Improving Embedding Transfer for Low-Resource Machine Translation

Van-Hien Tran tran.vanhien@nict.go.jp
Chenchen Ding chenchen.ding@nict.go.jp
Hideki Tanaka hideki.tanaka@nict.go.jp
Masao Utiyama mutiyama@nict.go.jp
National Institute of Information and Communications Technology,
3-5 Hikaridai, Seika-cho, Soraku-gun, Kyoto, 619-0289, Japan

Abstract

Low-resource machine translation (LRMT) poses a substantial challenge due to the scarcity of parallel training data. This paper introduces a new method to improve the transfer of the embedding layer from the Parent model to the Child model in LRMT, utilizing trained token embeddings in the Parent model's high-resource vocabulary. Our approach involves projecting all tokens into a shared semantic space and measuring the semantic similarity between tokens in the low-resource and high-resource languages. These measures are then utilized to initialize token representations in the Child model's low-resource vocabulary. We evaluated our approach on three benchmark datasets of low-resource language pairs: Myanmar-English, Indonesian-English, and Turkish-English. The experimental results demonstrate that our method outperforms previous methods regarding translation quality. Additionally, our approach is computationally efficient, leading to reduced training time compared to prior works.

1 Introduction

Neural machine translation (NMT) systems have revolutionized the field of natural language processing (NLP), offering remarkable performance gains. Extensive studies (Sutskever et al., 2014; Bahdanau et al., 2015; Vaswani et al., 2017) have consistently demonstrated that NMT systems trained on substantial parallel corpora yield exceptional results. However, low-resource machine translation (LRMT) remains a significant obstacle in the NLP domain. The need for more training data presents a formidable hurdle in training accurate and robust machine translation systems, particularly for languages with limited resources. Unfortunately, many languages fall into this category and require increased availability of parallel corpora for practical machine translation training. As a result, researchers have dedicated their efforts to developing innovative methods to enhance machine translation quality for low-resource languages.

The challenge of LRMT has sparked considerable research interest in recent years (Aji et al., 2020; Xu and Hong, 2022; Li et al., 2022), leading to innovative approaches to tackle the issue. Transfer learning, unsupervised learning, and active learning techniques are some of the methods that have been explored, all showing promising results in enhancing translation quality for low-resource languages. In particular, transfer learning has emerged as a highly effective and straightforward approach for the LRMT task. It has significantly improved translation model performance by leveraging pre-trained high-resource language models. In essence,

Proceedings of Machine Translation Summit XIX, Vol. 1: Research Track, pages 123–134
September 4–8, 2023, Macau SAR, China.
©2023 The authors. This article is licensed under a Creative Commons Attribution 4.0 International License (CC BY 4.0).

this approach involves transferring knowledge from a high-resource parent model to a low-resource child model, resulting in a remarkable enhancement in the latter's efficacy. Overall, transfer learning is a highly efficacious and practical technique that holds immense potential in improving machine translation for low-resource languages.

The Parent-Child transfer learning framework, initially introduced by Zoph et al. (2016), has been a vital breakthrough in improving the LRMT task. Several studies have optimized the technique's effectiveness by transferring additional information from the parent model's embedding layer through different means. For instance, Kocmi and Bojar (2018) and Gheini and May (2019) proposed using a shared vocabulary, while Kim et al. (2019) suggested a cross-lingual token mapping method. Aji et al. (2020) emphasized the importance of aligning the vocabulary before embedding transfer, which led to notable improvements. Recently, Xu and Hong (2022) have taken this work a step further by duplicating aligned sub-word embeddings, improving transferable Parent-Child NMT. These techniques have improved the transfer learning effect and enhanced the LRMT task's performance.

This study introduces a new method to enhance the parent-child transfer framework by transferring the embedding layer from the parent to child models. The previous work by Aji et al. (2020) only partially transferred word embeddings from the parent model for words with identical forms. Meanwhile, Xu and Hong (2022) used both aligned multilingual and morphologically-identical sub-words for embedding transfer, which may lead to inconsistencies. Our new approach overcomes the existing limitations in earlier works (Aji et al., 2020; Xu and Hong, 2022) and tends to optimize the embedding transfer process. Specifically, it involves projecting tokens from parent and child models into a shared semantic space, then computing their semantic similarity measure. This way, each token in the embedding layer of the child model can be represented using the relevant pre-trained embeddings of the related tokens in the parent model, leading to enhanced embedding transfer accuracy.

We validated our approach by conducting comprehensive experiments on three benchmark datasets, Myanmar-English, Indonesian-English, and Turkish-English. The results from the experiments showed that our approach not only outperformed the existing state-of-the-art methods but also reduced the training effort, thus proving its effectiveness and efficiency. In short, our contributions revolve around two key points: introducing a new approach to transferring token embeddings from the Parent to Child model by measuring their semantic similarity within the same semantic space and validating its effectiveness and efficiency through meticulous experiments on benchmark datasets.

2 Related Work

Transfer learning has been proven effective for NMT under low-resource conditions. Zoph et al. (2016) pioneered the transferable Parent-Child framework, significantly improving BLEU scores across various low-resource languages. Their method involved training a high-resource language pair as a parent model and using the trained weights to initialize a child model. The Child model was then trained on a limited parallel corpus of a low-resource language pair. However, this approach overlooked a significant challenge: the vocabulary mismatch between parent and child models. Subsequent research endeavors have tackled this challenge with determination and perseverance.

Kocmi and Bojar (2018) advocated for using a shared vocabulary between Parent and Child models, as it has proven advantageous. However, it comes with a catch: the Parent model needs prior knowledge of the Child's language during training. This can be limiting and may only sometimes be feasible. To overcome this obstacle, Gheini and May (2019) proposed a universal vocabulary strategy for transfer learning. This approach involves simultaneously training sub-word tokens across multiple languages and using Romanisation for languages with

non-Latin scripts. While this method is promising, it may only work for some languages in real-world scenarios. Additionally, it could result in overly aggressive and sub-optimal subword segmentation for unseen languages.

In another direction, several studies (Kim et al., 2018; Lample et al., 2018; Artetxe et al., 2018; Kim et al., 2019) have utilized bilingual word embedding alignment as an approach to initialize the embedding layer. Kim et al. (2018) proposed a simple yet effective method that improves word-by-word translation of cross-lingual embeddings using only monolingual corpora without resorting to back-translation. Lample et al. (2018), on the other hand, utilized careful parameter initialization, denoising effects of language models, and automatic generation of parallel data through iterative back-translation. Kim et al. (2019) demonstrated effective techniques for transferring a pre-trained NMT model to a new, unrelated language that lacks shared vocabularies. Their approach involved mitigating vocabulary mismatches through cross-lingual word embeddings, training a more language-agnostic encoder through artificial noise injection, and generating synthetic data from pretraining data without back-translation.

Recently, Aji et al. (2020) conducted a study to investigate the effects of various strategies for transferring token embeddings between Parent and Child models. The study found that aligning the vocabulary before transferring the embeddings is essential for practical performance improvements. However, their approach only involved partial token matching, where morphologically-identical tokens were duplicated embeddings while the rest were randomly assigned embeddings. Subsequently, Xu and Hong (2022) attempted to address this limitation by copying token embeddings among aligned multilingual tokens, enabling the transfer of embeddings for morphologically-identical and elaborately-aligned tokens. However, duplicating embeddings for the same token across different languages may only sometimes be appropriate as it could result in different meanings (Vernikos and Popescu-Belis, 2021). Furthermore, using distinct techniques to transfer embeddings for morphologically-similar and morphologically-dissimilar token types may lead to inconsistency.

Therefore, this paper presents a unified and comprehensive approach to transfer embeddings by projecting all tokens in the same semantic space and considering their relationships. By doing so, we can overcome the existing limitations of previous approaches and ensure consistency in transferring embeddings for morphologically-similar and morphologically-dissimilar token types.

3 Our Approach

3.1 Basic Parent-Child Transfer Framework

Following the research conducted by Aji et al. (2020) and Xu and Hong (2022), we also construct NMT models utilizing the 12-layer base transformer architecture proposed by Vaswani et al. (2017). As elucidated by Vaswani et al. (2017), this architecture composes the first six layers in the encoder and the subsequent six layers in the decoder, forming a total of 12 layers. The encoder is often coupled with a trainable embedding layer, which retains a fixed bilingual vocabulary and trainable subword embeddings. Also, each embedding is designated as a 512-dimensional real-valued vector.

Taking inspiration from the pioneering work of Zoph et al. (2016), we conduct Parent-Child transfer learning. For the Parent model, we have selected an off-the-self transformer-based NMT model[1], similar to the approach taken by Xu and Hong (2022), which was adequately trained on a substantial amount of De→En (German→English) parallel sentence pairs

[1] https://github.com/Helsinki-NLP/OPUS-MT-train/blob/master/models/de-en/README.md

(approximately 351.7 million pairs) from the OPUS dataset[2] (Tiedemann, 2012). We treat this NMT model as the Parent. Meanwhile, the Child model also uses the 12-layer base transformer architecture like the Parent, and it will be trained on the low-resource X→En language pairs after completing the transfer process. Specifically, we first transfer all inner parameters (non-embedding) of the 12-layer transformers from the Parent to the Child. Toward embedding transfer, it is not straightforward since different languages have distinct vocabularies. Thus, we make an effort to perform the embedding transfer more effectively.

3.2 Embedding Transfer

Let V_h denote the high-resource bilingual vocabulary (e.g., the aforementioned De-En) in the Parent model with the tokenizer T_h and the corresponding token embeddings $\mathbf{E}_h \in \mathbb{R}^{|V_h| \times d}$. Specifically, $\mathbf{E}_h$ maps each token v in the vocabulary V_h to its vector representation $\boldsymbol{v} \in \mathbb{R}^d$ with the hidden size of d (e.g., $d = 512$).

To handle the low-resource X→En language pair for the Child model, we employ two separate vocabularies for the source language X and the target language English. For the English target language side, we directly reuse the vocabulary V_h and its corresponding token embeddings $\mathbf{E}_h$. However, for the X source language X side, we use a low-resource vocabulary V_l with a tokenizer T_l and corresponding token embeddings $\mathbf{E}_l \in \mathbb{R}^{|V_l| \times d}$. Our primary objective is to initialize the token embeddings $\mathbf{E}_l$ effectively using the trained token embeddings $\mathbf{E}_h$. To achieve this, we follow these steps.

Train Subword Tokenizer Following Xu and Hong (2022), we train a subword tokenizer, denoted as T_l, for the low-resource source language X in the Child model (e.g., X is Myanmar, Indonesian, or Turkish). Specifically, we use the unigram model of SentencePiece[3] to train T_l. We collect monolingual plain texts from Wikipedia dumps[4] and use the toolkit Wikiextractor[5] to extract them from the semi-structured data. The statistics of the training data are presented in Table 1.

X	**Doc.**	**Sent.**	**Token**
Myanmar (My)	113K	1.1M	17.4M
Indonesian (Id)	1.1M	8.3M	156.2M
Turkish (Tr)	705K	5.8M	128.2M

Table 1: Statistics of the monolingual Wikipedia data for each low-resource language X.

We uniformly set the low-resource vocabulary size $|V_l|$ in the Child model to 50K when training the tokenizer T_l. Meanwhile, the size of the mixed De-En high-resource vocabulary $|V_h|$ in the trained Parent NMT model is 58K. For the training and inference phases of the Child model with the low-resource language pair X→En, we use T_l to tokenize only the source language X while T_h to tokenize the target language English.

Obtain Token Representation To accurately measure the semantic similarity between the vocabularies of V_h and V_l, it is crucial to obtain the representation of each token first. This important step allows us to thoroughly analyze and evaluate the tokens in the vocabulary sets, giving us a deeper understanding of their interconnectedness. This understanding then enhances our knowledge of the relationship between the two vocabularies and enables us to unlock their

[2] https://opus.nlpl.eu

[3] https://github.com/google/sentencepiece

[4] https://dumps.wikimedia.org/

[5] https://github.com/attardi/wikiextractor

full potential, creating more meaningful connections. Thus, obtaining token representation is a top priority in understanding semantic similarity comprehensively.

Following the work by Vernikos and Popescu-Belis (2021), we obtain token representations in V_l by utilizing the corresponding pre-trained FastText embeddings[6] for the low-resource language X. In particular, regarding tokens that are subwords in V_l, the FastText embeddings of the language X also create the corresponding representations by decomposing each subword into n-grams of characters and taking the average of the embeddings of all occurring these n-grams. It is equivalent to how creating embeddings for out-of-vocabulary words is introduced in FastText (Bojanowski et al., 2017). Similarly, we also obtain token representations in V_h using the pre-trained FastText embeddings for English.

Find A Rotation Matrix After utilizing the static pre-trained FastText embeddings in the previous step, we obtained representation vectors for the tokens in both V_h and V_l. However, it is essential to note that these token embeddings are located in two separate semantic spaces; one for the English language and the other for the X language. To properly analyze their semantic relationship, unifying these token embeddings into a shared semantic space is necessary. To achieve this, we need to find a rotation matrix.

In the quest for accurate estimations of semantic similarities between tokens, the use of optimal rotation matrices can be highly effective. Let $\mathbf{F}_h \in \mathbb{R}^{|V_h| \times 300}$ and $\mathbf{F}_l \in \mathbb{R}^{|V_l| \times 300}$ denote the obtained embedding matrices of the tokens in V_h and V_l by using FastText, respectively, after which we strive to find the optimal rotation matrix $\mathbf{M}$ that transforms $\mathbf{F}_h$ onto $\mathbf{F}_l$. This transformation paves the way for calculating semantic similarities between tokens in the same semantic space.

To achieve this matrix, the first step is to acquire the given train set of the low-resource (X-En) parallel pairs, which we then run through Eflomal[7], a powerful tool that enables us to acquire a bilingual word alignment list. Armed with the obtained X-En alignment list, we proceed to get two corresponding embedding matrices, one containing English word embeddings and the other containing embeddings for words in the X language, using the static pre-trained FastText. Following this, we treat the obtained bilingual alignment list as the supervised signal and leverage the Orthogonal Procrustes method (Schönemann, 1966; Artetxe et al., 2016), a highly effective learning method, to derive the rotation matrix $\mathbf{M}$.

Initialize the Token Embeddings $\mathbf{E}_l$ Using the trained matrix $\mathbf{M}$, we project $\mathbf{F}_h$ to the semantic space of $\mathbf{F}_l$. Through this transformation, we can easily calculate the cosine similarity of each token within V_l to each token in V_h. By doing so, we can establish meaningful connections between these two semantic spaces, allowing for heightened understanding. The cosine similarity between two tokens x and y is defined as follows:

$$sim(x, y) = \frac{\boldsymbol{x}\boldsymbol{y}^T}{\|\boldsymbol{x}\| * \|\boldsymbol{y}\|}$$

, where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the vectors of the tokens $x \in V_l$ and $y \in V_h$, respectively, in the shared semantic space of $\mathbf{F}_l$; $\|\boldsymbol{x}\|$ and $\|\boldsymbol{y}\|$ are the Euclidean norms of the two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively.

Through the above formulation, we can achieve results by attaining the cosine similarity of every individual token in V_l with all tokens in V_h. These similarities are subsequently ranked in descending order, creating a comprehensive and insightful view of our data. To transform this data into even more valuable insights, we consider two methods for creating embedding vectors for each token x in V_l in $\mathbf{E}_l$.

The first method is called the Top-1 method. For each token $x \in V_l$, we only keep the

[6] https://fasttext.cc/docs/en/crawl-vectors.html

[7] https://github.com/robertostling/eflomal

single token $y \in V_h$ with the highest cosine similarity to x and duplicate its embedding in $\mathbf{E}_h$ to the token embedding of x in $\mathbf{E}_l$, resulting in a highly effective and intuitive system.

On the other hand, the second method, known as the Softmax method, is equally compelling. For each token $x \in V_l$, we create the corresponding set $\mathcal{S}_x$, including the K nearest tokens of V_h to the given token x. The Softmax function is then applied to these similarity measures, producing a highly weighted token embedding of x in $\mathbf{E}_l$ as follows:

$$\mathbf{E}_l(x) = \sum_{y \in \mathcal{S}_x} \frac{\exp\left(sim(x, y)\right)}{\sum_{y' \in \mathcal{S}_x} \exp\left(sim(x, y')\right)} \cdot \mathbf{E}_h(y)$$

Once we have acquired the token embeddings $\mathbf{E}_l$ for the Child model's vocabulary, it is time to train the Child model using the provided X-En low-resource parallel train set. Our careful embedding transfer is expected to improve the system's performance and decrease the training time for the model.

4 Experiments

4.1 Datasets and Evaluation Metric

Following the previous work by Xu and Hong (2022), we use the same benchmark datasets and similar experimental settings. Specifically, we evaluate the transferable NMT models for three different source languages, including Myanmar (My), Indonesian (Id), and Turkish (Tr). In addition, English is fixed as the target language.

We use three low-resource parallel datasets for training the Child NMT model, including Asian Language Treebank (ALT) (Ding et al., 2018), PAN Localization BPPT[8], and the corpus of WMT17 news translation task (Bojar et al., 2017). The statistics in the training, validation, and test sets are shown in Table 2. Also, we evaluate all the considered NMT models with SacreBLEU (Post, 2018).

4.2 Experimental Settings

As introduced in Section 3.1, we used an off-the-shelf NMT model as Parent whose state variables (i.e., hyperparameters and transformer parameters) and embedding layer are all set. This Parent NMT model was adequately trained on high-resource De→En (German→English) language pairs.

We adopt the following hyperparameters to transfer the embedding layer and train the Child NMT model. We set K nearest tokens to 15 in the Softmax technique for our embedding transfer method. Also, each source language was tokenized using SentencePiece (Kudo and Richardson, 2018) with a 50K vocabulary size. The training process was carried out with HuggingFace Transformers library (Wolf et al., 2020) using the Adam optimizer with 0.1 weight decay rate. The maximum sentence length was set to 128 and the batch size to 64 sentences.

[8] `http://www.panl10n.net/english/OutputsIndonesia2.htm`

Dataset	**Train.**	**Val.**	**Test**
My-En (ALT)	18K	1K	1K
Id-En (BPPT)	22K	1K	1K
Tr-En (WMT17)	207K	3K	3K

Table 2: Statistics for low-resource parallel datasets.

The learning rate was set to $5e-5$ and checkpoint frequency to 500 updates. For each model, we chose the checkpoint with the lowest perplexity on the validation set for testing.

4.3 Results and Analysis

In this section, we perform extensive experiments and analysis results to evaluate our approach for the low-resource NMT task.

Baseline Models We compare our approach to three previous Parent-Child (PC) transfer NMT models. Our model and all the baseline models duplicate non-embedding parameters from the same Parent model, which we introduced in Section 3.1. However, these models differ in how they transfer the embedding layer. The first baseline Child model is named Random-PC, in which the embedding layer is randomly initialized with a Gaussian distribution. Meanwhile, the second baseline Child model, called MI-PC, uses the embedding transfer method by Aji et al. (2020), which only transfers the embeddings of morphologically-identical tokens. The last baseline Child model, Mean-PC (Xu and Hong, 2022), extends Aji et al. (2020)'s work by leveraging embedding duplication between aligned sub-words.

Main Results Table 3 presents the test results of various PC transfer models on three benchmark datasets, utilizing the SentencePiece tokenizer. From the analysis, it is evident that the Random-PC model performs the worst among all the models. This is because it overlooks the embedding transfer from the Parent model and randomly initializes all token embeddings for the embedding layer. As a result, the Random-PC model fails to comprehend the meaning of low-resource tokens, particularly in the low-resource NMT scenario, where the training set is limited. Therefore, leveraging embedding transfer from the Parent to the Child model is crucial in enabling low-resource models to understand the meaning of tokens and improve translation quality.

Our approach has proven more effective than the Random-PC baseline model, exhibiting a stable increase in the BLEU score across all three benchmark datasets. We significantly improve 3.1 BLEU points on the Id-En set. Additionally, our method surpasses the state-of-the-art work by Xu and Hong (2022) and consistently improves results on all three low-resource datasets. The most notable improvement is observed in the Id-En dataset, with an increase of up to 1.1 BLEU scores. Our approach effectively transfers the embedding layer, enhancing system performance in the LRMT task.

In our approach, we have analyzed and compared two techniques, namely Top-1 and Softmax, which have been discussed in Section 3.2. As shown in Table 3, the Softmax technique brings the best performance, while the remaining technique results in performance degradation. One possible reason is that using a single token for embedding duplication in the Top-1 technique does not express fully and precisely the meaning of each token in the Child model's vocabulary, especially when tokens are subwords in different languages (i.e., between high-resource and low-resource languages). Therefore, aggregating and normalizing embeddings of

Model		My-En	Id-En	Tr-En
Random-PC		20.5	26.0	17.0
MI-PC (Aji et al., 2020)		21.0	27.5	17.6
Mean-PC (Xu and Hong, 2022)		22.5	28.0	18.1
Ours	Top-1	22.1	28.0	18.4
	Softmax	**23.3***	**29.1***	**19.0***

Table 3: Results using **SentencePiece** tokenizer. The symbol * denotes statistically significant ($p < 0.02$) improvement (Koehn, 2004), compared to the Mean-PC model.

Model		My-En	Id-En	Tr-En
Random-PC		20.2	24.5	16.5
MI-PC (Aji et al., 2020)		20.4	24.2	16.8
Mean-PC (Xu and Hong, 2022)		21.9	27.1	16.9
Ours	Top-1	22.4	27.8	18.0
	Softmax	**23.2**†	**28.5**†	**18.2**†

Table 4: Results using **BPE** tokenizer. The symbol † denotes statistically significant ($p < 0.02$) improvement (Koehn, 2004), compared to the Mean-PC model.

the top K nearest tokens via the Softmax technique helps to overcome the existing problem and create token representations more comprehensively and accurately.

We further check the effectiveness of all the PC transfer models when using BPE tokenizer (Sennrich et al., 2016) instead of SentencePiece tokenizer (Kudo and Richardson, 2018). Table 4 shows all models' experimental results. Compared to all remaining models, our approach performs best when using a BPE tokenizer. In particular, compared to the Random-PC baseline model, our model substantially improves the system performance by 3.0, 4.0, and 1.7 BLEU scores on the My-En, Id-En, and Tr-en benchmark datasets, respectively. Additionally, our model outperforms the state-of-the-art work by Xu and Hong (2022) by over 1.0 BLEU scores on all three datasets. In our approach, the Softmax technique performs better than the Top-1 technique when using a BPE tokenizer.

In summary, the experimental findings presented in Tables 3 and 4 provide strong evidence supporting the efficacy of our proposed method for transferring the embedding layer. Our approach demonstrates the potential to effectively enhance system performance in the low-resource NMT task, indicating the effectiveness of our method. Additionally, our findings suggest that the Softmax technique is a more suitable and practical approach for creating an effective embedding layer initialization for the transfer PC model, compared to the Top-1 technique.

Training Time It has been speculated that the initialization of token embeddings through embedding transfer not only enhances the BLEU score of the system but also has the potential to reduce the training time of the Child model. Therefore, we delved into this matter and conducted a comprehensive investigation of the training time for each model. We used mixed precision to train the Child NMT model to achieve optimal results. Furthermore, all experiments were conducted on a single Tesla V100-SXM2-32GB GPU. Our findings are reported in Table 5.

Our model with the Softmax technique consumes less time during the training phase than other models. In particular, in the case of the Tr-En dataset, the training duration is even shortened from 4.51 hours in the Random-PC model to 2.06 hours in our model. Besides, compared to the method by Xu and Hong (2022), the training time of our approach is also competitive or slightly better. These advantages come from avoiding redundant learning over token embed-

Model		My-En	Id-En	Tr-En
Random-PC		1.78	1.50	4.51
MI-PC (Aji et al., 2020)		1.64	1.26	4.35
Mean-PC (Xu and Hong, 2022)		1.09	1.06	2.19
Ours	Top-1	1.05	0.95	2.07
	Softmax	**0.95**	**0.85**	**2.06**

Table 5: The training time (in hour) of the different NMT models on three benchmark datasets.

dings once they are initialized well before starting the training phase.

To sum up, initializing a good embedding layer in the PC transfer models is vital in enhancing the system's effectiveness and efficiency. Our embedding transfer method helps initialize the embedding layer of the Child model productively, thereby improving the BLEU scores as shown in Tables 3 and 4 and decreasing the training time as shown in Table 5.

Impact of the Hyperparameter K As outlined in Section 3.2, our proposed approach utilizing the Softmax technique searches for the top K nearest tokens in the Parent model's vocabulary for each token in the Child model's vocabulary. This process is instrumental in creating the initialization embedding of the given token. It is necessary to understand how the hyperparameter K affects the embedding quality, which has a certain impact on the overall system performance. Therefore, we fine-tune K in [1, 5, 15, 30, 45, 60] to investigate how the value K affects to the quality translation. In the special case of $K = 1$, the Softmax technique becomes the Top-1 technique in our approach. All experimental results on three low-resource benchmark datasets are visually represented in Figure 1.

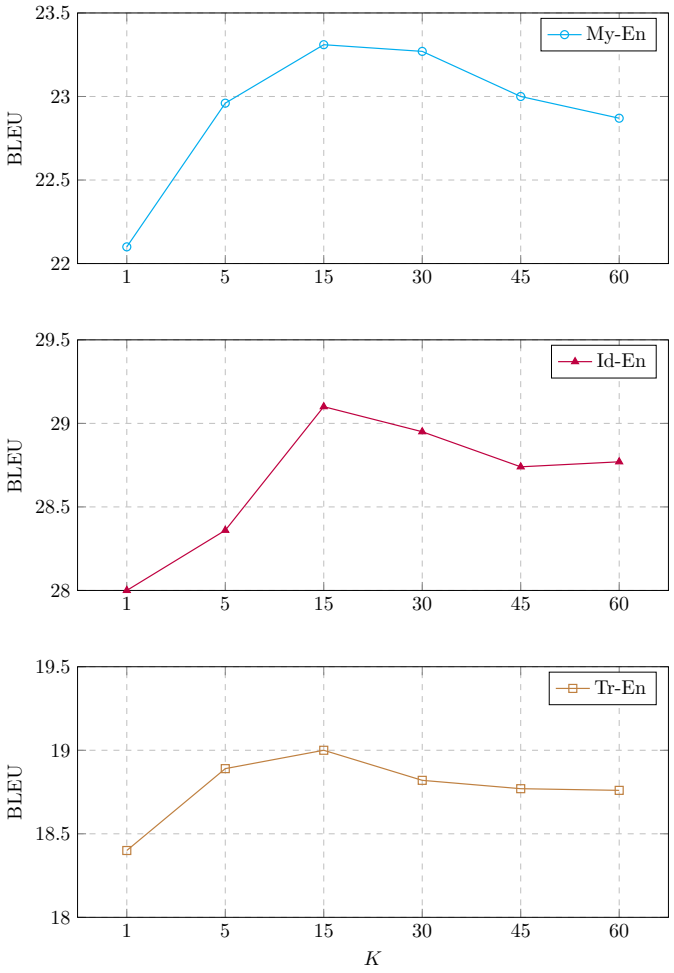

Figure 1: Impact of the Hyperparameter K to the performance of our model on the test set.

The importance of the hyperparameter K cannot be underestimated, as it plays an essential role in our approach to determining the quality of the embedding layer initialization and, ultimately, the overall system performance. The experimental findings demonstrate that a K value of 1 results in the lowest BLEU score across all three benchmark datasets compared to all other cases of $K > 1$. Meanwhile, our model outperforms all others when the K value is set to 15 across all three low-resource datasets. However, the system's performance deteriorates when K exceeds 15. Therefore, selecting the appropriate value of K is necessary for our approach since it affects achieving the most optimal token representation for the embedding layer of the Child model.

5 Conclusion

This paper introduced a new method to improve embedding transfer for the Child model in the LRMT task by leveraging trained token embeddings in the Parent model's high-resource vocabulary. By projecting all tokens of the Child and Parent models into a shared semantic space, it helps easily calculate the semantic similarity measure between tokens, thereby creating high-quality embeddings of the tokens in the Child model's low-resource vocabulary with the Softmax technique. Our approach is then thoroughly evaluated on the three benchmark low-resource datasets: Myanmar-English, Indonesian-English, and Turkish-English. The experimental results indicate that our method yields stable improvements in translation quality on all the datasets. Our approach is also computationally efficient, resulting in a reduction in training time consumption compared to baseline models. In future work, we will continue to enhance the embedding transfer technique since it is vital to improving the LRMT task in terms of effectiveness and efficiency.

References

Aji, A. F., Bogoychev, N., Heafield, K., and Sennrich, R. (2020). In neural machine translation, what does transfer learning transfer? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7701–7710, Online. Association for Computational Linguistics.

Artetxe, M., Labaka, G., and Agirre, E. (2016). Learning principled bilingual mappings of word embeddings while preserving monolingual invariance. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2289–2294, Austin, Texas. Association for Computational Linguistics.

Artetxe, M., Labaka, G., and Agirre, E. (2018). Unsupervised statistical machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3632–3642, Brussels, Belgium. Association for Computational Linguistics.

Bahdanau, D., Cho, K., and Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Bengio, Y. and LeCun, Y., editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Bojanowski, P., Grave, E., Joulin, A., and Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Bojar, O., Chatterjee, R., Federmann, C., Graham, Y., Haddow, B., Huang, S., Huck, M., Koehn, P., Liu, Q., Logacheva, V., Monz, C., Negri, M., Post, M., Rubino, R., Specia, L., and Turchi,

M. (2017). Findings of the 2017 conference on machine translation (WMT17). In *Proceedings of the Second Conference on Machine Translation*, pages 169–214, Copenhagen, Denmark. Association for Computational Linguistics.

Ding, C., Utiyama, M., and Sumita, E. (2018). Nova: A feasible and flexible annotation system for joint tokenization and part-of-speech tagging. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 18(2).

Gheini, M. and May, J. (2019). A universal parent model for low-resource neural machine translation transfer. *ArXiv*, abs/1909.06516.

Kim, Y., Gao, Y., and Ney, H. (2019). Effective cross-lingual transfer of neural machine translation models without shared vocabularies. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1246–1257, Florence, Italy. Association for Computational Linguistics.

Kim, Y., Geng, J., and Ney, H. (2018). Improving unsupervised word-by-word translation with language model and denoising autoencoder. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 862–868, Brussels, Belgium. Association for Computational Linguistics.

Kocmi, T. and Bojar, O. (2018). Trivial transfer learning for low-resource neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 244–252, Brussels, Belgium. Association for Computational Linguistics.

Koehn, P. (2004). Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 388–395, Barcelona, Spain. Association for Computational Linguistics.

Kudo, T. and Richardson, J. (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, Brussels, Belgium. Association for Computational Linguistics.

Lample, G., Ott, M., Conneau, A., Denoyer, L., and Ranzato, M. (2018). Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Li, Z., Liu, X., Wong, D. F., Chao, L. S., and Zhang, M. (2022). ConsistTL: Modeling consistency in transfer learning for low-resource neural machine translation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8383–8394, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Post, M. (2018). A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Schönemann, P. H. (1966). A generalized solution of the orthogonal procrustes problem. *Psychometrika*, 31:1–10.

Sennrich, R., Haddow, B., and Birch, A. (2016). Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Sutskever, I., Vinyals, O., and Le, Q. V. (2014). Sequence to sequence learning with neural networks. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc.

Tiedemann, J. (2012). Parallel data, tools and interfaces in opus. In *International Conference on Language Resources and Evaluation*.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. In Guyon, I., von Luxburg, U., Bengio, S., Wallach, H. M., Fergus, R., Vishwanathan, S. V. N., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Vernikos, G. and Popescu-Belis, A. (2021). Subword mapping and anchoring across languages. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2633–2647, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., and Rush, A. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Xu, M. and Hong, Y. (2022). Sub-word alignment is still useful: A vest-pocket method for enhancing low-resource machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 613–619, Dublin, Ireland. Association for Computational Linguistics.

Zoph, B., Yuret, D., May, J., and Knight, K. (2016). Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.