BiMax: Bidirectional MaxSim Score for Bilingual Document Alignment

Xiaotian Wang[1] Takehito Utsuro[1] Masaaki Nagata[2]
[1]Deg. Prog. Sys.&Inf. Eng., Grad. Sch. Sci.&Tech., University of Tsukuba
[2]NTT Communication Science Laboratories, NTT Corporation, Japan
[1]wangxiaotian1999_@_outlook.com [1]utsuro_@_iit.tsukuba.ac.jp
[2]masaaki.nagata_@_ntt.com

Abstract

Document alignment is necessary for the hierarchical mining [1, 2], which aligns documents across source and target languages within the same web domain. Several high-precision sentence embedding-based methods have been developed, such as TK-PERT [3] and Optimal Transport (OT) [4, 5]. However, given the massive scale of web mining data, both accuracy and speed must be considered. In this paper, we propose a cross-lingual sentence-level **Bi**directional **Max**sim score (BiMax) for computing doc-to-doc similarity, to improve efficiency compared to the OT method. Meanwhile, we also conduct a comprehensive analysis to investigate the performance of current state-of-the-art multilingual sentence embedding models.

1 Introduction

Document alignment is the task of finding parallel document pairs, which are identified as translations of each other, within a collection of documents obtained through web crawling. There are four mainstream approaches: URL matching [6, 7], bilingual lexicon [8, 9], machine translation [10, 11], sentence embedding [4, 3, 5, 12].

Wang et al. [13] proposed the overlapping fixed-Length segmentation (OFLS) as an alternative to sentence-based segmentation (SBS) for generating embeddings. When applied to Mean-Pool, TK-PERT [3], and OT [4, 5], this strategy led to speed and accuracy improvements. Among these methods, OT achieves the highest recall on the WMT16 bilingual document alignment shared task based on the LaBSE model [14]. However, the computation of OT inherently involves an optimization process, necessitating multiple iterative operations at the algorithmic level. This results in high computational complexity, limiting its performance in terms of speed. Furthermore, as the number of document segments increases, the processing speed of OT tends to decrease.

Thus, we propose the **Bi**directional **Max**Sim score (BiMax), which matches the maximum similarity between a given segment and the opposed segment collection (e.g., a given source segment and target segment collection) and then sums and averages the similarity scores. The implementation is computationally efficient, requiring only a single similarity matrix computation followed by two max-pooling operations. This idea is inspired by the MaxSim Score in ColBERT [15, 16], which employs a late interaction mechanism to reduce the computational cost between the query and passage by calculating only the maximum similarity for each token in the query relative to the tokens in the passage. We extend this score to the sentence level and make it bidirectional.

Additionally, we evaluate combinations of state-of-the-art embedding models (i.e., models that perform well in tasks such as bitext mining and STS) with various segmentation strategies and document alignment methods on the small-scale Ja-En MnRN dataset [13], aiming to find suitable models and methods for different scenarios.

2 Related Work

Currently, there are four mainstream approaches to document alignment. The first involves simply calculating similarity based on the URLs of the documents [6, 7]. The second uses a bag-of-words or bag-of-ngrams representation of the document contents, leveraging a bilingual lexicon for computation [17, 8, 9]. The third approach employs a Neural Machine Translation (NMT) model to translate documents into the same language, followed by similarity calculations using ngram-based metrics (e.g., BLEU,

ChrF) [18, 10, 11]. The fourth approach utilizes multilingual pre-trained embedding models to map documents into a shared vector space, where similarity is determined by calculating the distances between vectors [4, 3, 5, 12]. In the WMT16 bilingual document alignment shared task [19], numerous techniques and system tools were proposed to align cross-lingual document pairs. However, due to the limitations of technology at the time, all efforts focused on the first three approaches mentioned above, with no exploration of embedding-based methods.

With the proposal and development of pre-trained multilingual sentence embedding models (e.g., LASER [20], mSBERT [21, 22], LaBSE [14]), which map sentences from different languages into a shared multilingual vector space, bitext mining (i.e., matching translation pairs) and Semantic Textual Similarity (STS) calculation have become feasible. This progress also facilitates representing documents using segment embeddings and computing document pair similarities via vector-based methods.

Thompson and Koehn [3] introduced TK-PERT, a method that assigns weights to sentences using regionally emphasized windows derived from a modified PERT distribution [23] to form document feature vectors. Building on this, Sannigrahi et al. [12] evaluated TK-PERT using three multilingual sentence embedding models: LASER, mSBERT, and LaBSE. Optimal Transport (OT) was also applied in cross-lingual document alignment, evolving from word level with Word Movers' Distance (WMD) [24] to sentence level with Sentence Movers' Distance [4, 5]. Wang et al. [13] proposed overlapping fixed-length segmentation (OFLS) instead of sentence-based segmentation (SBS) for the embedding step, improving in both accuracy and speed when replicating previous works. However, their work is limited to using only the LaBSE model and does not explore new document alignment methods.

3 Method

Unlike the MaxSim method utilized in the late interaction of ColBERT [15, 16], which uses the query's hidden word embeddings to search for the most similar token in the passage undirectionally, we apply it to sentence-level as the Bidirectional MaxSim Score (BiMax), introducing the following key modifications: (1) transforming from monolingual to cross-lingual, (2) shifting from word-level embeddings to sentence-level embeddings and (3) moving from one-sided maximum similarity matching to a bidirectional approach.

3.1 Bidirectional MaxSim Score

We define the source document set as $\mathcal{D}_S$ and the target document set as $\mathcal{D}_T$. Following the research of Thompson and Koehn [3], we adopt a 2-stage approach to consider the $\mathcal{D}_S \times \mathcal{D}_T$ possible document pairs:

1. **Candidate Generation**: We first use Mean-Pool or TK-PERT method to generate a single feature vector for each document, and then employ Faiss Search [25] to retrieve K target documents as potential matches for each source document.
2. **Candidate Re-ranking**: We re-rank the $\mathcal{D}_S \times K$ pairs using a more accurate but slower and sometimes more memory-intensive scoring method, such as OT and our proposed BiMax.

Let s_i for $i \in \{0, ..., N_S - 1\}$ be the N_S segments in a given source document S and t_j for $j \in \{0, ..., N_T - 1\}$ be the N_T segments in a given target document T. The BiMax Score is defined as:

$$\text{MaxSim}(S,T) = \frac{1}{N_S} \sum_{i=1}^{N_S} \max_{t \in T} \text{Sim}(s_i, t) \tag{1a}$$

$$\text{BiMax}(S,T) = \frac{1}{2}(\text{MaxSim}(S,T) + \text{MaxSim}(T,S)) \tag{1b}$$

where $\text{Sim}(s, j)$ represents for the similarity score. In this work, we use a pre-trained multilingual sentence embedding model to map the source segment s and the target segment t into the same vector space, producing embeddings E_s and E_t, and then adopt their cosine similarity $cos(E_s, E_t)$ as the similarity score.

4 Analysis on the MnRN Dataset

We use the small-scale MnRN dataset [13], which contains 232 Japanese documents, 931 English documents, and 263 gold pairs[1)] within four web domains, to conduct the analysis under various sentence embedding models, two segmentation strategies, SBS[2)] and OFLS[3)], and four document alignment methods, focusing on three main points: (1) which models are suitable (or unsuitable) for

1) Because the English documents contain duplicates, the number of gold pairs exceeds that of the Japanese documents.
2) Sentence-based Segmentation (SBS): split a document into non-overlapping sentences using delimiters such as line breaks or periods.
3) Overlapping Fixed-Length Segmentation (OFLS): split a document into segments through a fixed-length sliding window, with a proportion of overlap between adjacent segments.

Table 1 The results for comparing SBS and OFLS under each embedding model on the Ja-En MnRN dataset, where "FL" represents for fixed-length, "OR" represents for overlapping rate. For each model and the four document alignment methods, we underline and bold the **result** that achieves the higher F1 score or shorter embedding time under SBS or OFLS.

Strategies & Models		Embedding Models						
		(a) LaBSE	(b) LEALLA-large	(c) paraphrase-multi-MiniLM-L12-v2	(d) distiluse-base-multi-cased-v2	(e) LASER-2	(f) BGE M3 (dense only)	(g) jina-embeddings-v3
Experiments (F1 Score ↑ / Embed. Time (sec.) ↓)								
SBS	Mean-Pool	0.8362 / 131.27s	0.3750 / 60.54s	0.7543 / 59.00s	**0.8362** / 80.40s	**0.5862** / **543.10s**	0.8448 / 637.01s	**0.8362** / **133.72s**
	TK-PERT	0.8448 / **206.19s**	0.5129 / **158.54s**	0.7845 / **158.38s**	0.8147 / **164.89s**	0.5819 / **652.32s**	0.8362 / 745.57s	0.8706 / **247.22s**
	OT w/Mean	0.8448 / 131.58s	0.4525 / 60.87s	0.7845 / 58.98s	0.8448 / 80.46s	**0.4784** / **543.87s**	0.8621 / 642.20s	0.8578 / **132.73s**
	BiMax w/Mean	0.8922 / 131.47s	0.4655 / 60.83s	0.8319 / 59.35s	0.9052 / 80.49s	0.7414 / **543.61s**	0.9181 / 640.27s	**0.9310** / **134.52s**
OFLS (FL 30, OR 0.5)	Mean-Pool	**0.8707** / **71.59s**	**0.3836** / **52.76s**	**0.7759** / **49.06s**	0.8233 / **49.23s**	0.5302 / 1246.64s	**0.8491** / **119.38s**	0.7716 / 380.98s
	TK-PERT	**0.9483** / 569.54s	**0.6034** / 548.93s	**0.8707** / 578.17s	**0.8966** / 591.48s	**0.8134** / 1860.80s	**0.9224** / **650.14s**	**0.9310** / 912.74s
	OT w/Mean	**0.9569** / **71.33s**	**0.4782** / **52.47s**	**0.8578** / **49.08s**	**0.9397** / **49.10s**	0.4354 / 1223.61s	**0.8879** / **119.36s**	**0.8966** / 379.59s
	BiMax w/Mean	**0.9612** / **71.14s**	**0.5348** / **52.93s**	**0.9052** / **49.09s**	**0.9569** / **49.32s**	**0.7845** / 1205.91s	**0.9483** / **119.36s**	0.9267 / 381.05s

OFLS segmentation, (2) how different document alignment methods perform under each model, and (3) which combination of these three factors yields the best results.

The reasons for selecting embedding models and the detailed model settings are recorded in Appendix A and B.

4.1 Performance Comparison

(1) *Which models are suitable (or unsuitable) for OFLS segmentation?*

As shown in Table 1, for models (a)~(d), and (f), OFLS demonstrates similar characteristics, with an improvement in the F1 score in most cases and an acceleration in embedding speed (except for TK-PERT) compared to the SBS segmentation. However, for the LASER-2 model, although the use of OFLS improves the accuracy of the TK-PERT and BiMax methods, its performance on Mean-Pool and OT remains poor. Additionally, the embedding speed is obviously reduced, which may be attributed to the chain structure of LSTM, due to the rise in the total number of tokens resulting from overlapping segments in OFLS.

Specifically, the jina-embeddings-v3 model achieves a relatively high F1 score compared to other models when using the SBS segmentation, with embedding time comparable to LaBSE. Although employing the OFLS strategy may further enhance accuracy, the embedding time for the jina-embeddings-v3 model, unlike other Transformer-based models, becomes longer, which may be caused by the use of RoPE [26] and FlashAttention 2 [27] mechanisms.

(2) *How different document alignment methods perform under each model?*

We select four well-performing models from Table 1, LaBSE, distiluse-base-multi-cased-v2, BGE M3, and jina-embeddings-v3, for further comparison of document alignment methods. As described in Table 3, firstly, as a common feature across all models and segmentation strategies, the embedding time required by the Mean-Pool method is less than TK-PERT. However, in terms of similarity computation, Mean-Pool and TK-PERT cost similarly, as they only involve cosine similarity calculations under sufficient GPU memory. Furthermore, due to the limited scale of the MnRN dataset, the times for similarity calculation under different segmentation strategies and embedding models do not differ significantly for the four document alignment methods. Thus, we present these times in ranges in Table 2, while Appendix A provides detailed results. It can be observed that the time required for BiMax to calculate similarity scores is shorter than OT.

Table 2 The time consumption for calculating similarity.

Methods	Mean-Pool	TK-PERT	OT w/Mean	BiMax w/Mean
Sim Time (sec.) ↓	2.06s~3.03s	2.09~2.97s	12.66s~24.57s	2.12s~3.23s

Subsequently, across the segmentation strategies for each model in Table 3, BiMax achieves the best performance in most cases, except for the jina-embeddings-v3 model employing OFLS with fixed-length 30 for segmentation. The method yielding the second-highest accuracy is generally OT or TK-PERT, but OT shows a higher sensitivity to the window length setting using OFLS.

Table 3 The results for comparing the four document alignment methods under each embedding model and the segmentation strategy. For each segmentation strategy under each model, we highlight the best and second among the document alignment methods. (Comparisons are conducted for each cell of the table.)

Strategies & Models		LaBSE	distiluse-base-multi-cased-v2	BGE M3	jina-embed-v3
Experiments (F1 Score ↑ / Embed. Time (sec.) ↓)					
SBS	Mean-Pool	0.8362 / 131.27s	0.8362 / 80.40s	0.8448 / 637.01s	0.8362 / 133.72s
	TK-PERT	0.8448 / 206.19s	0.8147 / 164.89s	0.8362 / 745.57s	0.8706 / 247.22s
	OT w/Mean	0.8448 / 131.58s	0.8448 / 80.46s	0.8621 / 642.20s	0.8578 / 132.73s
	BiMax w/Mean	0.8922 / 131.47s	0.9052 / 80.49s	0.9181 / 640.27s	0.9310 / 134.52s
OFLS (30, 0.5)	Mean-Pool	0.8707 / 71.59s	0.8233 / 49.23s	0.8491 / 119.38s	0.7716 / 380.98s
	TK-PERT	0.9483 / 569.54s	0.8966 / 591.48s	0.9224 / 650.14s	0.9310 / 912.74s
	OT w/Mean	0.9569 / 71.33s	0.9397 / 49.10s	0.8879 / 119.36s	0.8966 / 379.59s
	BiMax w/Mean	0.9612 / 71.14s	0.9569 / 49.32s	0.9483 / 119.36s	0.9267 / 381.05s
OFLS (100, 0.5)	Mean-Pool	0.8663 / 67.85s	0.8577 / 46.93s	0.8663 / 103.28s	0.7845 / 169.16s
	TK-PERT	0.8966 / 208.19s	0.9052 / 209.13s	0.8836 / 261.69s	0.8836 / 329.34s
	OT w/Mean	0.8922 / 68.35s	0.8707 / 47.01s	0.8405 / 103.16s	0.8491 / 167.97s
	BiMax w/Mean	0.9440 / 68.32s	0.9353 / 47.02s	0.9224 / 103.19s	0.9397 / 168.55s

(3) *Which combination of these three factors yields the best results?*

Furthermore, we record the maximum memory consumption[4] of the four well-performing models in Table 4.

Table 4 The maximum memory consumption of the four embedding models.

Strategies & Models		LaBSE	distiluse-base-multi-cased-v2	BGE M3	jina-embed-v3
Memory Consumption: Embedding (MB.)↓					
SBS	Mean-Pool	4455.33	7267.58	57924.36	7036.57
	TK-PERT	4478.97	7291.22	57948.21	7052.71
OFLS (30, 0.5)	Meam-Pool	2758.95	1685.84	2338.35	3203.90
	TK-PERT	2782.64	1715.25	2370.38	3235.67
OFLS (100, 0.5)	Mean-Pool	2541.99	1670.95	1731.11	2450.66
	TK-PERT	2565.64	1694.64	1762.69	2482.38

Overall, when using OFLS, LaBSE demonstrates superior accuracy compared to other models, and among the document alignment methods, according to Table 3, BiMax achieves the best performance. The model closest to LaBSE under OFLS, distiluse-base-multilingual-cased-v2, while lower in accuracy, offers advantages in terms of speed and memory efficiency.

Regarding SBS, the jina-embeddings-v3 model attains higher accuracy while demonstrating a speed comparable to LaBSE, performing the best in the BiMax method. Although the BGE M3 model also achieves a relatively high F1 score, its memory consumption indicates inefficiency in handling the long-text challenge caused by SBS.

In addition, in the case of low-resource language pairs, where regardless of the embedding model, high embedding accuracy cannot be fully guaranteed, if LaBSE covers the languages, the LaBSE + OFLS + BiMax approach, which achieves fast speed while maintaining a relatively high level of accuracy, may be a recommended method.

5 Experiment on the WMT16 document alignment shared task

To test the BiMax method further, we conduct experiments on the WMT16 document alignment task. For a comparison with the work of Wang et al. [13], we set the fixed-length to 100 and the overlapping rate to 0.5 for OFLS, while using the LaBSE model for embedding generation.

The results are presented in Table 5. Similarly, under the OFLS segmentation, the BiMax method improves 0.3% to 2.4% recall than SBS. Compared with the results of Wang et al. [13], the BiMax method demonstrates slightly higher accuracy than the OT and TK-PERT methods under SBS. However, the opposite trend is observed when employing OFLS. Although the BiMax method cannot comprehensively outperform OT and TK-PERT in terms of recall, we have shown its efficiency in speed in Section 4.1. Furthermore, rather than solely prioritizing precision, this research emphasizes the efficiency of the method. While there is still room for improvement in the accuracy of the BiMax score, such as incorporating weights (e.g., LIDF) for the maximum similarity score of each segment, we opt for a lightweight approach to minimize additional computational overhead and time consumption.

Table 5 The results of soft recall on WMT16 test data, compared to previous best-reported results, where the fixed-length is 100, the overlapping rate is 0.5 for OFLS.

Method	Segment Strategy	Recall
Wang et al. [13] (LaBSE)		
Mean-Pool	SBS	82.6%
Mean-Pool	OFLS	**92.6%**
TK-PERT	SBS	95.2%
TK-PERT	OFLS	**96.3%**
OT w/Mean-Pool	SBS	90.6%
OT w/Mean-Pool	OFLS	**93.7%**
OT w/TK-PERT	SBS	95.6%
OT w/TK-PERT	OFLS	**96.8%**
This work (LaBSE)		
BiMax w/Mean-Pool	SBS	90.7%
BiMax w/Mean-Pool	OFLS	**93.1%**
BiMax w/TK-PERT	SBS	95.8%
BiMax w/TK-PERT	OFLS	**96.1%**

6 Conclusion

This paper introduces a novel and efficient BiMax Score for the document alignment task, reducing computational complexity compared to OT. However, while BiMax shows the best performance across almost all models and various segmentation strategies on the small-scale MnRN dataset, results from the WMT16 document alignment task reveal that we cannot definitively assert BiMax's accuracy surpasses OT or TK-PERT. Instead, we advocate for BiMax primarily for its efficiency in scenarios such as processing large-scale web-crawled data or low-resource language pairs. In these cases, according to our analysis experiments, the LaBSE + OFLS + BiMax approach is recommended, which outperforms all of the other combinations.

4) Since the memory used to calculate similarity scores using OT and BiMax does not exceed the memory required during the embedding process, we limit our comparisons to Mean-Pool and TK-PERT.

References

[1] M. Bañón, P. Chen, B. Haddow, K. Heafield, H. Hoang, M. Esplà-Gomis, M. Forcada, A. Kamran, F. Kirefu, P. Koehn, S. Ortiz Rojas, L. Pla Sempere, G. Ramírez-Sánchez, E. Sarrías, M. Strelec, B. Thompson, W. Waites, D. Wiggins, and J. Zaragoza. ParaCrawl: Web-scale acquisition of parallel corpora. In **Proc 58th ACL**, pp. 4555–4567, 2020.

[2] M. Morishita, K. Chousa, J. Suzuki, and M. Nagata. JParaCrawl v3.0: A large-scale English-Japanese parallel corpus. In **Proc 13th LREC**, pp. 6704–6710, 2022.

[3] B. Thompson and P. Koehn. Exploiting sentence order in document alignment. In **Proc EMNLP 2020**, pp. 5997–6007, 2020.

[4] E. Clark, A. Celikyilmaz, and N. Smith. Sentence mover's similarity: Automatic evaluation for multi-sentence texts. In **Proc 57th ACL**, pp. 2748–2760, 2019.

[5] A. El-Kishky and F. Guzmán. Massively multilingual document alignment with cross-lingual sentence-mover's distance. In **Proc 1st AACL - 10th IJCNLP**, pp. 616–625, 2020.

[6] U. Germann. Bilingual document alignment with latent semantic indexing. In **Proc 1st WMT SIGMT**, pp. 692–696, 2016.

[7] V. Papavassiliou, P. Prokopidis, and S. Piperidis. The ILSP/ARC submission to the WMT 2016 bilingual document alignment shared task. In **Proc 1st WMT SIGMT**, pp. 733–739, 2016.

[8] A. Azpeitia and T. Etchegoyhen. DOCAL - vicomtech's participation in the WMT16 shared task on bilingual document alignment. In **Proc 1st WMT SIGMT**, pp. 666–671, 2016.

[9] M. Medveď, M. Jakubíček, and V. Kovář. English-French document alignment based on keywords and statistical translation. In **Proc 1st WMT SIGMT**, pp. 728–732, 2016.

[10] A. Dara and Y. Lin. YODA system for WMT16 shared task: Bilingual document alignment. In **Proc 1st WMT SIGMT**, pp. 679–684, 2016.

[11] C. Buck and P. Koehn. Quick and reliable document alignment via TF/IDF-weighted cosine distance. In **Proc 1st WMT SIGMT**, pp. 672–678, 2016.

[12] S. Sannigrahi, J. van Genabith, and C. España-Bonet. Are the best multilingual document embeddings simply based on sentence embeddings? In **Findings of EACL 2023**, pp. 2306–2316, 2023.

[13] X. Wang, T. Utsuro, and M. Nagata. Document alignment based on overlapping fixed-length segments. In **Proc. 62nd ACL-SRW**, pp. 51–61, 2024.

[14] F. Feng, Y. Yang, D. Cer, N. Arivazhagan, and W. Wang. Language-agnostic BERT sentence embedding. In **Proc 60th ACL**, pp. 878–891, 2022.

[15] O. Khattab and M. Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In **Proc. 43rd ACM SIGIR**, pp. 39–48, 2020.

[16] K. Santhanam, O. Khattab, J. Saad-Falcon, C. Potts, and M. Zaharia. ColBERTv2: Effective and efficient retrieval via lightweight late interaction. In **Proc NAACL 2022**, pp. 3715–3734, 2022.

[17] P. Fung and P. Cheung. Mining very-non-parallel corpora: Parallel sentence and lexicon extraction via bootstrapping and E. In **Proc EMNLP 2024 SIGDAT**, pp. 57–63, 2004.

[18] L. Gomes and G. Pereira Lopes. First steps towards coverage-based document alignment. In **Proc 1st WMT SIGMT**, pp. 697–702, 2016.

[19] C. Buck and P. Koehn. Findings of the WMT 2016 bilingual document alignment shared task. In **Proc 1st WMT SIGMT**, pp. 554–563, 2016.

[20] M. Artetxe and H. Schwenk. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. **TACL**, pp. 597–610, 2019.

[21] N. Reimers and I. Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In **Proc EMNLP 2019 - 9th IJCNLP**, pp. 3982–3992, 2019.

[22] N. Reimers and I. Gurevych. Making monolingual sentence embeddings multilingual using knowledge distillation. In **Proc EMNLP 2020**, pp. 4512–4525, 2020.

[23] D Vose. Risk analysis: a quantitative guide. John Wiley & Sons, 2000.

[24] M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger. From word embeddings to document distances. In **Proc 32nd PRML**, pp. 957–966, 2015.

[25] J. Johnson, M. Douze, and H. Jégou. Billion-scale similarity search with GPUs. **Journal IEEE 2019**, pp. 535–547, 2019.

[26] J. Su, M. Ahmed, Y. Lu, S. Pan, W. Bo, and Y. Liu. Roformer: Enhanced transformer with rotary position embedding. **Neurocomputing**, p. 127063, 2024.

[27] T. Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. In **12th ICLR**, 2024.

[28] K. Heffernan, O. Çelebi, and H. Schwenk. Bitext mining using distilled sentence representations for low-resource languages. In **Findings of EMNLP 2022**, pp. 2101–2112, 2022.

[29] Z. Mao and T. Nakagawa. LEALLA: Learning lightweight language-agnostic sentence embeddings with knowledge distillation. In **Proc 17th EACL**, pp. 1886–1894, 2023.

[30] N. Muennighoff, N. Tazi, L. Magne, and N. Reimers. MTEB: Massive text embedding benchmark. In **Proc 17th EACL**, pp. 2014–2037, 2023.

[31] J. Chen, S. Xiao, P. Zhang, K. Luo, D. Lian, and Z. Liu. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In **Findings of ACL 2024**, pp. 2318–2335, 2024.

[32] S. Sturua, I. Mohr, M. Kalim Akram, M. Günther, B. Wang, M. Krimmel, F. Wang, G. Mastrapas, A. Koukounas, A. Koukounas, N. Wang, and H. Xiao. jina-embeddings-v3: Multilingual embeddings with task lora, 2024.

[33] L. Wang, N. Yang, X. Huang, L. Yang, R. Majumder, and F. Wei. Multilingual e5 text embeddings: A technical report. **arXiv preprint arXiv:2402.05672**, 2024.

[34] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, and W. Chen. Lora: Low-rank adaptation of large language models. **CoRR**, 2021.

Table 6 The results from various sentence embedding models, segmentation strategies, and document alignment methods on the MnRN dataset. For the F1 Score, we highlight the best, second, third, and fourth best across the models for each combination of segmentation strategy and document method. For embedding time, we also highlight the best, second, third, and fourth best. Moreover, we put the highest F1 scores achieved by each model under each segmentation strategy in **bold**.

Info. & Methods		Embedding Models								
		(a) LaBSE	(b) LEALLA-large	(c) paraphrase-multi-MiniLM-L12-v2	(d) distiluse-base-multi-cased-v2	(e) paraphrase-multi-mpnet-base-v2	(f) LASER-2	(g) multi-e5-large	(h) BGE M3 (dense only)	(i) jina-embeddings-v3
Model Info.										
Suitable Task		Bitext.	Bitext.	STS	STS	STS	Bitext.	Multi-task	Multi-task	Multi-task
#Param.		471M	147M	118M	135M	278M	43M	560M	567M	572M
#Dim.		768	256	384	512	768	1024	1024	1024	1024
#Lang.		Multi.	Multi.	Multi.	Multi.	Multi.	Mono.	Multi.	Multi.	Multi.
#Arch.		Transformer	Transformer	Transformer	Transformer	Transformer	LSTM	Transformer	Transformer	Transformer
Experiments (F1 Score ↑ / Embed. Time (sec.) ↓ / Sim. Time (sec.) ↓)										
SBS	Mean-Pool	0.8362 / 131.27s / 2.12s	0.3750 / 60.54s / 2.07s	0.7543 / 59.00s / 2.06s	0.8362 / 80.40s / 2.10s	0.7716 / 148.60s / 2.16s	0.5862 / 543.10s / 2.12s	0.7802 / 457.94s / 2.12s	0.8448 / 637.01s / 3.03s	0.8362 / 133.72s / 2.38s
	TK-PERT	0.8448 / 206.19s / 2.18s	**0.5129** / 158.54s / 2.09s	0.7845 / 158.38s / 2.17s	0.8147 / 164.89s / 2.12s	0.7931 / 223.87s / 2.11s	0.5819 / 652.32s / 2.11s	0.7845 / 517.99s / 2.16s	0.8362 / 745.57s / 2.97s	0.8706 / 247.22s / 2.47s
	OT w/Mean	0.8448 / 131.58s / 24.57s	0.4525 / 60.87s / 18.96s	0.7845 / 58.98s / 18.39s	0.8448 / 80.46s / 21.87s	0.7974 / 149.07s / 19.83s	0.4784 / 543.87s / 17.24s	0.8060 / 461.78s / 17.13s	0.8621 / 642.20s / 19.32s	0.8578 / 132.73s / 19.60s
	BiMax w/Mean	**0.8922** / 131.47s / 2.19s	0.4655 / 60.83s / 2.13s	**0.8319** / 59.35s / 2.12s	**0.9052** / 80.49s / 2.16s	**0.8577** / 148.40s / 2.19s	**0.7414** / 543.61s / 2.20s	**0.8750** / 462.17s / 2.50s	**0.9181** / 640.27s / 3.23s	**0.9310** / 134.52s / 2.40s
OFLS (FL 30, OR 0.5)	Mean-Pool	0.8707 / 71.59s / 2.12s	0.3836 / 52.76s / 2.14s	0.7759 / 49.06s / 2.06s	0.8233 / 49.23s / 2.13s	0.7112 / 74.56s / 2.10s	0.5302 / 1246.64s / 2.11s	0.7543 / 259.61s / 2.14s	0.8491 / 119.38s / 2.92s	0.7716 / 380.98s / 2.43s
	TK-PERT	0.9483 / 569.54s / 2.10s	**0.6034** / 548.93s / 2.10s	0.8707 / 578.17s / 2.18s	0.8966 / 591.48s / 2.12s	0.8793 / 599.66s / 2.10s	**0.8134** / 1860.80s / 2.12s	0.8534 / 745.20s / 2.15s	0.9224 / 650.14s / 2.88s	**0.9310** / 912.74s / 2.33s
	OT w/Mean	0.9569 / 71.33s / 14.37s	0.4782 / 52.47s / 14.37s	0.8578 / 49.08s / 13.34s	0.9397 / 49.10s / 14.17s	0.8922 / 74.31s / 13.24s	0.4354 / 1223.61s / 14.48s	0.7801 / 258.70s / 13.91s	0.8879 / 119.36s / 14.67s	0.8966 / 379.59s /14.14s
	BiMax w/Mean	**0.9612** / 71.14s / 2.19s	0.5348 / 52.93s / 2.23s	**0.9052** / 49.09s / 2.21s	**0.9569** / 49.32s / 2.25s	**0.9138** / 74.47s / 2.23s	0.7845 / 1205.91s / 2.24s	**0.9181** / 258.35s / 2.28s	**0.9483** / 119.36s / 3.08s	0.9267 / 381.05s / 2.74s
OFLS (FL 100, OR 0.5)	Mean-Pool	0.8663 / 67.85s / 2.09s	0.4138 / 42.03s / 2.15s	0.7413 / 42.03s / 2.07s	0.8577 / 46.93s / 2.10s	0.7672 / 73.84s / 2.09s	0.5517 / 1053.51s / 2.10s	0.7500 / 221.28s / 2.17s	0.8663 / 103.28s / 2.91s	0.7845 / 169.16s / 2.35s
	TK-PERT	0.8966 / 208.19s / 2.10s	**0.5905** / 195.43s / 2.13s	0.8233 / 200.45s / 2.07s	0.9052 / 209.13s / 2.14s	0.8491 / 221.24s / 2.11s	0.7543 / 1257.60s / 2.10s	0.8491 / 322.00s / 2.16s	0.8836 / 261.69s / 2.93s	0.8836 / 329.34s / 2.39s
	OT w/Mean	0.8922 / 68.35s / 13.69s	0.4741 / 42.02s / 13.58s	0.8190 / 42.05s / 13.37s	0.8707 / 47.01s / 13.91s	0.8319 / 74.31s / 12.89s	0.4440 / 1056.15s / 12.67s	0.7586 / 221.19s / 12.28s	0.8405 / 103.16s / 13.38s	0.8491 / 167.97s / 12.66s
	BiMax w/Mean	**0.9440** / 68.32s / 2.13s	0.5431 / 42.09s / 2.19s	**0.9009** / 42.06s / 2.11s	**0.9353** / 47.02s / 2.18s	**0.8663** / 74.23s / 2.14s	**0.7629** / 1050.41s / 2.25s	**0.9009** / 221.45s / 2.27s	**0.9224** / 103.19s / 3.00 s	**0.9397** / 168.55s / 2.41s

A Embedding Model Selection

In Section 4, first, we choose the LaBSE [14] and LASER-2 models [28], which are frequently used for the bitext mining task, and also include a knowledge-distilled, light-weight variant of LaBSE, the LEALLA model [29]. Subsequently, we employ two representative multilingual models from the Sentence Transformers library[5]: paraphrase-multilingual-MiniLM-L12-v2, and distiluse-base-multilingual-cased-v2 [21], which perform strongly on the STS task. Finally, considering the MTEB benchmark [30], which encompasses several embedding tasks, we select two models that currently achieve state-of-the-art performance on the leaderboard[6], which are capable of processing long sentences and suitable for multi-task scenarios: BGE M3 [31], and jina-embeddings-v3 [32]. However, additionally, we also consider the multilingual-e5-large model [33] and the paraphrase-multilingual-mpnet-base-v2 model [21]. The results are presented in Table 6.

B Embedding Model Settings

We maintain the default configurations for all models, as these configurations represent the most general use cases. However, to establish method consistency, we implement a standardization protocol, converting all vectors to fp32 format and utilizing tensors after the embedding process.

Meanwhile, given that all models except LASER-2 are derived from Hugging Face[7] , we can achieve substantial uniformity in the Python library and code framework, thereby facilitating meaningful comparisons of inference speeds across models. However, due to the LASER-2 model's different library and code program, absolute parity in comparative speed analysis between LASER-2 and other models cannot be established.

Because of the multifunctionality of the three multi-task models, we specify distinct usage. For the multi-e5-large model, which can leverage a prefix (either "query:" or "passage:") as the start of the text, we find that appending "query:" to both the source and target produces the highest accuracy. Regarding the BGE M3 model, which provides three functions for generating different scores, we elect to use only its dense embedding as output. Finally, for the jina-embeddings-v3 model, which offers a selection among various LoRA adapters [34] depending on the desired task, we choose the "text-matching" task.

C Experiment Settings

We follow the experimental settings of Wang et al. [13], configuring the hyper-parameters for the WMT16 document alignment task and the MnRN dataset in the TK-PERT method as $J = 16, \gamma = 20$ and $J = 8, \gamma = 16$.

For evaluation of the WMT16 document alignment shared task, we adhered to previous work [19, 3, 12, 13] via a "soft" recall metric, which assigns credit to document pairs if either the English or French document (but not both) deviates from the reference document pair by less than 5%, based on text edit distance. For the MnRN dataset, the F1 Score is used for evaluation.

All experiments are conducted on two A6000 GPUs and one H100 GPU.

5) https://huggingface.co/sentence-transformers
6) https://huggingface.co/spaces/mteb/leaderboard
7) https://huggingface.co/