🐊 CroCoAlign: A Cross-Lingual, Context-Aware and Fully-Neural Sentence Alignment System for Long Texts

Francesco Maria Molfese[1], **Andrei Stefan Bejgu**[1,2], **Simone Tedeschi**[1,2]
Simone Conia[1] and **Roberto Navigli**[1]
[1]Sapienza NLP Group, Sapienza University of Rome
[2]Babelscape, Italy
{lastname}@diag.uniroma1.it, {bejgu,tedeschi}@babelscape.com

Abstract

Sentence alignment – establishing links between corresponding sentences in two related documents – is an important NLP task with several downstream applications, such as machine translation (MT). Despite the fact that existing sentence alignment systems have achieved promising results, their effectiveness is based on auxiliary information such as document metadata or machine-generated translations, as well as hyperparameter-sensitive techniques. Moreover, these systems often overlook the crucial role that context plays in the alignment process. In this paper, we address the aforementioned issues and propose CROCOALIGN: the first context-aware, end-to-end and fully-neural architecture for sentence alignment. Our system maps source and target sentences in long documents by contextualizing their sentence embeddings with respect to the other sentences in the document. We extensively evaluate CROCOALIGN on a multilingual dataset consisting of 20 language pairs derived from the Opus project, and demonstrate that our model achieves state-of-the-art performance. To ensure reproducibility, we release our code and model checkpoints at `https://github.com/Babelscape/CroCoAlign`.

1 Introduction

Sentence alignment is the task of matching sentences in two or more documents that are related to each other (Abdul-Rauf et al., 2012), as shown in Figure 1. The task is important in many downstream applications, including machine translation (MT, Shi et al., 2021), text simplification (Jiang et al., 2020) and paraphrase generation (Barzilay and Lee, 2003). Although current approaches have achieved promising results on standard benchmarks for the task (Volk et al., 2010), they are strongly focused on hyperparameter-sensitive heuristics and on using auxiliary MT systems, hence overlooking the primary role that context plays when performing sentence alignment (Sennrich and Volk,

Figure 1: Examples of 1-to-1, many-to-1 and 1-to-many alignments between source and target sentences written in English and Italian, respectively.

2011; Thompson and Koehn, 2019). Indeed, sentences can be ambiguous when taken out of context, whereas modeling the surrounding sentences helps in disambiguating meanings, leading to a more accurate alignment. In particular, we note that existing approaches are not suitable for fully addressing complex challenges like many-to-many alignments and the identification of non-alignable sentences, which also require the modeling and understanding of context. We emphasize that the foregoing are not uncommon challenges when aligning long texts, such as books, which may have been adapted through transcreation, according to socio-economic and cultural factors (Gaballo, 2012).

Moreover, current approaches are mainly focused on European parliament transcriptions (Europarl, Koehn, 2005) and on other extremely specific domains, such as the digitized heritage of German and French alpine literature (Text+Berg, Volk et al., 2010) and the Bible (Christodouloupoulos and Steedman, 2015). However, we observe that the peculiarities of the aforementioned corpora – the shortness of Text+Berg and its focus on a single language pair, the political-financial domain of Europarl and its transcriptive style, as well as the genre of the Bible – may not provide a suitable framework for training current approaches and

evaluating their generalizability.

To address the above-mentioned limitations, in this paper we carry out an in-depth investigation on the role of modeling cross-sentence context in the process of sentence alignment in long texts, and put forward the following three main contributions:

- We introduce CROCOALIGN, the first fully-neural sentence alignment system equipped with a novel cross-sentence encoder to model context in long texts;

- We train and evaluate CROCOALIGN on a multilingual dataset derived from the Opus books project,[1] which includes 16 languages, 64 language pairs, and 251 parallel books, showing that our system consistently outperforms the current state of the art on the task of sentence alignment in long texts;

- We demonstrate the quality of the data that CROCOALIGN can produce with downstream experiments on machine translation of books, reporting improved performance over strong MT baselines.

Ultimately, we hope that this study will stimulate further research into sentence alignment systems that can improve the understanding and analysis of long texts as a whole.

2   Related Work

In this section, we review the literature of the sentence alignment task. To highlight the unique aspects of this task, we also outline the differences between sentence alignment and bitext mining, a closely related task, and describe why bitext mining systems are not suitable for sentence alignment. Finally, we showcase applications of sentence alignment systems in MT, underlining the importance of the task in real-world scenarios.

2.1   Sentence Alignment

Traditional ways of aligning sentences were to leverage sentence length information, or to look for lexical patterns. The first sentence alignment systems relied solely on the number of words or characters within each sentence (Brown et al., 1991; Gale and Church, 1993). Similarly, Kay and Röscheisen (1993) presented an alignment algorithm based on word correspondences, while Chen

(1993) calculated the probability of an alignment by using a word-to-word translation model.

To speed up computation, later research merged word-level features with sentence-level translations (Moore, 2002; Varga et al., 2007). It is also possible to align sentences based on their degree of textual and metatextual structure. For instance, Tiedemann (2007) indicated that movie subtitles can be highly attractive for alignment thanks to their time stamps. MT-based methods were introduced in subsequent literature (Sennrich and Volk, 2011), followed five years later by pruned phrase tables from a statistical MT system (Gomes and Lopes, 2016). In both the foregoing methods, high-probability one-to-one alignments were anchored in the search space and then the alignments were filled in and refined.

More recently, Thompson and Koehn (2019) introduced Vecalign, which uses a dynamic programming algorithm based on a combination of LASER (Artetxe and Schwenk, 2019) sentence embeddings and Fast Dynamic Time-Warping, setting a new state of the art in sentence alignment. Although previous work has greatly improved the performance of sentence alignment systems, we still lack an in-depth investigation on how to encode cross-sentence context in long texts.

2.2   Bitext Mining

Bitext mining, also known as bitext retrieval, is the task of mining sentence pairs that are translations of each other from large text corpora. Differently from sentence alignment, where global context and sequentiality play a key role and many-to-many alignments are possible, bitext mining systems focus on standalone, 1-to-1 sentence pairs. Typically, bitext mining systems undergo assessment through established benchmarks, such as the United Nations (Ziemski et al., 2016, UN), BUCC (Zweigenbaum et al., 2017), and the Tatoeba corpora (Artetxe and Schwenk, 2019). Nevertheless, these datasets are organized in a manner where, given two monolingual corpora, only a portion of them is assumed to be parallel. This suggests that the source domain can vary greatly from one sentence to another, thereby being in significant contrast with sentence alignment datasets, where the domain tends to remain consistent throughout the entire document. For this reason, state-of-the-art bitext mining systems, such as LASER (Artetxe and Schwenk, 2019) and LaBSE (Feng et al., 2022), are not designed to handle sequential relationships between sentences within a document, and over-

---

[1] https://opus.nlpl.eu/Books.php

look situations where source or target sentences are fragmented into multiple segments.

2.3 Sentence Alignment for MT

Bitext mining has gained significant attention owing to its ability to generate high-quality parallel data for training MT systems as a result of its straightforward approach and wide-ranging utility (NLLB team et al., 2022). In contrast, sentence alignment systems have received limited recognition, despite the potential advantages they offer by capturing not only the broader context found within parallel documents but also the ordering of the sentences therein. As an example, Shi et al. (2021) illustrated the substantial advantages that an auxiliary sentence alignment system can yield during the training of MT models. Additional studies have also demonstrated that even sentence alignment systems can be employed effectively to automatically generate data for training MT models (Thompson and Koehn, 2019); however, to the best of our knowledge, we still lack an in-depth investigation of how sentence alignment can be used as a means of producing training data for fine-tuning MT systems on long texts or specific domains/genres.

3 CroCoAlign

In this section, we describe CROCOALIGN, a language-agnostic, fully-neural method for aligning sentences between pairs of documents, designed specifically to model context in long texts. Our core intuition is that the embedding of an individual sentence, independently of how expressive it may be, lacks information about its surrounding context, i.e., the previous and following sentences. Therefore, the fundamental novelty of CROCOALIGN lies in its modeling of the document-level sequentiality of the sentence representations. CROCOALIGN accomplishes this process by initially encoding source and target sentences using a sentence transformer (Section 3.1). It subsequently enhances the resulting sentence-level representations by employing a novel context encoder to incorporate additional contextual information at the document level (Section 3.2). Then, it proceeds by determining whether or not a source-target sentence pair is an alignment by feeding the element-wise product of the resulting contextualized sentence embeddings to a multi-label classifier (Section 3.3), minimizing the training objective explained in Section 3.4. Finally, CROCOALIGN also features a novel two-step procedure, which we refer to as POINTING and RECOVERY, needed at inference time to address the problem of locating the sentences to be aligned and refining the predictions, as described in Section 3.5. Figure 2 shows the overall architecture.

3.1 Cross-lingual Sentence Embeddings

There is a significant body of research that shows that sentence embeddings can be employed effectively in bitext mining to filter and locate parallel sentences across multiple corpora (Schwenk et al., 2021). Given the strong relationship between sentence alignment and bitext mining, we build our approach on pretrained sentence embeddings. Specifically, we exploit the inherent structure of cross-lingual sentence embeddings, where, given two sentences written in different languages but having similar meanings, these are mapped to nearby vectors in the space. Differently from previous approaches that exploit bilingual embeddings (Artetxe and Schwenk, 2019), hence requiring one model for each language pair, we employ a language-agnostic sentence transformer.

We stress that our method is independent of the sentence transformer used, and also allows the use of bilingual embeddings. Thus, let $s_1, ..., s_n$ and $t_1, ..., t_m$ be a sequence of source and target sentences, respectively. We encode each of these by means of the aforementioned sentence transformer in order to generate their respective sentence embeddings $E_{s_1}, ..., E_{s_n}$ and $E_{t_1}, ..., E_{t_m}$.

3.2 Encoding Context Across Sentences

In order to refine $E_{s_1}, ..., E_{s_n}$ and $E_{t_1}, ..., E_{t_m}$, we input these embeddings to a randomly-initialized transformer encoder, which we refer to as context encoder (see Figure 2), that, by means of positional embeddings and the attention mechanism, captures the inherent information in the surrounding context. The output of this procedure consists of source and target contextualized sentence embeddings, namely $C_{s_1}, ..., C_{s_n}$ and $C_{t_1}, ..., C_{t_m}$. It is worth noting that it is theoretically possible to obtain a contextual representation of every sentence in a document by encoding all its sentences in a single batch, and that this choice is dictated by hardware constraints.

3.3 Classification

Given $C_{s_1}, ..., C_{s_n}$ and $C_{t_1}, ..., C_{t_m}$ from the previous step, we create a matrix $M_{n \times m}$ where the entry M_{ij} contains the element-wise product of
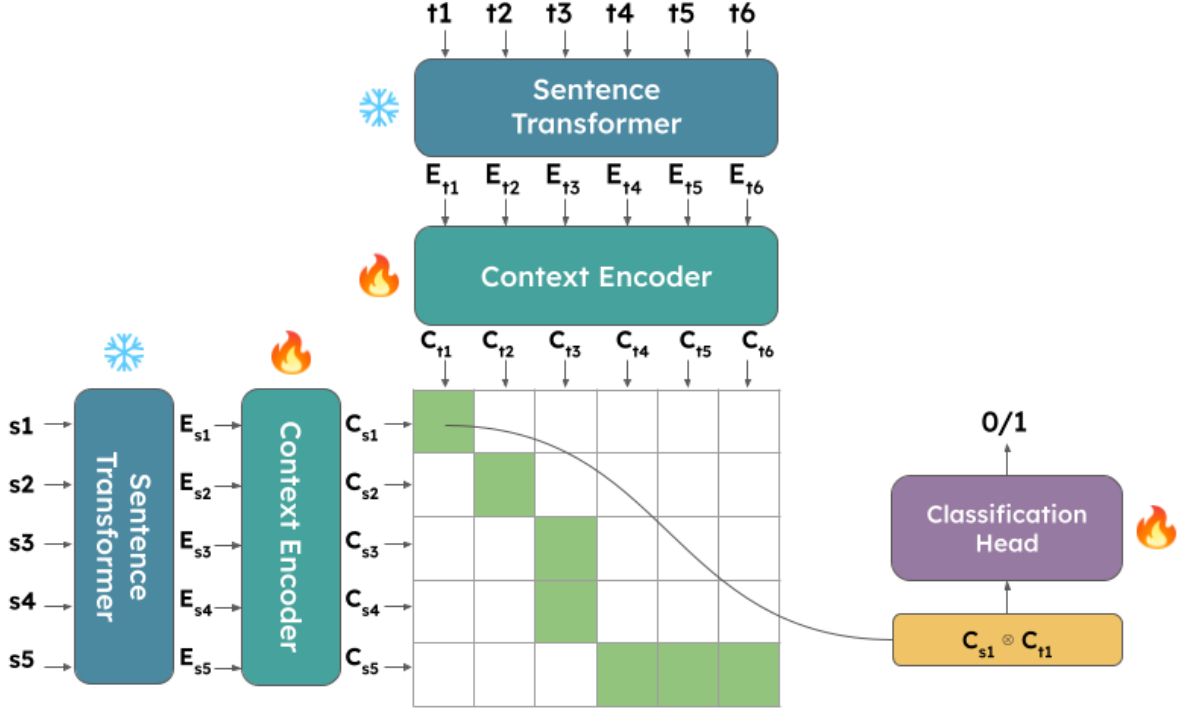
Figure 2: The overall architecture of CROCOALIGN. Given a set of source and a target sentences, each individual sentence s_i and t_j is first encoded to obtain sentence embeddings E_{s_i} and E_{t_j}, respectively. Then, the resulting sentence embeddings are given as input to a context encoder, which produces contextualized sentence embeddings, i.e. C_{s_i} and C_{t_j}. Subsequently, the element-wise product of C_{s_i} and C_{t_j} is computed and fed to a linear layer for classification.

the embeddings C_{s_i} and C_{t_j}. The resulting vector is then fed into a classification head in order to output the logits associated with the sentences to be aligned, updating the value of M_{ij}. These values are converted to probabilities by means of the sigmoid activation function and then rounded to binary values using a threshold of 0.5. This probability is calculated for every pair of source and target sentences in the input batch. As a result, each source sentence can be mapped to zero, one, or multiple target sentences, and vice versa, therefore modeling scenarios such as those in which no sentence in one book has an equivalent in another, or when the alignment is 1-to-many, many-to-1, or many-to-many. Figure 2 shows examples of 1-to-1, many-to-1 and 1-to-many alignments, pictured as green cells in the matrix.

3.4 Training Objective

The model is trained to maximize the element-wise product between the contextualized embeddings of source and target sentences that correspond to an alignment according to the ground truth. At the same time, the model is also trained to minimize the element-wise product of sentences that should

not be aligned. More formally, the loss is computed as follows:

$$\mathcal{L}(M, \hat{M}) = -\frac{1}{n}\frac{1}{m}\sum_{i=1}^{n}\sum_{j=1}^{m}\Big[\hat{M}_{ij} \cdot \log\left(\sigma(M_{ij})\right)$$
$$+(1 - \hat{M}_{ij}) \cdot \log\left(1 - \sigma(M_{ij})\right)\Big],$$

where M and $\hat{M}$ are both matrices in $\mathbb{R}^{n \times m}$ corresponding to the predicted and gold alignments, respectively, while σ is the sigmoid activation function, applied element-wise to the elements of M.

3.5 Identifying Target Contexts and Reducing Noise at Inference Time

While gold alignments are available at training time, allowing us to construct predefined source and target batches to be aligned, this information is missing at inference time. In order to mitigate this issue, we introduce a procedure that we refer to as POINTING, to first identify the source and target batches to be aligned. This procedure uses an approximate nearest neighbor algorithm to find the most closely related target contexts, given a source context. Specifically, using the FAISS algorithm

(Johnson et al., 2019), we identify the set of k candidate target sentence embeddings that are closest to a given source embedding E_{s_i}, according to their cosine similarity. Afterwards, in order to identify the correct target embedding, for each candidate we construct a context of N sentences surrounding and including both the source and the pointed target sentences. After obtaining such contexts, we provide them as input to our model and ask for a prediction. We finally select the target context T with the highest alignment probability, that is, the one with the highest number of alignments.[2]

Additionally, to guarantee alignment between each source sentence and its corresponding target fragments (if any), the subsequent iteration generates a new source context by intersecting $\lceil N/2 \rceil$ sentences with the previous context. This approach makes it possible to align source sentences with target fragments that were originally located outside the boundaries of T.

Due to the strategy just described, there may be some noise in the final prediction. As an example, in different iterations, a source sentence might be aligned with two or more non-adjacent target sentences. In this scenario, we apply a RECOVERY procedure in order to determine which of the non-adjacent target sentences (or combination of target sentences) is most similar to the source sentence. We experiment with different recovery procedures and we explain these in detail in Section 4.3.

4 Experimental Setup

Our system is developed using the Pytorch Lightning framework[3] and the HuggingFace models library.[4] In order to generate the initial sentence embeddings (Section 3.1) for each source and target batch of sentences, we use the sentence transformer Language-Agnostic Bert Sentence Embeddings[5] (LaBSE, Feng et al., 2022). During training, we keep the weights of the sentence transformer frozen. For the transformer encoder (Section 3.2), we employ DistilBERT[6] (Sanh et al., 2019), which is initialized with random weights and then trained with six attention heads, six layers and a dropout

---

[2]We experiment with $k \in \{5, 10, 20, 100\}$ and observe the best results with $k = 10$. We highlight that we penalize retrieved target contexts that are relatively far away in the document from the source context, regardless of the alignment probability.

[3]https://www.pytorchlightning.ai/

[4]https://huggingface.co/docs/transformers/

[5]https://huggingface.co/sentence-transformers/LaBSE

[6]https://huggingface.co/distilbert-base-uncased

of 0.2. For the classification head (Section 3.3), we use a dropout of 0.2 and the ReLU activation function. We train our system on a RTX 3090 Ti for a maximum of 1.25 million steps and an early stopping mechanism with a patience set to 10. We use the AdamW optimizer with a weight decay of 0.01, a learning rate of 10^{-5}, and a linear scheduler with a warmup of 10% of the maximum number of training steps. We select the best model based on its strict F1 score on the validation set, which demands an exact match between the predicted and the gold alignments. In the following, we describe the dataset we use for our experiments (Section 4.1), the baselines we compare with (Section 4.2), and the variants of our model (Section 4.3).

4.1 Dataset

We extract our dataset from the book section of the Opus project website.[7] The website provides a collection of copyright-free books aligned by Andras Farkas.[8] The dataset contains 16 languages, 64 language pairs and a total of 251 parallel books. For each available parallel book, there is a corresponding file specifying the ground truth of the sentences to be aligned according to their IDs. Table 1 summarizes dataset information, such as covered languages, number of books written in the source language, number of sentences and tokens. In addition, Table 2 shows the details about the occurrences of each alignment type contained in the dataset. For validation and test purposes, we select books whose language pairs appear at least 2 times in the overall corpus. We divide the resulting 20 books into half for validation and half for testing, and we use the remaining 231 books for training purposes. Further statistics about our dataset can be found in Appendix A.

4.2 Baseline Systems

Bleualign. The algorithm proposed by Sennrich and Volk (2011) makes use of an external MT system to guide the alignment based on the BLEU score between the given translation and the target sentence. The alignment can also be crossvalidated by entering both source and target translations in order to enhance the performance. The system uses the Gale and Church (1993) algorithm to obtain an initial alignment, and then refines it using MT. However, when the BLEU score between the target sentence and the source translation is

---

[7]https://opus.nlp.eu/Books.php

[8]http://www.farkastranslations.com/

Lang.	# Books	# Sentences	# Tokens
Cᴀ	1	5.0K	93.3K
Dᴇ	12	71.0K	1.3M
Eʟ	1	1.6K	36.5K
Eɴ	42	0.2M	5.9M
Eᴏ	2	2.0K	38.8K
Eѕ	18	0.1M	2.4M
Fɪ	1	3.8K	54.5K
Fʀ	29	0.2M	3.6M
Hᴜ	28	0.2M	3.3M
Iᴛ	8	36.0K	0.8M
Nʟ	9	55.1K	1.3M
Nᴏ	1	4.0K	67.9K
Pʟ	1	3.3K	43.5K
Pᴛ	1	1.5K	32.3K
Rᴜ	3	27.3K	0.5M
Sᴠ	1	3.2K	76.6K
Tᴏᴛᴀʟ	158	0.9M	19.5M

Table 1: Statistics of the dataset extracted from the Opus project. # Books, # Sentence and # Tokens represent the number of books, sentences and tokens associated with the corresponding language.

Split	Align. Type	# Ann.	%
Train	1-to-1	892,320	77.3
	1-to-0	11,136	1.0
	0-to-1	19,966	1.7
	n-to-1	110,580	9.6
	1-to-m	105,918	9.2
	n-to-m	13,853	1.2
Validation	1-to-1	34,282	75.4
	1-to-0	257	0.6
	0-to-1	669	1.5
	n-to-1	5,955	13.1
	1-to-m	3,597	7.9
	n-to-m	684	1.5
Test	1-to-1	35,162	77.4
	1-to-0	298	0.7
	0-to-1	396	0.9
	n-to-1	6,015	13.2
	1-to-m	2,978	6.6
	n-to-m	601	1.3

Table 2: Statistics of the alignment types in our dataset. # Ann. refers to the amount of examples annotated with each type of alignment, while % represents the ratio of each alignment type with respect to the total.

not sufficiently high, the algorithm returns the initial alignment. During their experiments, Sennrich and Volk (2011) used an old version of Google Translate as well as a statistical MT system. To ensure a fair comparison, we replace the latter with OPUS-MT, a robust neural MT system developed by Helsinki NLP, available on HuggingFace.[9]

Vecalign. Thompson and Koehn (2019) introduced Vecalign, which is the current state of the art in sentence alignment. The alignment is performed through the use of LASER bilingual sentence embeddings (Artetxe and Schwenk, 2019) and Fast-Dynamic Time-Warping (see Section 2).

4.3 Model Variants

Each version of our model uses the same POINT-ING strategy, discussed in Section 3.5, which employs the LaBSE sentence transformer (Feng et al., 2022). Therefore, in this section, we focus on describing the different variants of our RECOVERY procedure. Let s_i be a source sentence which has been aligned to non-adjacent target sentences t_j and t_k with $k > j + 1$, constituting an unlikely alignment. We underline that, in general, this procedure can be extended to cases where s_i is aligned

with more than two target sentences. For instance, if there are three candidate target sentences, namely t_x, t_y and t_z, with t_x adjacent to t_y, the procedure will be applied to each sentence individually, as well as to groups of adjacent sentences.

CROCOALIGN-LaBSE. As a means of determining the correct target sentence among the available options, we encode s_i, t_j and t_k independently using LaBSE (Feng et al., 2022). Afterwards, we select the target sentence having the embedding with the highest cosine similarity with the source sentence embedding.

CROCOALIGN-WSD. For the purpose of selecting which target sentence we should keep, we employ a state-of-the-art multilingual Word Sense Disambiguation (WSD) system, namely AMuSE-WSD (Orlando et al., 2022). The system identifies the meanings (i.e. BabelNet synsets) of the words associated with the sentences s_i, t_j, and t_k. Given the synsets associated with each sentence, we select the target sentence with the highest synset intersection. Importantly, we note that this is possible thanks to BabelNet encoding each synset multilingually, i.e. as the set of lexicalizations that are used

---

[9] https://huggingface.co/Helsinki-NLP

Algorithm	EN-IT	EN-ES	EN-FR	EN-NL	EN-RU	EN-HU	DE-IT	DE-HU	DE-FR	DE-ES	DE-EN
Bleualign	93.7	<u>87.0</u>	87.0	<u>87.8</u>	85.3	92.9	63.7	62.6	67.3	93.2	86.8
Vecalign-LASER	95.4	85.7	87.1	87.6	91.3	**96.3**	70.1	76.3	70.7	94.4	88.4
Vecalign-LaBSE	95.7	**89.1**	88.4	**90.6**	92.0	<u>95.7</u>	71.3	72.7	70.6	94.7	87.1
CROCOALIGN-WSD	<u>96.3</u>	77.5	<u>95.7</u>	82.8	**95.7**	90.4	<u>75.9</u>	<u>78.9</u>	68.9	<u>97.8</u>	<u>94.4</u>
CROCOALIGN-LaBSE	96.0	76.8	95.5	82.0	<u>95.4</u>	89.8	75.2	78.8	**78.7**	97.5	94.2
CROCOALIGN-LaBSE$_B$	**96.4**	77.7	**97.0**	83.8	**95.7**	90.8	**80.0**	**81.6**	<u>75.8</u>	**98.9**	**95.6**

(a)

Algorithm	ES-IT	ES-FR	ES-NL	ES-HU	FR-IT	FR-NL	FR-HU	HU-IT	HU-NL	Macro Average
Bleualign	72.8	88.8	<u>86.9</u>	—	—	—	74.4	—	—	82.0
Vecalign-LASER	86.4	91.2	86.5	<u>97.8</u>	65.9	82.7	**88.1**	<u>89.9</u>	86.9	85.9
Vecalign-LaBSE	85.7	88.0	**87.8**	92.7	65.6	82.3	<u>85.0</u>	**90.3**	85.1	85.5
CROCOALIGN-WSD	<u>87.3</u>	**93.9**	85.2	97.5	**66.6**	<u>90.7</u>	72.9	85.9	<u>88.1</u>	<u>86.1</u>
CROCOALIGN-LaBSE	87.1	<u>93.6</u>	85.0	97.5	65.8	90.4	75.5	85.2	88.0	85.7
CROCOALIGN-LaBSE$_B$	**89.5**	92.7	86.0	**98.9**	<u>66.2</u>	**92.3**	76.1	85.1	**88.9**	**87.5**

(b)

Table 3 (a) and (b): Results of CROCOALIGN and its variants (bottom part of the tables) compared with the baseline systems (upper part of the tables). The columns represent strict F1 scores (%) for the corresponding language pairs. In Table (b), the last column reports the average F1 scores obtained by each system across all language pairs. **Bold** represents the best results, while <u>underline</u> represents the second-best results.

in different languages to express the given concept (Navigli et al., 2021). Therefore, translated words are ideally assigned the same synset across languages. Our intuition is that, thanks to the assumption that parallel sentences should share the same semantics, our synset intersection approach is likely to select target sentences with the most accurate translation.

CROCOALIGN-LaBSE$_B$. By exploring in detail the output of the RECOVERY procedure explained in CROCOALIGN-LABSE, we observe that the source sentence s_i alone may not provide enough context to make the right choice between t_j and t_k. To better grasp the contextual information, we concatenate s_i together with N surrounding sentences and then encode the resulting text with LaBSE, generating a single sentence embedding. We then do the same for t_j and t_k. Finally, we select the target sentence associated with the embedding having the highest cosine similarity with the source sentence embedding. We experiment with $N \in \{3, 5, 7\}$ and observe the best results on the validation set with $N = 3$. We name this procedure as CROCOALIGN-LABSE$_{Batched}$, or alternatively CROCOALIGN-LABSE$_B$ for short.

5 Results

In this section, we present the results obtained by CROCOALIGN and its competitors on the dataset introduced in Section 4.1 in terms of strict F1 score.

Quantitative Results. Table 3 summarizes the results. We can observe that our model, along with its variants, outperforms the baselines 14 times out of 20, while the best variant, namely CROCOALIGN-LaBSE$_B$, outperforms all the other solutions 11 times out of 20. Indeed, on average our best model achieves +1.6 F1 points in comparison to Vecalign, the current state of the art in sentence alignment. However, we also point out that, for specific language pairs, the baselines achieve higher results. For instance, Bleualign is able to reach the highest score for the EN-ES, EN-NL and ES-NL language pairs, thanks mainly to the quality of the underlying MT systems for the three languages involved. We note that the absence of results for the Bleualign baseline for some language pairs is attributable to the non-availability of a bilingual MT model from Helsinki NLP for that specific language pair, which is an essential requirement for Bleualign and this, therefore, represents a possible limitation for lower-resource languages. Vecalign, instead, achieves the highest score 3 times out of 20, in the EN-HU, FR-HU and HU-IT language pairs, respectively, thanks mainly to the strength of its underlying bilingual encoder for the Hungarian language.

Finally, despite the fact that Vecalign employs LASER as the underlying sentence encoder, in order to ensure a fair comparison we also evaluate it using LaBSE. The main difference between the

two encoders is that the former is designed as a bilingual model – requiring a distinct model for each language pair – while the latter is language agnostic. As shown in Table 3, we observe that replacing LASER with LaBSE has no beneficial impact on Vecalign's performance. On the contrary, the results of Vecalign with LaBSE are lower than those with LASER, i.e., 85.5 versus 85.9 in F1 score on average across all languages.

Alignment Analysis. In addition to the quantitative results presented in the previous paragraph, we performed an analysis of the accuracy of different alignment types, comparing our best model with Vecalign. In Figure 3 we report the accuracy, expressed as a percentage, of the number of times a specific type of alignment is predicted correctly by the two systems. From the results we can see that, on average, both CROCOALIGN and Vecalign perform similarly when tested on 1-to-1 up to 1-to-7 alignments. However, when presented with particularly challenging types of alignment (upper entries in Figure 3), which can happen frequently in long texts, our system consistently outperforms its competitor. Moreover, CROCOALIGN is also more resilient to other very common situations in long texts where, given a sentence, there is no equivalent in its corresponding parallel document (1-to-0 and 0-to-1 alignments in Figure 3).

Inference Speed. When assessing sentence alignment systems, it is crucial to consider their inference speed, especially given their typical application to extensive datasets. In this context, we conducted a comparative analysis between CROCOALIGN and Vecalign, our primary competitor. Notably, Vecalign employs a combination of dynamic programming and sentence embeddings for a fast alignment process. However, it necessitates a preprocessing step where sentences from source and target documents undergo a complex encoding process using a sentence transformer. Indeed, the algorithm requires the encoding of clusters of adjacent sentences to identify many-to-many alignments, introducing an overhead in the overall process before the execution of the alignment step. As a consequence, CROCOALIGN is $2.7\times$ faster than Vecalign-LASER and $3.3\times$ faster than Vecalign-LaBSE. On a single GTX 1080 Ti, CROCOALIGN requires 84 seconds on average (9.49 of which are devoted to the generation of sentence embeddings and 75.12 to produce the final alignment) to align a pair of parallel books in our dataset. In con-
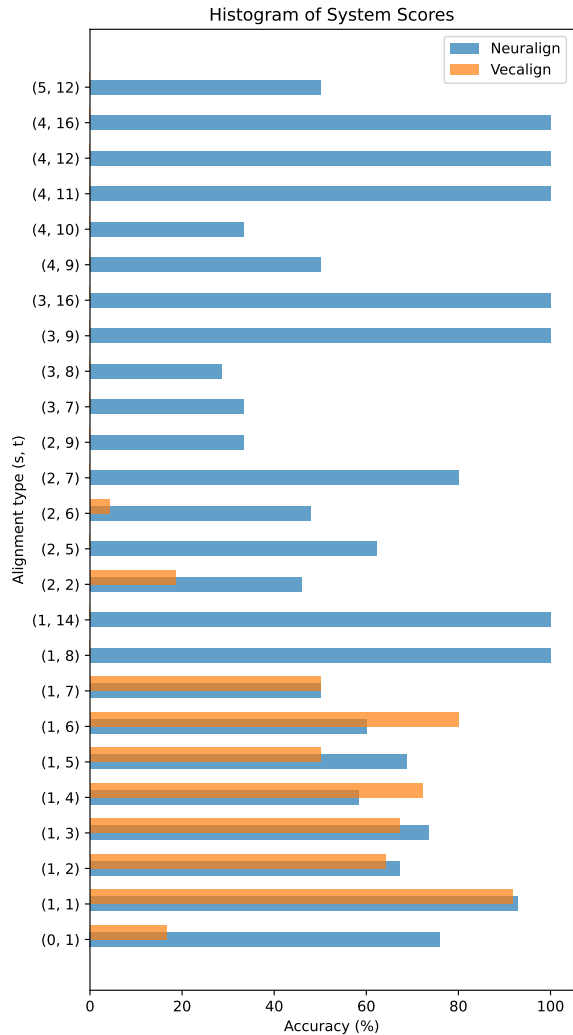

Figure 3: Histogram showing the accuracy (%) on one-to-many, many-to-one and many-to-many alignments. Alignments on the Y-axis are clustered together, e.g. $(1, 2)$ includes 1-to-2 and 2-to-1 alignments.

trast, Vecalign requires 224 seconds on average to encode both source and target documents (112 seconds for a single document) and 3 seconds to find the optimal alignment, for a total of 227 seconds. As a rough estimate, when a single GPU is adopted, CROCOALIGN can align around 1000 pairs of books and 50 million tokens in a day, compared to the 380 pairs of books and 18.5 million tokens of Vecalign.

6 Experiments on Machine Translation

Sentence alignment serves as a fundamental tool in MT systems, as it can be used for the creation of parallel datasets for training purposes. Here, we demonstrate the effectiveness of CROCOALIGN for the automatic creation of a high-quality parallel fine-tuning set that can be used to adapt an existing

MT system to specific domains. Our hypothesis is that, with a high-quality alignment, we only require a small amount of parallel sentences to significantly boost the performance of a pretrained MT model.

Experimental setup We compare the performance of a strong MT baseline when used out of the box with the performance of the same system when fine-tuned on the parallel data created with CROCOALIGN. We evaluate the impact of our data on the bilingual MT models from the OPUS-MT family, for a total of 15 bilingual models. We fine-tune each MT model on the alignments produced by CROCOALIGN-LaBSE$_B$ when applied on the validation set of the dataset introduced in Section 4.1. Finally, we report the sacreBLEU scores obtained by the MT models on the corresponding test sets, before and after fine-tuning.

Results Table 4 provides an overview of the results obtained by the OPUS-MT models when used to translate books with and without fine-tuning. We can observe a significant increase in terms of sacreBLEU across all 15 language pairs, resulting in an average improvement of 4.1 points with a minimum improvement of 2.0 points in DE-HU and DE-IT and a maximum improvement of 7.5 points in EN-FR. These results further demonstrate the high quality of our automatically-aligned data, which leads autoregressive models to better translate domain-specific texts.

7 Conclusion and Future Work

In this paper, we presented CROCOALIGN, the first fully-neural and language-agnostic architecture to perform sentence alignment in very long texts. The strength of our approach lies in its ability to create better sentence representations by taking advantage of their surrounding context in a fully-neural model equipped with a novel encoder that captures cross-sentence information, including the position and the meaning of a sentence with respect to the previous and following ones. Our experiments on sentence alignment in books – which feature extremely long contexts and present various instances of many-to-many alignments – show that CROCOALIGN outperforms the previous state of the art, i.e. Vecalign, by a significant margin across 20 language pairs (+1.6 points in F1 score on average). Moreover, we evaluate the impact that the data produced by CROCOALIGN has on the task of machine translation, and show

Language Pair	Fine-Tuning	BLEU Score	Δ
DE-ES	✗	23.9	+4.4
	✓	**28.3**	
DE-EN	✗	20.7	+2.1
	✓	**22.8**	
DE-FR	✗	9.2	+4.1
	✓	**13.3**	
DE-HU	✗	8.5	+2.0
	✓	**10.5**	
DE-IT	✗	4.2	+2.0
	✓	**6.2**	
EN-ES	✗	17.8	+6.1
	✓	**23.9**	
EN-FR	✗	33.6	+7.5
	✓	**41.1**	
EN-HU	✗	11.8	+3.0
	✓	**14.8**	
EN-IT	✗	17.9	+6.0
	✓	**23.9**	
EN-NL	✗	15.6	+6.9
	✓	**22.5**	
EN-RU	✗	20.6	+4.9
	✓	**25.5**	
ES-FR	✗	19.5	+2.6
	✓	**22.1**	
ES-IT	✗	12.7	+2.1
	✓	**14.8**	
ES-NL	✗	10.4	+5.2
	✓	**15.6**	
FR-HU	✗	8.2	+2.8
	✓	**11.0**	

Table 4: sacreBLEU results on the machine translation downstream task. Each bilingual model is evaluated with and without fine-tuning over the test split of our dataset. The fine-tuning data is produced by applying CROCOALIGN on the validation split.

that fine-tuning strong MT systems on our parallel data enables them to increase their performance in domain-specific translations by a significant margin (+4.1 points in BLEU on average). We publicly release CROCOALIGN and our alignments to the research community. We hope that our contributions can foster the development of better systems for long-text sentence alignment and the creation of better silver MT datasets, as well as renewing the interest in the task and encouraging its utilization in other downstream tasks such as extractive text summarization, paraphrase generation, and plagiarism detection.

8 Limitations

The system does not present any significant limitations. However, due to hardware constraints, we implemented two procedures during inference: one to identify the source and target batches of sentences to be aligned, and the other to correct errors resulting from multiple alignments to the same source sentence. We emphasize that these procedures are only required when the entire source and target documents do not fit within the GPU memory available. Moreover, if sufficient computational power was available, the model would not only eliminate the need for these two procedures, but it would also make use of the larger textual context in order to align all the sentences simultaneously and, possibly, more accurately.

Acknowledgements

References

Sadaf Abdul-Rauf, Mark Fishel, Patrik Lambert, Sandra Noubours, and Rico Sennrich. 2012. Extrinsic evaluation of sentence alignment systems. In *Workshop on Creating Cross-language Resources for Disconnected Languages and Styles*, pages 6–10.

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 16–23.

Peter F. Brown, Jennifer C. Lai, and Robert L. Mercer. 1991. Aligning sentences in parallel corpora. In *29th Annual Meeting of the Association for Computational Linguistics*, pages 169–176, Berkeley, California, USA. Association for Computational Linguistics.

Stanley F. Chen. 1993. Aligning sentences in bilingual corpora using lexical information. In *31st Annual Meeting of the Association for Computational Linguistics*, pages 9–16, Columbus, Ohio, USA. Association for Computational Linguistics.

Christos Christodouloupoulos and Mark Steedman. 2015. A massively parallel corpus: the bible in 100 languages. *Language resources and evaluation*, 49:375–395.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic BERT sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891, Dublin, Ireland. Association for Computational Linguistics.

Viviana Gaballo. 2012. Exploring the boundaries of transcreation in specialized translation. *ESP Across Cultures*, 9(1):95–113.

William A. Gale and Kenneth W. Church. 1993. A program for aligning sentences in bilingual corpora. *Computational Linguistics*, 19(1):75–102.

Luís Gomes and Gabriel Pereira Lopes. 2016. First steps towards coverage-based sentence alignment. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 2228–2231, Portorož, Slovenia. European Language Resources Association (ELRA).

Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. 2020. Neural crf model for sentence alignment in text simplification. *arXiv preprint arXiv:2005.02324*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Martin Kay and Martin Röscheisen. 1993. Text-translation alignment. *Comput. Linguistics*, 19:121–142.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of Machine Translation Summit X: Papers*, pages 79–86, Phuket, Thailand.

Robert C. Moore. 2002. Fast and accurate sentence alignment of bilingual corpora. In *Proceedings of the 5th Conference of the Association for Machine Translation in the Americas: Technical Papers*, pages 135–144, Tiburon, USA. Springer.

Roberto Navigli, Michele Bevilacqua, Simone Conia, Dario Montagnini, and Francesco Cecconi. 2021. Ten years of BabelNet: A survey. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 4559–4567. International Joint Conferences on Artificial Intelligence Organization. Survey Track.

NLLB team, Marta Costa-jussa, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janicec Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loïc Barrault, Gabriel Gonzalez, Prangthip Hansanti, and Jeff Wang. 2022. No language left behind: Scaling human-centered machine translation. *ArXiv*, abs/2207.04672.

Riccardo Orlando, Simone Conia, Stefano Faralli, and Roberto Navigli. 2022. Universal semantic annotator: the first unified API for WSD, SRL and semantic parsing. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2634–2641, Marseille, France. European Language Resources Association.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

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, Online. Association for Computational Linguistics.

Rico Sennrich and Martin Volk. 2011. Iterative, mt-based sentence alignment of parallel texts. In *Proceedings of the 18th Nordic conference of computational linguistics (NODALIDA 2011)*, pages 175–182.

Xuewen Shi, Heyan Huang, Ping Jian, and Yi-Kun Tang. 2021. Improving neural machine translation with sentence alignment learning. *Neurocomputing*, 420:15–26.

Brian Thompson and Philipp Koehn. 2019. Vecalign: Improved sentence alignment in linear time and space. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1342–1348, Hong Kong, China. Association for Computational Linguistics.

J. Tiedemann. 2007. Improved sentence alignment for movie subtitles. In *Proceedings of RANLP 07, Borovets, Bulgaria*. INCOMA Ltd. 2007/j.tiedemann/pub006.

Dániel Varga, Péter Halácsy, András Kornai, Viktor Nagy, László Németh, and Viktor Trón. 2007. Parallel corpora for medium density languages. *Amsterdam Studies In The Theory And History Of Linguistic Science Series 4*, 292:247.

Martin Volk, Noah Bubenhofer, Adrian Althaus, Maya Bangerter, Lenz Furrer, and Beni Ruef. 2010. Challenges in building a multilingual alpine heritage corpus. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534, Portorož, Slovenia. European Language Resources Association (ELRA).

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2017. Overview of the second BUCC shared task: Spotting parallel sentences in comparable corpora. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 60–67, Vancouver, Canada. Association for Computational Linguistics.

A Dataset Statistics

The following tables (Table 5, 6, 7) present the number of source sentences, target sentences, and annotations for each language pair in the validation, training and test set, respectively. The data highlights the variation in dataset size and annotation levels across different languages. An annotation is defined as an alignment between 1-to-1, many-to-1, 1-to-many and many-to-many source and target sentences.

Language pair	# Source	# Target	# Annotations
DE-EN	2,449	2,456	2,310
DE-ES	349	341	324
DE-FR	4,476	3,178	2,810
DE-HU	5,311	6,597	5,174
DE-IT	522	374	345
EN-ES	4,528	5,062	4,265
EN-FR	6,777	5,838	5,811
EN-HU	3,127	3,115	3,012
EN-IT	1,612	1,567	1,517
EN-NL	1,444	1,303	1,287
EN-RU	4,831	4,337	4,141
ES-FR	607	680	578
ES-HU	142	140	140
ES-IT	890	673	634
ES-NL	2,501	2,408	2,321
FR-HU	4,836	4,320	4,034
FR-IT	1,656	1,563	1,250
FR-NL	4,430	3,288	3,263
HU-IT	419	375	358
HU-NL	2,154	2,113	1,942

Table 5: Validation dataset statistics

Language pair	# Source	# Target	# Annotations
CA-DE	5,010	4,825	4,646
CA-EN	5,010	4,943	4,760
CA-HU	5,010	5,455	4,874
CA-NL	5,010	4,879	4,718
DE-EN	54,008	53,177	47,663
DE-EO	1,454	1,985	1,528
DE-ES	30,520	34,573	28,340
DE-FR	34,119	33,343	30,056
DE-HU	47,126	47,357	43,087
DE-IT	30,520	28,998	27,189
DE-NL	17,029	17,463	16,184
DE-PT	1,168	1,454	1,171
DE-RU	18,157	18,412	17,422
EL-EN	1,587	1,526	1,345
EL-ES	1,587	1,198	1,130
EL-FR	1,587	1,348	1,258
EL-HU	1,587	1,191	1,120
EN-EO	1,723	1,985	1,648
EN-ES	91,347	94,235	84,637
EN-FR	123,043	122,760	114,052
EN-HU	145,831	152,962	136,243
EN-IT	34,421	32,903	30,092
EN-NL	4,3481	39,957	37,702
EN-PL	3,832	3,284	2,976
EN-PT	1,440	1,454	1,413
EN-RU	10,677	9,786	9,279
EN-SV	3,203	3,210	3,106
EO-ES	1,986	2,077	1,754
EO-FR	1,985	1,822	1,642
EO-HU	1,985	1,994	1,694
EO-IT	1,986	1,609	1,511
EO-PT	1,701	1,454	1,300
ES-FR	55,278	52,685	49,567
ES-HU	86,511	87,431	78,344
ES-IT	36,534	30,836	28,465
ES-NL	33,709	30,248	28,759
ES-NO	3,716	4,049	3,610
ES-PT	1,787	1,454	1,343
ES-RU	21,208	18,412	16,973
FI-FR	3,758	3,937	3,556
FI-HU	3,758	4,136	3,541
FI-PL	3,758	3,284	2,960
FR-HU	88,877	95,414	84,352
FR-IT	14,418	13,883	12,773
FR-NL	36,886	36,217	34,704
FR-PL	3,937	3,284	2,976
FR-PT	1,539	1,454	1,312
FR-RU	9,672	8,843	8,284
FR-SV	3,651	3,210	3,026
HU-IT	37,409	32,155	30,741
HU-NL	48,980	43,513	41,385
HU-PL	4,136	3,284	3,006
HU-PT	1,714	1,454	1,240
HU-RU	27,017	27,255	26,219
IT-NL	3,157	2,935	2,429
IT-PT	1,319	1,454	1,220
IT-RU	18,245	18,412	17,941
IT-SV	3,130	3,210	3,010

Table 6: Training dataset statistics

Language pair	# Source	# Target	# Annotations
DE-EN	2,377	2,488	2,310
DE-ES	339	346	324
DE-FR	4,323	3,083	2,810
DE-HU	5,599	6,236	5,174
DE-IT	479	375	344
EN-ES	4,605	4,746	4,264
EN-FR	7,074	5,817	5,811
EN-HU	3,088	3,129	3,011
EN-IT	1,592	1,564	1,517
EN-NL	1,430	1,291	1,287
EN-RU	4,771	4,290	4,141
ES-FR	592	669	578
ES-HU	148	140	139
ES-IT	898	647	634
ES-NL	2,600	2,423	2,321
FR-HU	4,837	4,257	4,033
FR-IT	1,560	1,595	1,249
FR-NL	4,406	3,328	3,262
HU-IT	413	373	358
HU-NL	2,094	2,086	1,941

Table 7: Test dataset statistics