-hop setting as our architecture and conducted our three stages of training as described in Section 4.

[*] For example, the emoji of "Grinning Face with Open Eyes" will convert to a string "U+1F600".

5 Results

Table-2 shows our offline and online evaluation results according to each training stage described in Section 4. We still use Flores-200 to conduct offline evaluations. The online results are reported by WMT23 background BLEU evaluations. Stage-1, -2, and -3 refer to "Pretraining with Synthetic Data", "Training without Synthetic Data", and "Finetuning on Task-Specific Data" respectively.

The results of online and offline evaluations are quite consistent. Both of them achieve best results when training with stage 1 and stage 2. It shows that by step-by-step narrowing training data from mixing with synthetic data to real data distribution, we can further boost our MMT system's performance. However, when we further conduct fine-tuning on direction-specific data, i.e., applying stage 3, there is an evident performance drop. It seems that tuning in a specific direction upon MMT may not be a good practice, at least when the training data are a subset of that for MMT. We leave this point for future exploration.

Our final system achieves 35.0 and 51.0 in EN→HE and HE→EN direction respectively, which are both in the first place for constrained tracks.

6 Conclusion

In this competition, we show that: 1) It is possible to achieve comparable results with conventional bilingual translation by using MMT training fashion to handle two dual translation directions. 2) Previous embedding re-parameterized method (Wu and Monz, 2023) also works for bilingual translation, which is not verified in the original paper. However, when training data scales up to 30+M level, the improvements become marginal. 3) By step-by-step narrowing training data (especially for stage-1 and stage-2) from mixing with synthetic data to real data distribution, we successfully boost the final performance, even in a quite high-resource scenario (30+M).

Acknowledgments

This research was funded in part by the Netherlands Organization for Scientific Research (NWO) under project number VI.C.192.080.

References

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

Liang Ding, Di Wu, and Dacheng Tao. 2021. Improving neural machine translation by bidirectional training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3278–3284, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, et al. 2021. Beyond english-centric multilingual machine translation. *The Journal of Machine Learning Research*, 22(1):4839–4886.

Christian Federmann, Tom Kocmi, and Ying Xin. 2022. Ntrex-128–news test references for mt evaluation of 128 languages. In *Proceedings of the First Workshop on Scaling Up Multilingual Evaluation*, pages 21–24.

Kenneth Heafield. 2011. Kenlm: Faster and smaller language model queries. In *Proceedings of the sixth workshop on statistical machine translation*, pages 187–197.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2017. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351.

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

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75.

Taku Kudo and John Richardson. 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*, pages 66–71.

Robert Östling and Jörg Tiedemann. 2016. Efficient word alignment with markov chain monte carlo. *The Prague Bulletin of Mathematical Linguistics*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

David Stap, Vlad Niculae, and Christof Monz. 2023. Viewing knowledge transfer in multilingual machine translation through a representational lens. *arXiv preprint arXiv:2305.11550*.

Simeng Sun, Angela Fan, James Cross, Vishrav Chaudhary, Chau Tran, Philipp Koehn, and Francisco Guzmán. 2022. Alternative input signals ease transfer in multilingual machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5291–5305, Dublin, Ireland. Association for Computational Linguistics.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2021. Multilingual translation from denoising pre-training. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3450–3466.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Max Welling and Thomas N Kipf. 2016. Semi-supervised classification with graph convolutional networks. In *J. International Conference on Learning Representations (ICLR 2017)*.

Di Wu and Christof Monz. 2023. Beyond shared vocabulary: Increasing representational word similarities across languages for multilingual machine translation. *arXiv preprint arXiv:2305.14189*.

Changtong Zan, Keqin Peng, Liang Ding, Baopu Qiu, Boan Liu, Shwai He, Qingyu Lu, Zheng Zhang, Chuang Liu, Weifeng Liu, Yibing Zhan, and Dacheng Tao. 2022. Vega-MT: The JD explore academy machine translation system for WMT22. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 411–422, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.