Probabilistic Bilingual Subword Segmentation with Latent Subword Alignment

Shoto Nishida[1] **Daiki Matsui**[1] **Takashi Ninomiya**[1] **Isao Goto**[1] **Akihiro Tamura**[2]
[1]Ehime University [2]Doshisha University
nishida@ai.cs.ehime-u.ac.jp matsui@ai.cs.ehime-u.ac.jp
ninomiya@cs.ehime-u.ac.jp goto.isao.fn@ehime-u.ac.jp
aktamura@mail.doshisha.ac.jp

Abstract

This study proposes a method for learning subword correspondences in parallel sentence pairs using the EM algorithm. Conventional neural machine translation typically employs subword segmentation models trained. However, since existing methods do not consider parallel relationships, inconsistencies in word segmentation between source and target languages may hinder translation model training. Our approach leverages direct modeling of subword correspondences in parallel corpora, thereby improving segmentation consistency across languages. Experiments across multiple machine translation tasks confirm that our proposed method improves translation accuracy for many tasks.

1 Introduction

Neural machine translation (NMT) relies on a predefined vocabulary, so its performance degrades when the source text contains low-frequency or unknown words during translation. To address this vocabulary problem, byte-pair encoding (BPE) (Sennrich et al., 2016) and subword segmentation based on unigram language models (Kudo, 2018) are widely used. These methods independently train segmentation models for each language, or train a single segmentation model on multiple language corpora (Liu et al., 2020).

However, these methods do not directly model the correspondence based on parallel sentence pairs, and thus do not reflect the translation relationship. As a result, word-internal segmentation may become inconsistent between the source and target languages, potentially hindering the training of the translation model. For example, in Japanese-English translation, consider the paired sentences “nonextended” and “延長されなかった (not extended)”. Suppose “nonextended” is segmented as “no next end ed” and “延長されなかった (not extended)” is segmented as “延長 (extend) され (ed) なかった (not)”. If the NMT model learns “next” as “延長 (extend)”, it will fail to produce the correct translation result. To address this issue, subword segmentation considering translation pairs (Deguchi et al., 2020; Hiraoka et al., 2021) has been proposed. However, Deguchi et al. (2020)’s method adjusts the shorter sequence between the source and target sentences to match the token count of the longer one. While this balances sequence lengths, there is no guarantee that word-internal segmentation will be consistent across languages. Hiraoka et al. (2021)’s bilingual subword segmentation requires training the NMT model, entailing consistent computational costs for both subword segmentation and machine translation model training.

We propose a novel subword segmentation method that acquires subword sequences based on the correspondence between subwords in parallel sentence pairs. The proposed method uses SentencePiece (Kudo and Richardson, 2018), a unigram language model, to obtain candidate subword segmentations for source and target sentences in the parallel corpus. It then learns the correspondence between subwords in each bilingual subword sentence pair as alignment probabilities. Since subword alignments are unobserved, we employ the EM algorithm, which is standard for training latent variable models. The generation probability from the unigram language model is multiplied by the alignment probability, and the subword pair with the highest probability is used as training data. During translation, since the target language sentence does not exist, marginalization of the alignment probability is performed on the target language subword, and similarly, the subword sentence with the highest probability is used as translation input. The proposed method outperformed conventional ones in 13 out of 16 translation tasks in terms of BLEU.

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 4: Student Research Workshop, pages 528–534
March 26, 2026 ©2026 Association for Computational Linguistics

2 Conventional Method

This section describes the subword segmentation method based on the unigram language model (Kudo, 2018), which serves as the foundation for the proposed approach. The unigram language model assumes subword independence and expresses the occurrence probability $P_{\mathrm{U}}(\boldsymbol{x})$ of a subword sequence using the following equation:

$$P_{\mathrm{U}}(\boldsymbol{x}) = \prod_{i=1}^{I} P(u_i) \quad s.t. \quad \sum_{u \in V} P(u) = 1, \tag{1}$$

where $\boldsymbol{x} = (u_1, \ldots, u_i, \ldots, u_I)$ is a subword sequence, and each u_i is an element of the subword set V. The subword occurrence probability $P(u)$ is estimated by the EM algorithm to maximize the marginal likelihood L_{lm} expressed by

$$L_{\mathrm{lm}} = \sum_{n=1}^{N} \log P(X_n) = \sum_{n=1}^{N} \log \left(\sum_{\boldsymbol{x} \in S(X_n)} P_{\mathrm{U}}(\boldsymbol{x}) \right), \tag{2}$$

where N denotes the number of sentences in the training data, X_n is the n-th sentence, and $S(X_n)$ represents the candidate set of subword sequences that can be generated for X_n.

After model training, the subword sequence with the maximum probability for sentence X is calculated using the following formula.

$$\boldsymbol{x}^* = \operatorname*{argmax}_{\boldsymbol{x} \in S(X)} P_{\mathrm{U}}(\boldsymbol{x}) \tag{3}$$

Additionally, k-best segmentation candidates can similarly be computed based on $P_{\mathrm{U}}(\boldsymbol{x})$. Our method uses these to construct a set of subword segmentation candidates, whereas the conventional method uses 1-best segmentation.

3 Proposed Method

This section describes the subword segmentation method that learns the correspondence between subwords in bilingual sentence pairs. We define the probabilistic model for subword-aligned sentences (Section 3.1), derive the alignment probability update using the EM algorithm (Section 3.2), and perform subword segmentation on both training and test data (Sections 3.3 and 3.4).

3.1 Probabilistic Model

Given source language sentence X and target language sentence Y, the probabilistic model for subword segmentation in the proposed method is defined by

$$\begin{aligned} P(X,Y) &= \sum_{\boldsymbol{x} \in S(X)} \sum_{\boldsymbol{y} \in S(Y)} \sum_{a \in A(\boldsymbol{x},\boldsymbol{y})} P_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{y}, a) \\ &\approx \sum_{k,l} \sum_{a \in A(\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(l)})} P_{\mathrm{M}}(\boldsymbol{x}^{(k)}, \boldsymbol{y}^{(l)}, a), \end{aligned} \tag{4}$$

where among the candidate sets $S(X)$ of subword sequences for X, the top-K sequences with the highest probability $P_{\mathrm{U}}(\boldsymbol{x})$ are respectively denoted as $\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(k)}, \ldots, \boldsymbol{x}^{(K)}$, and the top-$L$ subword sequences from the candidate set $S(Y)$ for Y with the highest probability $P_{\mathrm{U}}(\boldsymbol{y})$ are denoted as $\boldsymbol{y}^{(1)}, \ldots, \boldsymbol{y}^{(l)}, \ldots, \boldsymbol{y}^{(L)}$. Here, $A(\boldsymbol{x}, \boldsymbol{y})$ represents the set of all possible alignments between each subword of the source subword sequence $\boldsymbol{x}$ and each subword of the target subword sequence $\boldsymbol{y}$. Furthermore, $a \in A(\boldsymbol{x}, \boldsymbol{y})$ represents one specific subword alignment. P_{M} is a probability model for a subword sequence $\boldsymbol{x}$ in the source language sentence, a subword sequence $\boldsymbol{y}$ in the target language sentence, and their alignment a. This model is defined as follows:

$$P_{\mathrm{M}}(\boldsymbol{x}, \boldsymbol{y}, a) = P_{\mathrm{U}}(\boldsymbol{x}) P_{\mathrm{U}}(\boldsymbol{y}) \prod_{(u,v) \in a} \alpha_{uv}, \tag{5}$$

where α_{uv} is the joint probability of the source language subword u and the target language subword v. Here we call it *alignment probability*.

3.2 Learning the Alignment Probability

The alignment probability α_{uv} is computed using the EM algorithm. Calculating the Q function using Equation 5 (Appendix A.1) yields the following equation:

$$Q = \sum_{n,k,l,a} \frac{P_{\mathrm{M}}^{\mathrm{old}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a) \log P_{\mathrm{M}}^{\mathrm{new}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a)}{\sum_{k',l',a'} P_{\mathrm{M}}^{\mathrm{old}}(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')}, a')}. \tag{6}$$

By maximizing the Q function in Equation 6 with respect to $\alpha_{uv}^{\mathrm{new}}$, we obtain the update equation for $\alpha_{uv}^{\mathrm{new}}$ (Appendix A.2) as :

$$\alpha_{uv}^{\mathrm{new}} = \frac{\sum_{n,k,l} E_{nkluv}}{\sum_{u' \in V'_{\mathrm{src}}} \sum_{v' \in V'_{\mathrm{tgt}}} \sum_{n,k,l} E_{nklu'v'}}, \tag{7}$$

$$E_{nkluv} \approx \frac{\left(P_{\mathrm{U}}(\boldsymbol{x}_n^{(k)}) P_{\mathrm{U}}(\boldsymbol{y}_n^{(l)}) \prod_{u \in \boldsymbol{x}^{(k)}} \sum_{v \in \boldsymbol{y}^{(l)}} \alpha_{uv}^{\mathrm{old}} \right) C_{nkluv}}{\sum_{k',l'} P_{\mathrm{U}}(\boldsymbol{x}_n^{(k')}) P_{\mathrm{U}}(\boldsymbol{y}_n^{(l')}) \prod_{u \in \boldsymbol{x}^{(k')}} \sum_{v \in \boldsymbol{y}^{(l')}} \alpha_{uv}^{\mathrm{old}}}, \tag{8}$$

where V'_{src} is the source language's subword set, V'_{tgt} is the target language subword set, and C_{nkluv} is the number of times subwords u and v cooccur in the bilingual subword sentence pair $\boldsymbol{x}_n^{(k)}$ and $\boldsymbol{y}_n^{(l)}$ of the n-th sentence.

3.3 Subword Segmentation of Training Data

For each sentence pair X, Y in the training data, we calculate the subword sequences $\boldsymbol{x}^{(\hat{k})}$, $\boldsymbol{y}^{(\hat{l})}$ that maximize the correspondence based on the alignment probability according to the following equation, and adopt these as the subword sentence pair.

$$\hat{k}, \hat{l} = \underset{k,l}{\operatorname{argmax}}\, P_{\mathrm{U}}(\boldsymbol{x}^{(k)}) P_{\mathrm{U}}(\boldsymbol{y}^{(l)}) \prod_{u \in \boldsymbol{x}^{(k)}} \sum_{v \in \boldsymbol{y}^{(l)}} \alpha_{uv} \quad (9)$$

3.4 Subword Segmentation of Test Data

In subword segmentation for test data, since the target language sentence does not exist, the probability of source language subwords is calculated by marginalizing the alignment probability on the target language subwords as follows:

$$\alpha'_u = \sum_{v \in V_{\mathrm{tgt}}} \alpha_{uv}. \quad (10)$$

Each test sentence X is segmented into $\boldsymbol{x}^{(\hat{k})}$ according to the following equation:

$$\hat{k} = \underset{k}{\operatorname{argmax}}\, P_{\mathrm{M}'}(\boldsymbol{x}^{(k)}), \quad (11)$$

$$P_{\mathrm{M}'}(\boldsymbol{x}) = P_{\mathrm{U}}(\boldsymbol{x}) \prod_{u \in \boldsymbol{x}} \alpha'_u. \quad (12)$$

4 Experiments

To verify the effectiveness of the proposed method, we conducted machine translation experiments comparing it with a conventional method (unigram language model) across six different language pairs (en-ja, ja-zh, en-de, en-hi, en-id, en-th). Additionally, for en-ja, we used three datasets with different data distributions and similarly.

4.1 Dataset

For the en-ja dataset, we used WAT Asian Scientific Paper Excerpt Corpus (ASPEC) (Nakazawa et al., 2016) for English-Japanese and Japanese-English translation tasks, the Kyoto Free Translation Task (KFTT) (Neubig, 2011), and Wikimatrix v1 (Wikimatrix) (Schwenk et al., 2021). For ja-zh, we used the ASPEC Japanese-Chinese and Chinese-Japanese translation tasks. For en-de, WMT18 News Commentary v13 (WMT18)[1] was used for training data, WMT17 testsets for validation data, and WMT18 testsets for test data. For en-hi and en-id, Wikimatrix was used. For en-th, the large-scale English-Thai parallel corpus (scb-mt-en-th) (Lowphansirikul et al., 2022) was used. For Wikimatrix and scb-mt-en-th, the validation data used the flores200[2] (Team et al., 2022) dev set, and the test data used the flores200 devtest set. The dataset composition is shown in Table 4 (Appendix B).

4.2 Experimental Setup

SentencePiece (Kudo and Richardson, 2018) was used to obtain candidate sets of subword sequences for the unigram language model. The unigram language models for the source and target languages were trained independently with a vocabulary size of 16k each. The number of subword candidates was set to the top 10 most probable occurrences (K=L=10) generated by the unigram language model for both the source and target languages. We conducted subword segmentation using a conventional method and the proposed method, and evaluated the performance of NMT models trained on each segmentation output.

The NMT model used Fairseq (Ott et al., 2019), employing the Transformer base (Vaswani et al., 2017) model. For all NMT models, Adam (Kingma and Ba, 2015) was used for parameter optimization with a learning rate of 1e-4 and a batch size of 128. Other parameters used Fairseq's default values. Training was terminated after 30 epochs. For evaluation, the model from each epoch that achieved the highest SacreBLEU (Post, 2018) score on the validation data was used to translate the test data.

Translation performance was evaluated using SacreBLEU and COMET scores[3] (Rei et al., 2022). For SacreBLEU, flores200 was used for tokenization of flores200, ja-mecab (Kudo et al., 2004) for Japanese, zh for Chinese, and 13a for English and German. Experiments were run three times with different random seeds, and the average was taken as the experimental result.

4.3 Experimental Results

Table 1 shows the results of automatic evaluation using BLEU. As shown in the table, the proposed method achieved improved performance over conventional methods in 13 out of 16 machine translation tasks. Furthermore, for machine translation tasks involving languages without word segmentation, performance improvements exceeding

[1] https://www.statmt.org/wmt18/translation-task.html

[2] https://github.com/facebookresearch/flores

[3] https://huggingface.co/Unbabel/wmt22-comet-da

	ASPEC				WMT18		Wikimatrix						scb-mt-en-th		KFTT	
	en-ja	ja-en	ja-zh	zh-ja	en-de	de-en	en-hi	hi-en	en-id	id-en	en-ja	ja-en	en-th	th-en	en-ja	ja-en
Conventional	27.2	27.0	35.4	28.9	21.4	21.7	22.0	19.9	**41.9**	**36.5**	19.3	20.9	28.3	17.2	22.0	20.9
Proposed	**27.6**	**27.5**	**35.5**	**29.2**	**22.0**	**21.8**	22.0	**20.2**	41.6	36.0	**20.3**	**21.2**	**28.8**	**17.3**	**22.8**	**21.3**

Table 1: Results of the BLEU Evaluation

	ASPEC				WMT18	
	en-ja	ja-en	ja-zh	zh-ja	en-de	de-en
Conventional	**0.8882**	0.8182	0.8675	0.9049	0.6482	0.6650
Proposed	0.8880	**0.8195**	**0.8680**	**0.9055**	**0.6517**	**0.6676**

	Wikimatrix					
	en-hi	hi-en	en-id	id-en	en-ja	ja-en
Conventional	**0.6215**	0.7403	**0.8735**	**0.8395**	0.8321	0.8037
Proposed	0.6196	**0.7439**	0.8721	0.8375	**0.8354**	**0.8064**

	scb-mt-en-th		KFTT	
	en-th	th-en	en-ja	ja-en
Conventional	0.7576	**0.7519**	0.8102	**0.7576**
Proposed	**0.7629**	0.7509	**0.8137**	0.7554

Table 2: Results of the COMET Evaluation

	Segmentation results	Translation results
Gold	quilibrious interval disorder	平衡間隔失調 (equilibrious interval disorder)
Conventional	_ qui lib r ious _ interval _ disorder	巧 妙 な 区間 障害 (clever interval disorder)
Proposed	_ qui lib ri ous _ interval _ disorder	平衡 間隔 障害 (equilibrious interval disorder)

Table 3: Example of improved translation using the proposed method

conventional methods were confirmed across all tasks. Notably, translation performance from segmented languages to unsegmented languages improved substantially. This is attributed to the proposed method eliminating unnatural segmentation common in conventional approaches for unsegmented languages, enabling the learning of correct subword correspondences within translation pairs.

Table 2 shows the results of the automatic evaluation using COMET. The proposed method outperformed the conventional method in 10 out of 16 machine translation tasks, while no consistent improvement was observed in the remaining 6 tasks. Although BLEU performance improved, no consistent improvement was observed in COMET. This suggests that the proposed method contributed to improving lexical choices without affecting the overall semantic quality of the sentences.

4.4 Analysis

Table 3 shows examples where translation quality was improved by applying the proposed method. While the conventional method failed to segment correctly, leading to erroneous translations, the proposed method improved segmentation, resulting in translations closer to the correct answers.

When examining the percentage of segments matching between the conventional and proposed methods in the training data, a high match rate was observed on the source language side, whereas a low match rate was seen on the target language side. For specific numerical values, Table 5 (Appendix C) shows the percentage of segments where segmentation matches between the conventional and proposed methods. This is because $\prod_{u \in \boldsymbol{x}^{(k)}} \sum_{v \in \boldsymbol{y}^{(l)}} \alpha_{uv}^{\text{old}}$ in Equation 8 is asymmetrical between the source sentence and the target sentence. With the search space restricted to the top-k candidates, the source-side segmentation is largely determined by the unigram likelihood. As a result, the model prefers high-probability source tokens, while allowing the target-side segmentation to adapt flexibly to align with the source tokens.

5 Conclusion

This study proposes a novel subword segmentation method that learns subword correspondences within parallel translation pairs using the EM algorithm. Experimental results confirm the effectiveness of the proposed method, demonstrating improved translation performance compared to conventional segmentation methods. As future work, we plan to extend the proposed method to multilingual settings and assess its effectiveness in multilingual subword segmentation.

Limitations

This study has several limitations.

First, since this method learns correspondences between subwords based on parallel sentence pairs, it requires a certain amount of parallel corpus data. Therefore, direct application is difficult for low-resource language pairs where sufficient parallel data cannot be prepared.

Furthermore, this study formulates the correspondence between subwords as a context-independent probabilistic model and estimates it using the EM algorithm. Consequently, it cannot explicitly handle more complex alignments, such as correspondences that vary depending on context, correspondences between groups composed of multiple subwords, or even non-continuous correspondences.

Furthermore, since only the top-K segmentations generated by the unigram language model are used as subword segmentation candidates, any segmentation not included in this candidate set is disregarded. Depending on the choice of top-K, the optimal segmentation may be omitted from the candidate set.

From the perspective of computational cost, the need to estimate alignment probabilities among multiple subword segmentation candidates increases the training cost compared to the conventional subword segmentation methods.

Acknowledgments

These research results were obtained from the commissioned research (No.22501) by National Institute of Information and Communications Technology (NICT), Japan. This work was supported by JSPS KAKENHI Grant Number JP24K15071.

References

Hiroyuki Deguchi, Masao Utiyama, Akihiro Tamura, Takashi Ninomiya, and Eiichiro Sumita. 2020. Bilingual Subword Segmentation for Neural Machine Translation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4287–4297.

Tatsuya Hiraoka, Sho Takase, Kei Uchiumi, Atsushi Keyaki, and Naoaki Okazaki. 2021. Joint Optimization of Tokenization and Downstream Model. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 244–255.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR 2015)*.

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.

Taku Kudo, Kaoru Yamamoto, and Yuji Matsumoto. 2004. Applying Conditional Random Fields to Japanese Morphological Analysis. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 230–237.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual Denoising Pre-training for Neural Machine Translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Lalita Lowphansirikul, Charin Polpanumas, Attapol T Rutherford, and Sarana Nutanong. 2022. A Large English–Thai Parallel Corpus from the Web and Machine-Generated Text. *Language Resources and Evaluation*, 56(2):477–499.

Toshiaki Nakazawa, Manabu Yaguchi, Kiyotaka Uchimoto, Masao Utiyama, Eiichiro Sumita, Sadao Kurohashi, and Hitoshi Isahara. 2016. ASPEC: Asian Scientific Paper Excerpt Corpus. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2204–2208.

Graham Neubig. 2011. The Kyoto Free Translation Task. http://www.phontron.com/kftt.

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.

Matt Post. 2018. A Call for Clarity in Reporting BLEU Scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST

2022 Submission for the Metrics Shared Task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2021. WikiMatrix: Mining 135M Parallel Sentences in 1620 Language Pairs from Wikipedia. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1351–1361.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 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.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, and 20 others. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30.

Appendix

A Derivation of the Equation Using the EM Algorithm

A.1 Q Function Derivation

In the probabilistic model of the proposed method, given N pairs of source-target subword sequences $\{(\boldsymbol{x}_n, \boldsymbol{y}_n)\}_{n=1}^{N}$ are provided as training data. The objective is to estimate parameters $\hat{\alpha}_{uv}$ that maximize the probability $p(\boldsymbol{x}_n, \boldsymbol{y}_n|\alpha_{uv})$. In maximum likelihood estimation, we seek the parameters that maximize the joint probability of the entire observed data. Therefore, $\hat{\alpha}_{uv}$ is obtained by the following equation.

$$\begin{aligned}\hat{\alpha}_{uv} &= \operatorname*{argmax}_{\alpha_{uv}} \prod_{n=1}^{N} p(\boldsymbol{x}_n, \boldsymbol{y}_n|\alpha_{uv}) \\ &= \operatorname*{argmax}_{\alpha_{uv}} \sum_{n=1}^{N} \log p(\boldsymbol{x}_n, \boldsymbol{y}_n|\alpha_{uv})\end{aligned} \tag{13}$$

Here, we transform the maximization problem into a logarithmic likelihood expression to simplify calculations. However, since this model includes latent variables, we cannot directly maximize the incomplete data's logarithmic likelihood $\log p(\boldsymbol{x}_n, \boldsymbol{y}_n|\alpha_{uv})$ derived solely from observed data. Therefore, we find the maximum value by iteratively maximizing the difference in the log-likelihood when the parameter changes from α_{uv}^{old} to α_{uv}^{new}.

$$\hat{\alpha}_{uv} = \operatorname*{argmax}_{\alpha_{uv}} Q(\alpha_{uv}^{\text{old}}, \alpha_{uv}^{\text{new}}) \tag{14}$$

Here, the Q function is determined by the following equation.

$$\begin{aligned}Q &= \sum_{n,k,l} p(k,l|\boldsymbol{x}_n, \boldsymbol{y}_n, \alpha_{uv}^{\text{old}}) \log p(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}|\alpha_{uv}^{\text{new}}) \\ &= \sum_{n,k,l} \frac{p(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}|\alpha_{uv}^{\text{old}}) \log p(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}|\alpha_{uv}^{\text{new}})}{p(\boldsymbol{x}_n, \boldsymbol{y}_n|\alpha_{uv}^{\text{old}})} \\ &= \sum_{n,k,l} \sum_{a \in A(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)})} \frac{P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a) \log P_{\text{M}}^{\text{new}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a)}{\sum\limits_{k',l'} \sum\limits_{a' \in A(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')}, a')}\end{aligned} \tag{15}$$

A.2 Update of α_{uv}^{new}

A.2.1 E-Step

We transform the equation to find the probability distribution of α_{uv}^{new}. Taking the logarithm, since $P_{\text{U}}(\boldsymbol{x})$ and $P_{\text{U}}(\boldsymbol{y})$ contained in $P_{\text{M}}^{\text{new}}$ are constant terms, we can ignore them. Thus, the Q function can be modified as follows.

$$\begin{aligned}Q &= \sum_{n,k,l} \sum_{a \in A(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)})} \frac{P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a) \log P_{\text{M}}^{\text{new}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a)}{\sum\limits_{k',l'} \sum\limits_{a' \in A(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')}, a')} \\ &= \sum_{n,k,l} \sum_{a \in A(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)})} \frac{P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a) \sum_{(u,v) \in a} \log \alpha_{uv}^{\text{new}}}{\sum\limits_{k',l'} \sum\limits_{a' \in A(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')}, a')} \\ &= \sum_{u \in V_{\text{src}}} \sum_{v \in V_{\text{tgt}}} \sum_{n,k,l} E_{nkluv} \log \alpha_{uv}^{\text{new}}\end{aligned} \tag{16}$$

$$E_{nkluv} = \frac{\sum\limits_{a \in A(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a) C_{nkluv}}{\sum\limits_{k',l'} \sum\limits_{a' \in A(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')}, a')} \tag{17}$$

where C_{nkluv} is the number of times subwords u and v simultaneously appear in the subword sentence pair $\boldsymbol{x}_n^{(k)}$ and $\boldsymbol{y}_n^{(l)}$ of the n-th sentence.

A.3 M-Step

The Lagrangian function is defined by the following equation.

$$\begin{aligned}\mathcal{L}(\alpha_{uv}^{\text{new}}) = &\sum_{u \in V_{\text{src}}} \sum_{v \in V_{\text{tgt}}} \sum_{n,k,l} E_{nkluv} \log \alpha_{uv}^{\text{new}} \\ &- \lambda(\sum_{u \in V_{\text{src}}} \sum_{v \in V_{\text{tgt}}} \alpha_{uv}^{\text{new}} - 1)\end{aligned} \tag{18}$$

$$\sum_{u \in V_{\text{src}}} \sum_{v \in V_{\text{tgt}}} \alpha_{uv}^{\text{new}} = 1, \quad \alpha_{uv}^{\text{new}} > 0$$

Taking the partial derivative of the Lagrangian yields the following equation.

$$\frac{\partial \mathcal{L}(\alpha_{uv}^{\text{new}})}{\partial \alpha_{uv}^{\text{new}}} = \frac{\sum_{n,k,l} E_{nkluv}}{\alpha_{uv}^{\text{new}}} - \lambda = 0 \tag{19}$$

From the normalization constraint, $\lambda = \sum_{u \in V_{\text{src}}} \sum_{v \in V_{\text{tgt}}} \sum_{n,k,l} E_{nkluv}$. Therefore, the parameter α_{uv}^{new} we seek is given by the following equation.

$$\alpha_{uv}^{\text{new}} = \frac{\sum_{n,k,l} E_{nkluv}}{\sum_{u' \in V'_{\text{src}}} \sum_{v' \in V'_{\text{tgt}}} \sum_{n,k,l} E_{nklu'v'}} \tag{20}$$

$$\begin{aligned} E_{nkluv} &= \frac{\left(\sum_{a \in A(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k)}, \boldsymbol{y}_n^{(l)}, a) \right) C_{nkluv}}{\sum_{k',l'} \sum_{a' \in A(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')})} P_{\text{M}}^{\text{old}}(\boldsymbol{x}_n^{(k')}, \boldsymbol{y}_n^{(l')}, a')} \\ &\approx \frac{\left(P_{\text{U}}(\boldsymbol{x}_n^{(k)}) P_{\text{U}}(\boldsymbol{y}_n^{(l)}) \prod_{u \in \boldsymbol{x}^{(k)}} \sum_{v \in \boldsymbol{y}^{(l)}} \alpha_{uv}^{\text{old}} \right) C_{nkluv}}{\sum_{k',l'} P_{\text{U}}(\boldsymbol{x}_n^{(k')}) P_{\text{U}}(\boldsymbol{y}_n^{(l')}) \prod_{u \in \boldsymbol{x}^{(k')}} \sum_{v \in \boldsymbol{y}^{(l')}} \alpha_{uv}^{\text{old}}} \end{aligned} \tag{21}$$

B Dataset details

	Training	Verification	Test
ASPEC (en-ja)	1,000,000	1,790	1,812
ASPEC (ja-zh)	672,315	2,090	2,107
WMT18 (en-de)	284,246	3,004	2,998
Wikimatrix (en-hi)	231,459	997	1,012
Wikimatrix (en-id)	1,019,170	997	1,012
Wikimatrix (en-ja)	851,706	997	1,012
scb-mt-en-th (en-th)	988,259	997	1,012
KFTT (en-ja)	440,288	1,166	1,160

Table 4: Data Set Statistics

C Analysis details

	ASPEC				WMT18		scb-mt-en-th	
	en-ja	ja-en	ja-zh	zh-ja	en-de	de-en	en-th	th-en
Source	98.4	98.1	98.1	97.4	99.1	98.5	97.5	95.3
Target	29.8	82.6	24.2	28.9	70.6	33.4	18.6	66.7

	Wikimatrix						KFTT	
	en-hi	hi-en	en-id	id-en	en-ja	ja-en	en-ja	ja-en
Source	70.9	92.8	85.7	87.7	95.3	97.9	98.6	98.5
Target	22.9	52.0	54.6	33.0	31.5	36.8	38.0	65.7

Table 5: Percentage of segments matching between conventional and proposed methods (%)