Cross-Align: Modeling Deep Cross-lingual Interactions for Word Alignment

Siyu Lai[1]*, **Zhen Yang**[2] , **Fandong Meng**[2], **Yufeng Chen**[1]†,
Jinan Xu[1] and **Jie Zhou**[2]
[1]Beijing Key Lab of Traffic Data Analysis and Mining,
Beijing Jiaotong University, Beijing, China
[2]Pattern Recognition Center, WeChat AI, Tencent Inc, China
{siyulai,chenyf,jaxu}@bjtu.edu.cn,
{zieenyang,fandongmeng,withtomzhou}@tencent.com

Abstract

Word alignment which aims to extract lexicon translation equivalents between source and target sentences, serves as a fundamental tool for natural language processing. Recent studies in this area have yielded substantial improvements by generating alignments from contextualized embeddings of the pre-trained multilingual language models. However, we find that the existing approaches capture few interactions between the input sentence pairs, which degrades the word alignment quality severely, especially for the ambiguous words in the monolingual context. To remedy this problem, we propose **Cross-Align** to model deep interactions between the input sentence pairs, in which the source and target sentences are encoded separately with the shared self-attention modules in the shallow layers, while cross-lingual interactions are explicitly constructed by the cross-attention modules in the upper layers. Besides, to train our model effectively, we propose a two-stage training framework, where the model is trained with a simple Translation Language Modeling (TLM) objective in the first stage and then finetuned with a self-supervised alignment objective in the second stage. Experiments show that the proposed Cross-Align achieves the state-of-the-art (SOTA) performance on four out of five language pairs.[1]

1 Introduction

Word alignment which aims to extract the lexicon translation equivalents between the input source-target sentence pairs (Brown et al., 1993; Zenkel et al., 2019; Garg et al., 2019; Jalili Sabet et al., 2020), has been widely used in machine translation (Och and Ney, 2000; Arthur et al., 2016; Yang et al., 2020, 2021), transfer text annotations (Fang and Cohn, 2016; Huck et al., 2019), typological analysis (Lewis and Xia, 2008), generating adversarial examples (Lai et al., 2022), etc. Statistical word aligners based on the IBM translation models (Brown et al., 1993), such as GIZA++ (Och and Ney, 2003) and FastAlign (Dyer et al., 2013), have remained popular over the past thirty years for their good performance. Recently, with the advancement of deep neural models, neural aligners have developed rapidly and surpassed the statistical aligners on many language pairs. Typically, these neural approaches can be divided into two branches: ***Neural Machine Translation (NMT) based aligners*** and ***Language Model (LM) based aligners***.

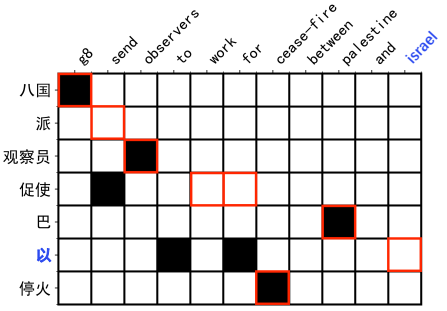


Figure 1: An example from Dou and Neubig (2021). There is a misalignment between "以" and "to" and "for". Red boxes denote the gold alignments.

NMT based aligners (Garg et al., 2019; Zenkel et al., 2020; Chen et al., 2020, 2021; Zhang and van Genabith, 2021) take alignments as a by-product of NMT systems by using attention weights to extract alignments. As NMT models are unidirectional, two NMT models (source-to-target and target-to-source) are required to obtain the final alignments, which makes the NMT based aligners less efficient. As opposed to the NMT based aligners, the ***LM based aligners*** generate alignments from the contextualized embeddings of the directionless multilingual language models. They extract contextualized embeddings from LMs and induce alignments based on the matrix of embed-

*Work done when Siyu were interning at Pattern Recognition Center, WeChat AI, Tencent Inc, China.
† Yufeng Chen is the corresponding author.
[1]The code is publicly available at: https://github.com/lisasiyu/Cross-Align

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 3715 - 3725
December 7-11, 2022

ding similarities (Jalili Sabet et al., 2020; Dou and Neubig, 2021). While achieving some improvements in the alignment quality and efficiency, we find that the existing LM based aligners capture few interactions between the input source-target sentence pairs. Specifically, SimAlign (Jalili Sabet et al., 2020) encodes the source and target sentences separately without attending to the context in the other language. Dou and Neubig (2021) further propose Awesome-Align, which considers the cross-lingual context by taking the concatenation of the sentence pairs as inputs during training, but still encodes them separately during inference.

However, the lack of interaction between the input source-target sentence pairs degrades the alignment quality severely, especially for the ambiguous words in the monolingual context. Figure 1 presents an example of our reproduced results from Awesome-Align. The ambiguous Chinese word "以" has two different meanings: 1) a preposition ("to", "as", "for" in English), 2) the abbreviation of the word "以色列" ("Israel" in English). In this example, the word "以" is misaligned to "to" and "for" as the model does not fully consider the word "Israel' in the target sentence. Intuitively, the cross-lingual context is very helpful for alleviating the meaning confusion in the task of word alignment.

Based on the above observation, we propose **Cross-Align**, which fully considers the cross-lingual context by modeling deep interactions between the input sentence pairs. Specifically, Cross-Align encodes the monolingual information for source and target sentences independently with the shared self-attention modules in the shallow layers, and then explicitly models deep cross-lingual interactions with the cross-attention modules in the upper layers. Besides, to train Cross-Align effectively, we propose a two-stage training framework, where the model is trained with the simple TLM objective (Conneau and Lample, 2019) to learn the cross-lingual representations in the first stage, and then finetuned with a self-supervised alignment objective to bridge the gap between training and inference in the second stage. We conduct extensive experiments on five different language pairs and the results show that our approach achieves the SOTA performance on four out of five language pairs.[2] Compared to the existing approaches which apply many complex training objectives, our approach is simple yet effective.

Our main contributions are summarized as follows:

- We propose Cross-Align, a novel word alignment model which utilizes the self-attention modules to encode monolingual representations and the cross-attention modules to model cross-lingual interactions.
- We propose a two-stage training framework to boost model performance on word alignment, which is simple yet effective.
- Extensive experiments show that the proposed model achieves SOTA performance on four out of five different language pairs.

2 Related Work

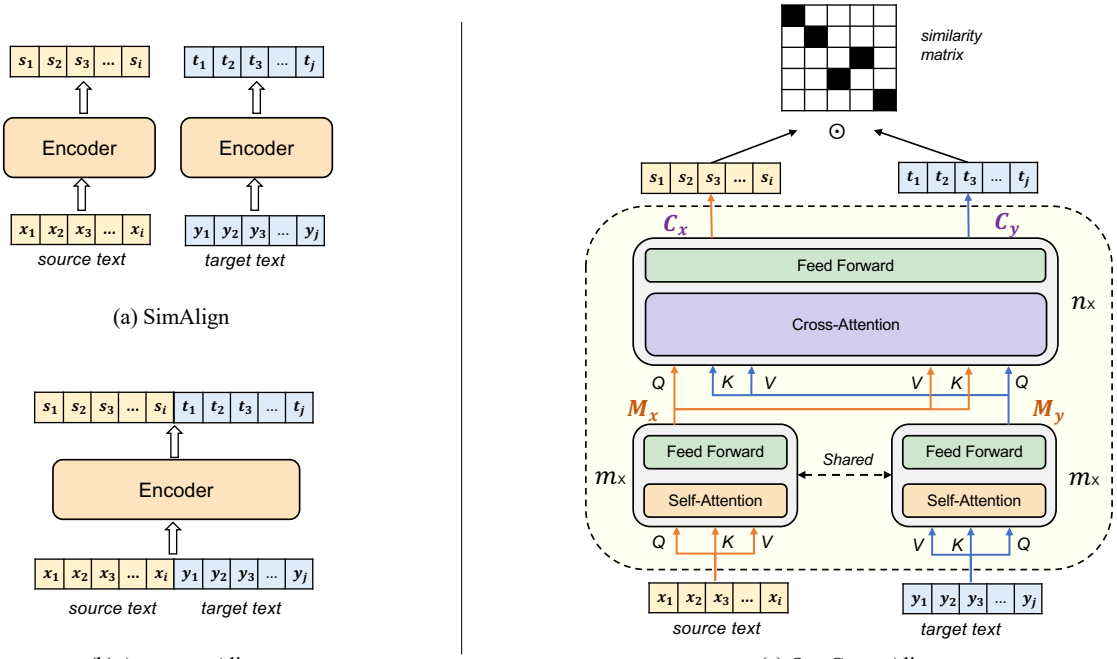
2.1 NMT based Aligner

Recently, there is a surge of interest in studying alignment based on the attention weights (Vaswani et al., 2017) of NMT systems. However, the naive attention may fails to capture clear word alignments (Serrano and Smith, 2019). Therefore, Zenkel et al. (2019) and Garg et al. (2019) extend the Transformer architecture with a separate alignment layer on top of the decoder, and produce competitive results compared to GIZA++. Chen et al. (2020) further improve alignment quality by adapting the alignment induction with the to-be-aligned target token. Recently, Chen et al. (2021) and Zhang and van Genabith (2021) propose self-supervised models that take advantage of the full context on the target side, and achieve the SOTA results. Although NMT based aligners achieve promising results, there are still some disadvantages: 1) The inherent discrepancy between translation task and word alignment is not eliminated, so the reliability of the attention mechanism is still under suspicion (Li et al., 2019); 2) Since NMT models are unidirectional, it requires NMT models in both directions to obtain final alignment, which is lack of efficiency.

2.2 LM based Aligner

Recent pre-trained multilingual language models like mBERT (Devlin et al., 2019) and XLM-R (Conneau and Lample, 2019) achieve promising results on many cross-lingual transfer tasks (Liang et al., 2020; Hu et al., 2020; Wang et al., 2022a,b). Jalili Sabet et al. (2020) prove that multilingual LMs are also helpful in word alignment task and propose SimAlign to extract alignments

[2] In Ro-En, we achieve the best performance among models in the same line, but perform a little poorer than the NMT based models which have much more parameters than ours.



(a) SimAlign

(b) Awesome-Align

(c) Our Cross-Align

Figure 2: Comparison between different LM based aligners. (a) SimAlign (Jalili Sabet et al., 2020) encodes source and target sentences separately. (b) AwesomeAlign (Dou and Neubig, 2021) concatenates source and target sentences together as inputs. (c) The proposed Cross-Align model.

from similarity matrices of contextualized embeddings without relying on parallel data (Figure 2(a)). Awesome-Align further improves the alignment quality of LMs by crafting several training objectives based on parallel data, like masked language modeling, TLM, and parallel sentence identification task. Although Awesome-Align has achieved the SOTA performance among LM based aligners, we find it still has two main problems: 1) During training, they simply concatenate the source and target sentences together as the input of self-attention module (Figure 2(b)). However, Luo et al. (2021) prove that self-attention module tends to focus on their own context, while ignores the paired context, leading to few attention patterns across languages in the self-attention module. 2) During inference, they still encode the language pairs individually, which causes the cross-lingual context unavailable when generating alignments.[3] Therefore, Awesome-Align models few interactions between cross-lingual pairs. Based on the above observation, we propose Cross-Align, which aims to model deep interactions of cross-lingual pairs to solve these problems.

[3]For Awesome-Align, concatenating the input sentence pair during inference leads to poor results compared to separately encoding. Please refer to Table 2 for comparison results.

3 Method

In this section, we first introduce the model architecture and then illustrate how we extract alignments from Cross-Align. Finally, we describe the proposed two-stage training framework in detail.

3.1 Model Architecture

As shown in Figure 2(c), Cross-Align is composed of a stack of m self-attention modules and n cross-attention modules (Vaswani et al., 2017). Given a sentence $\mathbf{x} = \{x_1, x_2, \ldots, x_i\}$ in the source language and its corresponding parallel sentence $\mathbf{y} = \{y_1, y_2, \ldots, y_j\}$ in the target language, Cross-Align first encodes them separately with the shared self-attention modules to extract the monolingual representations, and then generate the cross-lingual representations by fusing the source and target monolingual representations with the cross-attention modules. We elaborate the self-attention module and cross-attention module as follows.

Self-Attention Module. Each self-attention module contains a self-attention sub-layer and a fully connected feed-forward network (FFN). The attention function maps a query ($\mathbf{Q}$) and a set of key-value ($\mathbf{K}$-$\mathbf{V}$) pairs to an output. As for self-attention, all queries, keys and values are from the same language. Formally, the output of a self-

attention module in the l-th layer ($1 \leq l \leq m$) is calculated as:

$$\mathbf{Q} = \mathbf{H}^{l-1}\mathbf{W}_s^Q, \mathbf{K} = \mathbf{H}^{l-1}\mathbf{W}_s^K, \mathbf{V} = \mathbf{H}^{l-1}\mathbf{W}_s^V, \tag{1}$$

$$\hat{\mathbf{H}}^l = \mathrm{LN}(\mathrm{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_k}})\mathbf{V} + \mathbf{H}^{l-1}), \tag{2}$$

$$\mathbf{H}^l = \mathrm{LN}(\mathrm{FFN}(\hat{\mathbf{H}}^l) + \hat{\mathbf{H}}^l), \tag{3}$$

where $\mathbf{W}_s^Q, \mathbf{W}_s^K, \mathbf{W}_s^V$ are parameter matrices of the self-attention module, $\mathbf{H}^{l-1}$ is output from previous layer, $\mathrm{LN}(\cdot)$ refers to the Layer-Normalization operation. With the above stacked m self-attention modules, we get the monolingual representations $\mathbf{M}_x$ and $\mathbf{M}_y$ when $\mathbf{H}^0$ is set to the embeddings of $\mathbf{x}$ and $\mathbf{y}$, respectively.

Cross-Attention Module. Although the self-attention modules can effectively encode monolingual information, the interactive information between $\mathbf{x}$ and $\mathbf{y}$ is not explored. Recently, cross-attention modules have been successfully used to learn cross-modal interactions in multi-modal tasks (Wei et al., 2020; Li et al., 2021), which motivates us to leverage cross-attention modules for exploring cross-lingual interactions in word alignment.

Specifically, each cross-attention module contains a cross-attention sub-layer and an FFN network. Different from self-attention, the queries of cross-attention come from one language, while keys and values come from the other language. Formally, the output of a cross-attention module in the l-th layer ($m < l \leq m + n$) is computed as:

$$\mathbf{Q}_x = \mathbf{H}_x^{l-1}\mathbf{W}_c^Q, \mathbf{K}_x = \mathbf{H}_y^{l-1}\mathbf{W}_c^K, \mathbf{V}_x = \mathbf{H}_y^{l-1}\mathbf{W}_c^V, \tag{4}$$

$$\mathbf{Q}_y = \mathbf{H}_y^{l-1}\mathbf{W}_c^Q, \mathbf{K}_y = \mathbf{H}_x^{l-1}\mathbf{W}_c^K, \mathbf{V}_y = \mathbf{H}_x^{l-1}\mathbf{W}_c^V, \tag{5}$$

$$\hat{\mathbf{H}}_x^l = \mathrm{LN}(\mathrm{softmax}(\frac{\mathbf{Q}_x\mathbf{K}_x^T}{\sqrt{d_k}})\mathbf{V}_x + \mathbf{H}_x^{l-1}), \tag{6}$$

$$\hat{\mathbf{H}}_y^l = \mathrm{LN}(\mathrm{softmax}(\frac{\mathbf{Q}_y\mathbf{K}_y^T}{\sqrt{d_k}})\mathbf{V}_y + \mathbf{H}_y^{l-1}), \tag{7}$$

$$\mathbf{H}_x^l = \mathrm{LN}(\mathrm{FFN}(\hat{\mathbf{H}}_x^l) + \hat{\mathbf{H}}_x^l), \tag{8}$$

$$\mathbf{H}_y^l = \mathrm{LN}(\mathrm{FFN}(\hat{\mathbf{H}}_y^l) + \hat{\mathbf{H}}_y^l), \tag{9}$$

where $\mathbf{W}_c^Q, \mathbf{W}_c^K, \mathbf{W}_c^V$ are parameter matrices of the cross-attention module, $\mathbf{H}_x^{l-1}$ is output from the previous layer corresponding to $\mathbf{x}$ and $\mathbf{H}_y^{l-1}$ is output from the previous layer corresponding to $\mathbf{y}$. With the above stacked n cross-attention modules, we get the cross-lingual representation $\mathbf{C}_x$ and $\mathbf{C}_y$ by setting $\mathbf{H}_x^m$ and $\mathbf{H}_y^m$ to $\mathbf{M}_x$ and $\mathbf{M}_y$, respectively.

3.2 Alignments Extraction

The proposed Cross-Align aims to extract alignments from the input sentence pair $\mathbf{x}$ and $\mathbf{y}$, and we illustrate the extraction method as follows.

Firstly, we extract the hidden states $\mathbf{s} = \{s_1, s_2, \ldots, s_i\}$ and $\mathbf{t} = \{t_1, t_2, \ldots, t_j\}$ for $\mathbf{x}$ and $\mathbf{y}$ respectively. Secondly, we get a similarity matrix $S_{I\times J}$ by computing the dot products between $\mathbf{s}$ and $\mathbf{t}$ and apply the $softmax$ normalization to convert $S_{I\times J}$ into the source-to-target probability matrices $P^f_{I\times J}$ and target-to-source probability matrices $P^b_{I\times J}$. After that, we obtain the final alignment matrix $G_{I\times J}$ by taking the intersection of the two matrices following Dou and Neubig (2021):

$$G = (P^f > \tau) * (P^b > \tau), \tag{10}$$

where τ is a threshold. $G_{ij} = 1$ means x_i and y_j are aligned. Note that the current alignments generated from Cross-Align are on BPE-level. We follow previous work to convert BPE-level alignments to word-level alignments (Dou and Neubig, 2021; Zhang and van Genabith, 2021) by adding an alignment between a source word and a target word if any parts of these two words are aligned.

3.3 Two-stage Training Framework

This sub-section describes the proposed two-stage training framework. In the first stage, the model is trained with TLM to learn the cross-lingual representations. After the first training stage, the model is then finetuned with a self-supervised alignment objective to bridge the gap between the training and inference.

Stage1: Translation Language Modeling. TLM is a simple training objective first proposed by Conneau and Lample (2019) for learning cross-lingual representations of LMs. Since Cross-Align aims to learn interactions between the input sentence pairs, TLM is a suitable objective for effectively training Cross-Align. Different from Conneau and Lample (2019) which train TLM objective based on the self-attention modules, Cross-Align applies the cross-attention modules to enforce the model to infer the masked tokens based on the cross-lingual representations $\mathbf{C}_x$ and $\mathbf{C}_y$, encouraging deep interactions between the input sentence pair.

Following the previous works (Devlin et al., 2019; Conneau and Lample, 2019), we choose 15% of the token positions randomly for both $\mathbf{x}$ and $\mathbf{y}$. For each chosen token, we replace it with

the [MASK] token 80% of the time, a random token 10% of the time, and remain token 10% of the time. The model is trained to predict the original masked words based on the bilingual context. Thus, the training objective can be formulated as:

$$\begin{aligned}\mathcal{L}_{TLM} = &-\log P(\mathbf{x}|\hat{\mathbf{x}},\hat{\mathbf{y}};\theta_{\mathbf{s}},\theta_{\mathbf{c}}) \\ &-\log P(\mathbf{y}|\hat{\mathbf{x}},\hat{\mathbf{y}};\theta_{\mathbf{s}},\theta_{\mathbf{c}}),\end{aligned} \tag{11}$$

where $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ are the masked sentences for $\mathbf{x}$ and $\mathbf{y}$ respectively, $\theta_{\mathbf{s}}$ denotes all the parameters of the m self-attention modules, and $\theta_{\mathbf{c}}$ represents the parameters of n cross-attention modules.

Stage2: Self-Supervised Alignment. In the first training stage, the model is trained with TLM by feeding the masked sentence pairs as input. However, the model is required to extract the alignments from the original sentence pairs during inference. Therefore, there is a gap between the training and inference which may limit the alignment quality. To solve this problem, we propose a self-supervised alignment (SSA) objective in the second stage. SSA takes the alignments generated by the model trained in the first stage as golden labels and trains the model with the alignment task directly in this stage.

As previous studies (Jalili Sabet et al., 2020; Dou and Neubig, 2021) have shown that the middle layer of LM always has better alignment performance than the last layer, we take the c-th layer of Cross-Align as the alignment layer to train the alignment objective, where c ($1 \leq c \leq m+n$) is a hyper-parameter chosen from the experimental results.[4] From the alignment layer of the first-stage model, we extract the 0-1 alignment labels $G_{I\times J}$ according to extraction method described in Section 3.2.[5] $P^f_{I\times J}$ and $P^b_{I\times J}$ denotes the source-to-target and target-to-source probability matrices extracted from the alignment layer of the current model, respectively. Following Garg et al. (2019), we optimize the alignment objective by minimizing the cross-entropy loss:

$$\begin{aligned}\mathcal{L}_{SSA} = &-\frac{1}{I}\sum_{i=1}^{I}\sum_{j=1}^{J} G_{ij}\log(P^f_{ij}) \\ &-\frac{1}{J}\sum_{j=1}^{J}\sum_{i=1}^{I} G_{ij}\log(P^b_{ij}).\end{aligned} \tag{12}$$

To alleviate the catastrophe of forgetting knowledge learned by the TLM, we only finetune the alignment layer and freeze other layers of the model. With the SSA objective, Cross-Align directly learns the word alignment task in the alignment layer instead of the masked language modeling, making the training consistent with the inference process. During inference, we extract hidden states from the alignment layer and get the final alignments.

Dataset	De-En	En-Fr	Ro-En	Zh-En	Ja-En
training	1.9M	1.1M	447k	1.2M	442k
dev	-	-	-	450	653
test	508	447	248	450	582

Table 1: The number of sentences in each dataset.

4 Experimental Settings

In this section, we first describe the details of datasets and implementation, then present the baselines, and finally introduce evaluation measures.

4.1 Datasets

We conduct our experiments on five publicly available datasets, including German-English (De-En), English-French (En-Fr), Romanian-English (Ro-En), Chinese-English (Zh-En), and Japanese-English (Ja-En). The training sets only contain the parallel sentences without word alignment labels, the development and test sets contain parallel sentences with gold word alignment labels annotated by experts. Table 1 gives the detailed data statistics. Considering that De-En, En-Fr, and Ro-En do not have development sets, we use Zh-En development sets to tune the hyper-parameters for them.

4.2 Implementation Details

Our implementation is based on the code base released by Awesome-Align.[6] We randomly initialize the parameters of cross-attention modules and leverage the pre-trained mBERT-base (Devlin et al.,

[4]The analysis about the alignment layer is conducted in Section 5.2.

[5]Now the alignment labels are on word level, while SSA objective is on BPE level, so we convert labels back to BPE-level as follows: a source BPE token is aligned to a target BPE token if their corresponding source word and target word are aligned. Besides, a target BPE tokens will be aligned with [CLS] token, if its corresponding target word is not aligned with any source word.

[6]https://github.com/neulab/awesome-align

Method	De-En	En-Fr	Ro-En	Zh-En	Ja-En
Statistic Based					
FastAlign (Dyer et al., 2013)	26.2†	10.5†	31.4†	23.7†	51.1†
GIZA++ (Och and Ney, 2003)	18.9†	5.5†	26.6†	19.4†	48.0†
Neural Machine Translation Based					
Zenkel et al. (2019)	21.2	10.0	27.6	-	-
Garg et al. (2019)	20.2	7.7	26.0	22.5†	49.8†
Zenkel et al. (2020)	16.3	5.0	23.4	-	-
SHIFT-AET (Chen et al., 2020)	15.4	4.7	21.2	18.6†	44.3†
Zhang and van Genabith (2021)	14.3	6.7	**18.5**	-	-
MASK-ALIGN (Chen et al., 2021)	14.4	4.4	19.5	13.8	43.5†
Multilingual Language Model Based					
SimAlign (Jalili Sabet et al., 2020)	18.8	7.6	27.2	21.6†	46.6
Awesome-Align (Dou and Neubig, 2021)	15.6	4.4	23.0	12.9†	38.4
Awesome-Align (*concatenation*)	16.8†	4.7†	23.2†	14.2†	39.3†
Cross-Align (ours)	**13.6**	**3.4**	20.9	**10.1**	**35.4**

Table 2: AER on the test sets with different alignment methods. The lower AER, the better performance. We highlight the best results for each language pair in **bold**. To make a fair comparison, we only report the results for all baselines under bilingual settings and without guidance from external word alignment tools. 'Awesome-Align (*concatenation*)' means the source and target sentences are concatenated as inputs during inference. '†' denotes the re-implement results based on their released code for those results not reported in the original paper.

2019) to initialize the rest parameters of our Cross-Align. The AdamW (Loshchilov and Hutter, 2019) is used as the optimizer, and the learning rate is set to 5e-4 and 1e-5 for the two stages of training, respectively. The batch size per GPU is set to 12 and gradient accumulation steps is set to 4. All models are trained on 8 NVIDIA Tesla V100 (32GB) GPUs. We train 2 epochs for each language pair in the first stage and then finetune 1 epoch in the second stage. The number of self-attention layers m and cross-attention layers n are set to 10 and 2, respectively. The alignment layer c is set to 11. In the first stage, the threshold of extraction τ is set to 0.001. In the second stage, τ is set to 0.15.

4.3 Baselines

To test the effectiveness of Cross-Align, we take the current three types of aligners as baselines.

Statistic based methods:

- FastAlign (Dyer et al., 2013) and GIZA++ (Och and Ney, 2003) are two popular statistical aligners that are implementations of IBM model.

NMT based methods:

- Zenkel et al. (2019) and Zenkel et al. (2020) propose to add an extra attention layer on top of NMT model which produces translations and alignment simultaneously.
- Garg et al. (2019) propose a multi-task framework to align and translate with transformer models jointly.
- SHIFT-AET: Chen et al. (2020) induce alignments when the to-be-aligned target token is the decoder input instead of the output.
- Zhang and van Genabith (2021) predict the target word based on the source and both left-side and right-side target context to produce attention.
- MASK-ALIGN: Chen et al. (2021) proposed a self-supervised word alignment model that takes advantage of the full context on the target side.

LM based methods:

- SimAlign: Jalili Sabet et al. (2020) extract alignment from multilingual pre-trained language models without using parallel training data.
- Awesome-Align: Dou and Neubig (2021) further finetune multilingual pre-trained language models on parallel corpora to get better alignments

4.4 Evaluation Measures

Alignment Error Rate (AER) is the standard evaluation measure for word alignment (Och and Ney, 2003). The quality of an alignment A is computed

Objective	De-En	En-Fr	Ro-En	Zh-En	Ja-En
None	59.5	38.1	82.1	73.6	83.0
+TLM	15.3	4.3	22.1	12.5	37.1
++SSA	13.6	3.4	20.9	10.1	35.4

Table 3: Ablation studies on the two-stage training objective. 'None' means the naive Cross-Align without further training on parallel corpus. '+TLM' means training Cross-Align on TLM objective. '++SSA' denotes further finetuned on SSA objective after TLM.

by:

$$\text{AER} = 1 - \frac{|A \cap S| + |A \cap P|}{|A| + |S|}, \qquad (13)$$

where S (sure) are unambiguous gold alignments and P (possible) are ambiguous gold alignments.

5 Results and Analysis

5.1 Main Results

Table 2 compares the performance of Cross-Align against statistical aligners, NMT based aligners, and LM based aligners. We can see that Cross-Align significantly outperforms the statistical method GIZA++ by 2.1~12.6 AER points across different language pairs. Compared to other LM based aligners, Cross-Align also achieves substantial improvement on all datasets. For example, on the Ja-En dataset, Cross-Align achieves 3.0 AER points improvement compared to Awesome-Align, demonstrating that modeling cross-lingual interactions based on the bilingual context is crucial for improving alignment quality. Compared to the strong NMT baselines with more parameters, we find Cross-Align still achieves the best results on all language pairs except Ro-En. We suppose the reason is that the parameters of cross-attention modules are initialized randomly, and the data size of Ro-En is too small to sufficiently train these parameters, resulting in unsatisfactory results compared to NMT based methods. We tried to use the self-attention parameters of mBERT to initialize it, but the results are not as good as random initialization. We will investigate the word alignment on low-resource language pairs in future work.

5.2 Analysis

Ablation Study. To understand the importance of the two-stage training objective, we conduct an ablation study by training multiple versions of the alignment models with some training stages removed. For all models, we extract the alignments on the alignment layer. The experimental results are shown in Table 3. From Table 3, we can find that the naive Cross-Align without training on the parallel corpus achieves very bad performance (see line "None" in Table 3). This is mainly because that the cross-attention modules are initialized randomly. TLM objective plays a critical role in training Cross-Align since it greatly improves the quality of alignment across all language pairs (see line"+TLM" in Table 3). In the second stage, the SSA objective further improves the performance by 0.9~2.4 AER points (see line "++SSA"). This shows that bridging the gap between the training and inference is helpful to the final alignment performance.

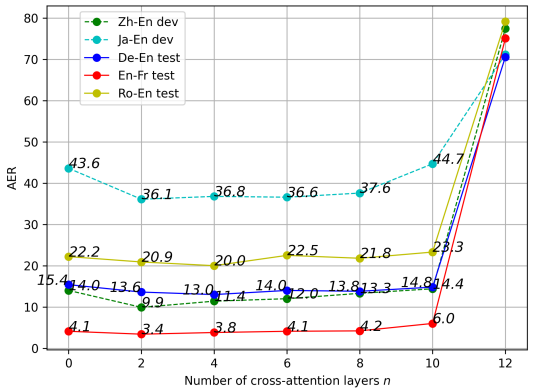
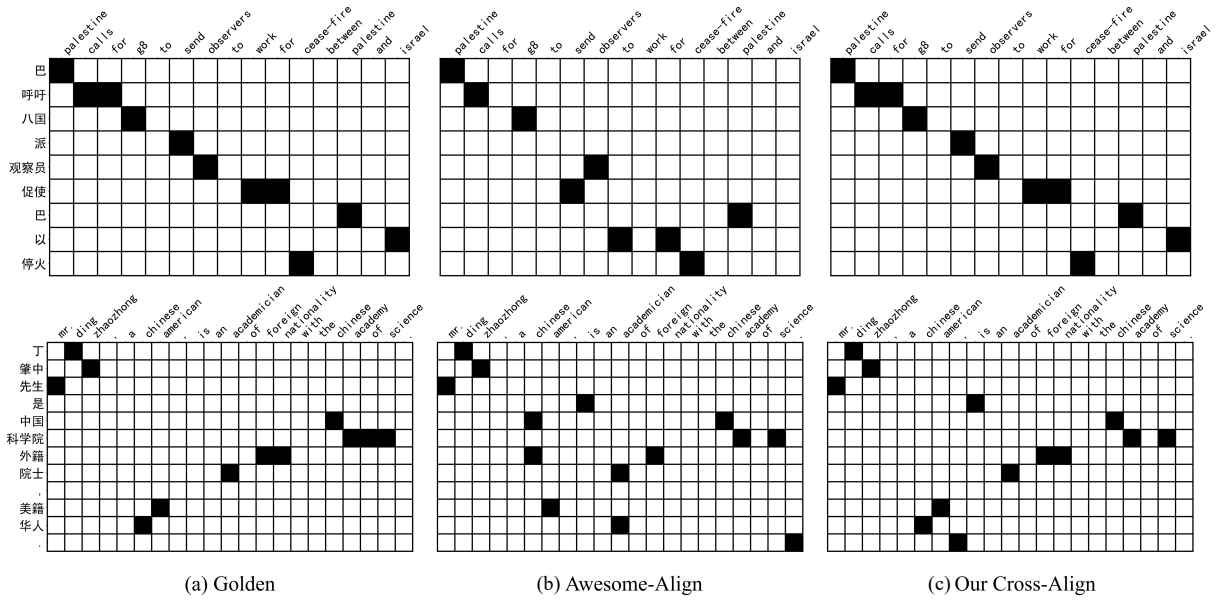


Figure 3: Word alignment performance with different number of cross-attention layers n.

Number of Cross-Attention Layers. Since the self-attention and cross-attention modules play different roles in the final alignments, we are curious about how the number of cross-attention layers affects the final alignment performance. We investigate this problem by studying the alignment performance with different n, where n ranges from 0 to 12 with an interval of 2. Meanwhile, we keep $m + n = 12$ to ensure that Cross-Align has the same number of layers as mBERT. Figure 3 shows the AER results on the dev sets with different n. For a more comprehensive analysis, we also show the results on the test sets for language pairs without dev sets. As shown in Figure 3, Cross-Align degenerates into the separate encoding framework when $n = 0$, achieving bad alignment performance. This shows that modeling the cross-lingual interactions is very helpful for enhancing the alignment performance. Additionally, when $n = 12$, the performance drops sharply, which shows that



(a) Golden (b) Awesome-Align (c) Our Cross-Align

Figure 4: Two examples from Zh-En alignment test set. (a) Gold alignments. (b) Results of Awesome-Align. (c) Results of Cross-Align.

the monolingual representations built by the self-attention modules are necessary for the following cross-attention modules to generate reliable cross-lingual representations. Almost all of the language pairs achieve the best performance when n is set around 2 and there is a trade-off between the self-attention and cross-attention module layers.

Alignment Layer. After the training of TLM, we need to decide the alignment layer c used to generate alignments. Figure 5 shows the AER results with c varying from 0 to 12. We observe that Cross-Align obtains the best performance when c is set around 11. This observation is consistent with previous studies (Jalili Sabet et al., 2020; Conneau et al., 2020). For Cross-align, the context representations in the lower self-attention layers are too language-specific to achieve high-quality alignment performance. In the upper cross-attention layers, the contextual representations are too specialized in the masked language modeling. The contextualized representations in the middle of cross-attention layers contain rich cross-lingual knowledge that help generate high-quality alignments.

5.3 Case Study

In Figure 4, we present two examples from different alignment methods on Zh-En test set. In the first example, Cross-Align correctly aligns the ambiguous Chinese word "以" to "Israel" and "促使" to "work for" based on the bilingual context, but

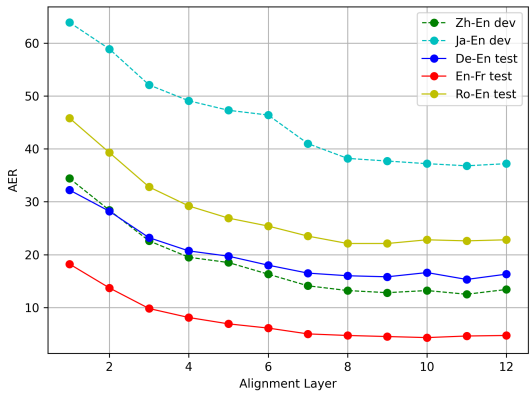


Figure 5: Word alignment performance across different alignment layers of Cross-Align in the first stage.

Awesome-Align does not. In the second example, there are two "chinese" in the target sentence with different meanings. Due to lack of cross-lingual context, Awesome-Align could not distinguish the difference and wrongly aligns "中国" to both "chinese", but Cross-Align gives correct alignments for them. It demonstrates that learning interaction knowledge between the source-target sentence pairs is beneficial to word alignment.

6 Conclusion

This paper presents a novel LM based aligner named Cross-Align, which models deep interactions between the input sentence pairs. Cross-Align first encodes the source and target sentences sep-

arately with the shared self-attention modules in the shallow layers, then explicitly constructs cross-lingual interactions with the cross-attention modules in the upper layers. Additionally, we propose a simple yet effective two-stage training framework, where the model is first trained with a simple TLM objective and then finetuned with a self-supervised alignment objective. Experimental results show that Cross-Align achieves new SOTA results on four out of language pairs. In future work, we plan to improve the alignment quality on more low-resource language pairs.

Limitations

Although the proposed Cross-Align has achieved promising results, we find it still has two main limitations. Firstly, Cross-Align has limited performance in low-resource language pairs like Ro-En and Ja-En, as shown in Table 2. We hypothesize the reason is that the cross-attention modules of Cross-Align are randomly initialized, so it needs a large number of data to train. We tried to use the self-attention parameters of mBERT to initialize it, but the results are not as good as random initialization. Secondly, we find current LM based aligners including Cross-Align have bad performance for phrase alignments. As shown in the second example in Figure 4, "academy of science" is a phrase that should be aligned to the Chinese word "科学院", but Cross-Align only aligns part of it. It is because Cross-Align generates subword-level alignments without considering the word-level and phrase-level information. In future work, we will investigate these two limitations and further improve the quality of alignments.

Acknowledgements

The research work descried in this paper has been supported by the National Key R&D Program of China (2020AAA0108001) and the National Nature Science Foundation of China (No. 61976016, 61976015, and 61876198). The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

References

Philip Arthur, Graham Neubig, and Satoshi Nakamura. 2016. Incorporating discrete translation lexicons into neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1557–1567, Austin, Texas. Association for Computational Linguistics.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Chi Chen, Maosong Sun, and Yang Liu. 2021. Maskalign: Self-supervised neural word alignment. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4781–4791, Online. Association for Computational Linguistics.

Yun Chen, Yang Liu, Guanhua Chen, Xin Jiang, and Qun Liu. 2020. Accurate word alignment induction from neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 566–576.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alexis Conneau, Shijie Wu, Haoran Li, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Emerging cross-lingual structure in pretrained language models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6022–6034, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2112–2128, Online. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Meng Fang and Trevor Cohn. 2016. Learning when to trust distant supervision: An application to low-resource POS tagging using cross-lingual projection.

In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 178–186, Berlin, Germany. Association for Computational Linguistics.

Sarthak Garg, Stephan Peitz, Udhyakumar Nallasamy, and Matthias Paulik. 2019. Jointly learning to align and translate with transformer models. 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 4453–4462, Hong Kong, China. Association for Computational Linguistics.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR.

Matthias Huck, Diana Dutka, and Alexander Fraser. 2019. Cross-lingual annotation projection is effective for neural part-of-speech tagging. In *Proceedings of the Sixth Workshop on NLP for Similar Languages, Varieties and Dialects*, pages 223–233, Ann Arbor, Michigan. Association for Computational Linguistics.

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. SimAlign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1627–1643, Online. Association for Computational Linguistics.

Siyu Lai, Zhen Yang, Fandong Meng, Xue Zhang, Yufeng Chen, Jinan Xu, and Jie Zhou. 2022. Generating authentic adversarial examples beyond meaning-preserving with doubly round-trip translation. *arXiv preprint arXiv:2204.08689*.

William D. Lewis and Fei Xia. 2008. Automatically identifying computationally relevant typological features. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-II*.

Junnan Li, Ramprasaath Selvaraju, Akhilesh Gotmare, Shafiq Joty, Caiming Xiong, and Steven Chu Hong Hoi. 2021. Align before fuse: Vision and language representation learning with momentum distillation. In *Advances in Neural Information Processing Systems*, volume 34, pages 9694–9705. Curran Associates, Inc.

Xintong Li, Guanlin Li, Lemao Liu, Max Meng, and Shuming Shi. 2019. On the word alignment from neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1293–1303, Florence, Italy. Association for Computational Linguistics.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.

I. Loshchilov and F. Hutter. 2019. Decoupled weight decay regularization. In *Proceedings of the International Conference on Learning Representations*.

Fuli Luo, Wei Wang, Jiahao Liu, Yijia Liu, Bin Bi, Songfang Huang, Fei Huang, and Luo Si. 2021. VECO: Variable and flexible cross-lingual pre-training for language understanding and generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3980–3994, Online. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 440–447, Hong Kong. Association for Computational Linguistics.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. Association for Computational Linguistics.

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

Jiaan Wang, Fandong Meng, Ziyao Lu, Duo Zheng, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2022a. Clidsum: A benchmark dataset for cross-lingual dialogue summarization. *arXiv preprint arXiv:2202.05599*.

Jiaan Wang, Fandong Meng, Duo Zheng, Yunlong Liang, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2022b. A survey on cross-lingual summarization. *arXiv preprint arXiv:2203.12515*.

Xi Wei, Tianzhu Zhang, Yan Li, Yongdong Zhang, and Feng Wu. 2020. Multi-modality cross attention network for image and sentence matching. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10941–10950.

Zhen Yang, Bojie Hu, Ambyera Han, Shen Huang, and Qi Ju. 2020. CSP:code-switching pre-training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2624–2636, Online. Association for Computational Linguistics.

Zhen Yang, Yingxue Zhang, Ernan Li, Fandong Meng, and Jie Zhou. 2021. Wets: A benchmark for translation suggestion. *arXiv preprint arXiv:2110.05151*.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2019. Adding interpretable attention to neural translation models improves word alignment. *arXiv preprint arXiv:1901.11359*.

Thomas Zenkel, Joern Wuebker, and John DeNero. 2020. End-to-end neural word alignment outperforms GIZA++. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1605–1617, Online. Association for Computational Linguistics.

Jingyi Zhang and Josef van Genabith. 2021. A bidirectional transformer based alignment model for unsupervised word alignment. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 283–292, Online. Association for Computational Linguistics.