# Using Optimal Transport as Alignment Objective for fine-tuning Multilingual Contextualized Embeddings

**Sawsan Alqahtani, Garima Lalwani, Yi Zhang, Salvatore Romeo, Saab Mansour**
AWS, Amazon AI
sawsa,glalwani,yizhngn,romeosr,saabm@amazon.com

## Abstract

Recent studies have proposed different methods to improve multilingual word representations in contextualized settings including techniques that align between source and target embedding spaces. For contextualized embeddings, alignment becomes more complex as we additionally take context into consideration. In this work, we propose using Optimal Transport (OT) as an alignment objective during fine-tuning to further improve multilingual contextualized representations for downstream cross-lingual transfer. This approach does not require word-alignment pairs prior to fine-tuning that may lead to sub-optimal matching and instead learns the word alignments within context in an unsupervised manner. It also allows different types of mappings due to soft matching between source and target sentences. We benchmark our proposed method on two tasks (XNLI and XQuAD) and achieve improvements over baselines as well as competitive results compared to similar recent works.

## 1 Introduction

Contextualized word embeddings have advanced the state-of-the-art performance in different NLP tasks (Peters et al., 2018; Howard and Ruder, 2018; Devlin et al., 2019). Similar advancements have been made for languages other than English using models that learn cross-lingual word representations leveraging monolingual and/or parallel data (Devlin et al., 2019; Conneau et al., 2020; Artetxe et al., 2020). Such cross-lingual ability helps in mitigating the lack of abundant data (labelled or unlabelled) and computational resources for languages other than English, with lesser cost. Yet, there exists a challenge for improving multilingual representations and cross-lingual transfer learning, especially for low resource languages. Recent studies proposed different techniques to improve multilingual representations in contextualized settings with additional objectives such as translation

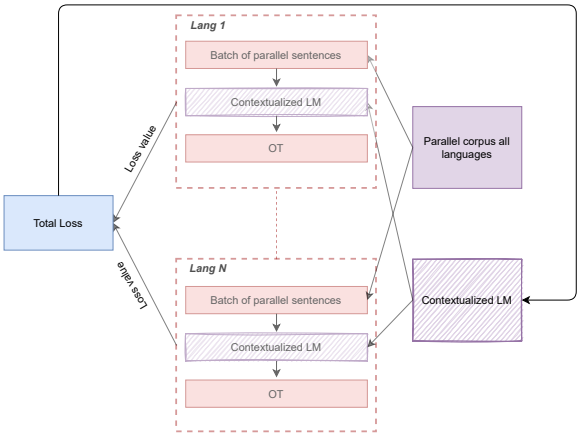


Figure 1: One iteration of fine-tuning a contextualized language model (LM) with optimal transport (OT) as the loss value.

language modeling (Lample and Conneau, 2019), integrating language and task adapters (Pfeiffer et al., 2020b), and applying alignment techniques in the embedding spaces (Cao et al., 2020; Wu and Dredze, 2020).

Previous studies concerning alignment in the embedding space show promising directions to improve cross-lingual transfer abilities for low resource languages (Aldarmaki et al., 2018; Schuster et al., 2019; Wang et al., 2019; Cao et al., 2020). The objective is to align source and target language representations into the same embedding space, for instance by encouraging similar words to be closer to each other (e.g. *cat* in English and *Katze* in German) with least cost in terms of data and computational resources. Such methods require some form of cross-lingual signal, such as alignment in non-contextualized embeddings, mainly utilize bilingual/multilingual lexicon that have been learned with unsupervised or supervised techniques (Mikolov et al., 2013; Smith et al., 2017; Aldarmaki et al., 2018). However, when it comes to contextualized embeddings, alignment becomes more complex as we additionally utilize context (e.g. *"match"* in *"Your shoes don't match clothes"*

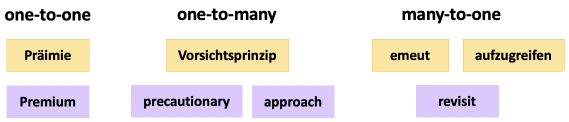


Figure 2: Examples of word alignments between English and German.

is similar to word *"passen"* in *"Ihre Schuhe passen nicht zu Kleidung"* but not to "match" in *"Hast du das Cricket Match gesehen?"*). In this work, we use only parallel sentences as an informative cross-lingual training signal.

Along these lines, previous studies mainly followed two approaches: (1) rotation based techniques with the Procrustes objective where the source embedding space is rotated to match that of the target (Wang et al., 2019; Aldarmaki and Diab, 2019); (2) fine-tuning the pre-trained language model (LM) with explicit alignment objectives such that similar words in parallel sentences are closer in representation space (Cao et al., 2020; Wu and Dredze, 2020; Hu et al., 2021a). Fine-tuning with alignment objective function provides simple yet effective and promising solution to improve contextualized word representations, especially for low resource languages. As opposed to rotation based approaches which require generating a transformation matrix for each language pair of interest, the alignment objective allows simultaneous learning from multiple languages.

Majority of previous studies concerning fine-tuning with alignment objective start with pre-collected word pairs generated using unsupervised or supervised methods (e.g. fastAlign (Dyer et al., 2013a)) which aligns words in source and target sentences based on semantics, and subsequently applies some heuristics to obtain one-to-one word alignments. However, this leads to losing other word relationships (e.g. many-to-one) which exist in some language pairs (Figure 2)

Inspired by the limitations of previous works, we propose the use of optimal transport (OT henceforward) to transfer knowledge across languages and improve multilingual word representation for cross-lingual transfer in zero-shot setting. This method learns word alignments while fine-tuning the pre-trained representations in an end-to-end fashion. As opposed to previous studies, this eliminates the need for pre-collected word pairs and allows many-to-many mappings between source and target words. Furthermore, our approach directly utilizes the continuous representation of contextualized word embeddings for alignment which helped broaden the scope of alignments to include additional linguistic information embedded in the LM (e.g. semantic and syntactic structure).

Specifically, we optimize a regularized variant of OT, i.e. Sinkhorn divergence (Feydy et al., 2019), on parallel sentences and use that as a guidance to fine-tune the pre-trained LM. We learn several independent OT mappings per language pair, each guiding the model to further shift contextualized word embeddings in the source language towards the ones in the target language (refer to Figure 1). Compared to the baseline mBERT, we obtain improvements of 1.9% and 1.3% F1 on average in XNLI (Conneau et al., 2018) and XQuAD (Rajpurkar et al., 2016; Artetxe et al., 2020) benchmarks, respectively.

Before we dive deep into our method (Section 4), we briefly describe OT in Section 2 and related work in Section 3. We discuss the experimental setup, results, analysis and finally conclusion in Sections 5, 6, 7 and 8 respectively. Our contribution is mainly three-fold:

- We propose the use of OT to align source and target embeddings in an unsupervised fashion eliminating the need for pre-collected one-to-one word pairs,
- We use OT within the space of contextual embeddings in an end-to-end manner by leveraging loss from OT optimization for fine-tuning contextualized embeddings.
- We show improvements compared to the baselines and competitive results compared to more recent works evaluating on XNLI and XQuAD.

## 2 Optimal Transport in NLP

Optimal transport (OT) provides powerful tools to compare between different probability distributions and learn the similarities/differences to move the mass from source to target distributions (Peyré and Cuturi, 2019). It has a strict requirement where source distribution must be completely transferred to target distribution making it rigid for machine learning based models. Santambrogio (2015) relaxed this constraint by allowing masses to be partially transferred to more than one point in the target distribution resulting in development of different regularized OT variants that improve both computational and statistical properties (Cuturi, 2013; Schmitzer, 2019; Alaya et al., 2019).

Regularized OT variants led to the increased adoption of OT in a wide range of areas such as graph applications, computer vision, and natural language processing (Vayer et al., 2019; Xu et al., 2019; B'ecigneul et al., 2020; Singh et al., 2019; Alvarez-Melis et al., 2020). Downstream applications utilize OT properties to obtain either of the following outputs (Flamary et al., 2021): (1) optimal mapping (or transport) that aligns between points in the two distributions for applications like graph matching (Vayer et al., 2019) or word translations (Alvarez-Melis and Jaakkola, 2018); (2) the optimal values (Wasserstein distance) which is computed based on the optimal mappings and subsequently guide the model learning for applications like document similarity (Kusner et al., 2015) and machine translation (Chen et al., 2019). Similar to this line of work, we use optimal values for guiding model learning.

**Optimal transport for alignment:** OT has been used as a fully unsupervised algorithm to align points between two distributions making use of the linguistic and structural similarity for applications like bilingual lexical induction (Zhang et al., 2017; Schmitz et al., 2018) and word translations (Alvarez-Melis and Jaakkola, 2018; Alvarez-Melis et al., 2019; Grave et al., 2018). For instance, Zhang et al. (2017) use OT to induce translations for a source word from the target language in bilingual lexicon induction whereas Xu et al. (2018) use OT and back-translation losses to align traditional monolingual word embeddings that do not leverage context (i.e. word type level).

## 3 Related Work

Alignment as a post-processing technique on distributional embedding spaces provides an effective solution to improve cross-lingual downstream applications. For non-contextualized embeddings, alignment based techniques for word embeddings have been thoroughly surveyed in Ruder et al. (2019). For contextualized embeddings, one direction of efforts to improve cross-lingual word representations is to use the Procrustes objective to project the monolingual embeddings from one language to the monolingual embedding space in another (Wang et al., 2019; Schuster et al., 2019). However, this generates a transformation matrix for each language pair which can be inconvenient to apply in downstream tasks. Another direction is to use explicit alignment objective at the sentence level, word level, or both which allows simultaneous learning from different languages, as opposed to rotation based approaches.

Studies that depend on sentence level alignment achieve significantly high performance on bitext sentence retrieval tasks (Artetxe and Schwenk, 2019; Zweigenbaum et al., 2017), and by design they are not applicable to word based applications. For instance, LASER (Artetxe and Schwenk, 2019) learns massively multi-lingual encoder using a huge parallel corpus whereas Feng et al. (2020) trains a bi-directional dual encoder with an additive softmax margin loss to perform translation ranking among in-batch examples. Similar to this line of work, we rely on only parallel sentences as external sources to fine-tune the model, but we define word alignment objective instead.

Other studies use word alignment objective to align parallel word pairs and fine-tune the contextualized multi-lingual LM (Cao et al., 2020; Wu and Dredze, 2020; Nagata et al., 2020). Cao et al. (2020) use regularized L2 based alignment objective to align parallel word pairs. Wu and Dredze (2020) use contrastive learning to align parallel word pairs relative to negative pairs in the batch. These approaches rely on unsupervised word aligners which are often sub-optimal to generate the parallel word pairs (e.g. FastAlign (Dyer et al., 2013b) or optimal transport (Grave et al., 2018)) and use these pairs as weak form of supervision. Our work is most similar to these methods in that we use word level alignment objective; however, we learn the aligned word pairs implicitly during optimization rather than obtaining them beforehand using external aligners and applying heuristics to keep only one-to-one mappings.

More recently, Chi et al. (2021) developed an end-to-end model that first aligns both source and target words with OT and then use the alignments as self-labels to fine-tune the contextualized LM. They use three objective functions for fine-tuning: Masked Language Modeling (MLM) (Devlin et al., 2019), Translation Language Modeling (TLM) (Lample and Conneau, 2019), and the cross entropy between predicted masked words and their corresponding alignments obtained from OT. Similar to their work, we use OT based signals to fine-tune the contextualized LM, but we instead use the average cost of OT alignments for fine-tuning. There are other studies that attempt to combine various objectives for learning cross-lingual super-

vision. For example, Dou and Neubig (2021); Hu et al. (2021b) incorporate the following objectives on cross-lingual data: MLM, TLM, sentence level alignment (e.g. parallel sentence identification objective), and word level alignment. In this paper, we do not investigate combined objective functions similar to these works. We believe that adding more objectives can further boost the performance and we leave it for future work.

## 4 Method

Figure 1 shows the overall fine-tuning process. As input, we require parallel sentences (i.e. pairs of aligned sentences in source and target languages) and contextualized multilingual LM. We use English as fixed target language and other non-English languages as source language (more details in Section 5.1). For each model iteration, we first embed words in source and target sentences independently with the pre-trained contextualized LM (Section 4.1). These representations are then used as input for OT optimization applied for each source-target language pair. We then fine-tune the contextualized LM with the accumulated regularized loss across all language pairs as a guidance (Section 4.3). We formulate the task of OT as minimizing the cost of transferring knowledge within context, from a non-English source sentence to an English target sentence in an unsupervised fashion.[1]

### 4.1 Input Representation

OT optimization is flexible to align different textual units such as words and subwords. We provide contextualized representations for words/subwords in source and target sentences as input for the OT optimization process. We use the last layer of pre-trained LMs to obtain the contextualized representations.[2] For word representations, to have a fair comparison with Cao et al. (2020), we follow their assumption that the last subword embedding for a word contains sufficient contextual information to represent the word.[3] Subwords allow for more nuanced alignments and an increase in the vocabulary coverage which can be beneficial for languages that are rich with compounds or morphemes (e.g. Arabic and German).[4]

For each source-target language pair, we pass a batch of parallel sentences, represented by contextualized words/subwords embeddings as input for the OT optimization, which in turn learns to minimize the cost of transferring from source to target distributions. This process is applied to compute independent OT optimization for each source-target language pairs independently.[5] We base our model on multilingual BERT (mBERT), that is jointly trained on 104 languages in which a shared vocabulary is constructed. Techniques discussed here are agnostic to the choice of pre-trained multilingual LMs.[6]

### 4.2 Optimal Transport Optimization

We use Sinkhorn divergence which interpolates between Wasserstein distance (i.e. Optimal Transport) and Maximum Mean Discrepancy (MMD), leveraging both OT geometrical properties and MMD efficiency in high-dimensional spaces (Ramdas et al., 2017; Feydy et al., 2019). MMD is an energy distance or kernel which adds an entropic penalty/regularization for the optimizer and is mathematically cheaper to compute (Gretton et al., 2006). We use the variant introduced in Feydy et al. (2019) which leads to entropic smoothing for the weights and more stabilized and unbiased gradients as the following:

$$
\begin{aligned}
S_\epsilon(\alpha,\beta) &= OT_\epsilon(\alpha,\beta) - \frac{1}{2}OT_\epsilon(\alpha,\alpha) - \frac{1}{2}OT_\epsilon(\beta,\beta) \\
OT_\epsilon(\alpha,\beta) &= \min_\pi \langle \pi, C\rangle + \epsilon KL(\pi, \alpha\otimes\beta) \quad (1) \\
& s.t.\ \pi >= 1,\ \pi 1 = \alpha,\ \pi^T 1 = \beta,
\end{aligned}
$$

where $\alpha$ and $\beta$ (initialized with uniform distribution) represent weights of words for each sample in the source and target distributions, respectively.[7]

[1] The method can be applied to any language pair (e.g. Bulgarian-Russian). We choose English since resources are available in abundance including parallel datasets and evaluation benchmarks for cross-lingual transfer.

[2] We also investigated other layers for word representation but empirically found the last layer to be the best.

[3] The choice of subword embedding to represent each word (e.g. first or last subword or mean pool of all subwords) can be empirical and can differ depending on the language properties. For instance, the morphological inflection in most European languages lies in the suffix while head words of compounds tend to occur on the right in Germanic languages. Hence, the last subword representation may contain more morpho-syntactic information than head word depending on the language.

[4] For example, the morpheme *h* in the Arabic word "ktbh" corresponds to it in the English segment "he wrote it".

[5] Combining different languages in one OT process increases the learning complexity - refer to Appendix E for more details.

[6] We started with mBERT to have a fair comparison with other works that fine-tune with alignment objective.

[7] We also found improvements in some languages with TF-IDF initialization; however, TF-IDF relies on computing statistics on the overall corpus which can be insufficient to compute such statistics for low resource languages.

Note that each ($\alpha$ and $\beta$) must sum to 1. We use Euclidean distance to encode $C$ as the ground cost in Equation (1). $C$ is a $n \times m$ matrix, which represents the effort or cost of moving a point in source distribution to a point or a set of points in target distribution; $n$ and $m$ are the number of words in source and target languages respectively. $\pi$ is also a $n \times m$ matrix denoting soft alignment between a word in source language to word(s) in target language (i.e. how much probability mass from a point in source distribution is assigned to a point in target distribution).

The $OT_\epsilon$ optimizer works by finding word matches between source and target sentences while minimizing the ground cost. The $OT_\epsilon$ solver is controlled by $\epsilon = 0.05$ to balance between Wasserstein distance and MMD ($KL$ term in Equation (1)).[8] To minimize this distance, we use Sinkhorn iterative matching algorithm which finds the solution of Equation (1) in terms of dual expression by iteratively updating the dual vectors between source and target until convergence (Feydy et al., 2019). We use $\pi$ as the final alignment between words in the source and target sentences.

Given that our approach works on contextualized embedding, where the individual word representation is different based on the context, applying OT to the entire training data is computationally prohibitive. Previous studies proposed the use of mini-batch strategy to apply OT on large scale datasets and proved its effectiveness as an implicit regularizer in machine learning settings (Fatras et al., 2021, 2020). We follow the mini-batch strategy to learn OT on a batch of parallel sentences for each language pair independently and use the resultant loss function to fine-tune our model as shown in Figure 1.

### 4.3 Fine-tuning with OT

'To fine-tune the pre-trained LM with OT, we first accumulate the cost of alignments obtained by $S_\epsilon(\alpha, \beta)$ in Equation (1) for each source-target language pair as discussed in Section 4.2. Similar to Cao et al. (2020), we additionally add a regularization term to the OT loss to penalize the model if the target language embeddings in the tuned model shifts far from its initialization.

$$l(c; P^k) = -S^k_\epsilon(\alpha, \beta) + \lambda \sum_{t \in P^k} \sum_{i=1}^{len(t)} \|c(j,t) - c_0(j,t)\|_2^2 , \quad (2)$$

where $\lambda$ is set to 1 and t is a target sentence in the parallel corpus $P^k$ for language k. $c(j,t)$ represents the contextualized representation for a word j in sentence t with the language model being tuned whereas $c_0(j,t)$ represents the initial representation with the un-tuned contextualized language model. We then back-propagate the resultant regularized loss (as shown in Equation (2)), summed over all $K$ languages, i.e., $L(c) = \sum_{i=1}^{K} l(c; P^i)$ to fine-tune the contextualized word representations.

## 5 Experimental Setup

### 5.1 Data Pre-processing

Following previous studies (Lample and Conneau, 2019; Cao et al., 2020), we use parallel data (approximately 32M sentence pairs) from a variety of corpora to cover different language pairs and domains as shown in Appendix A - Europarl corpora (Koehn, 2005), MultiUN (Eisele and Chen, 2010), IIT Bombay (Kunchukuttan et al., 2018), Tanzil and GlobalVoices (Tiedemann, 2012), and OpenSubtitles (Lison and Tiedemann, 2016). In all cases, we use English (en) as the target language and the tokenizer in Koehn et al. (2007). We use 250K sentences for training, upsampling from language pairs where this much data is not available. We shuffled the data to break their chronological order if any. For our main model, we consider the following five languages: Bulgarian (bg), German (de), Greek (el), Spanish (es), and French (fr), similar to Cao et al. (2020). For our larger model, we additionally used the following languages: Russian (ru), Arabic (ar), Mandarin (zh), Hindi (hi), Thai (th), Turkish (tr), Urdu (ur), Swahili (sw), and Vietnamese (vi).

### 5.2 Model Optimization

We use Adam (Kingma and Ba, 2015) for fine-tuning pre-trained LM using OT with learning rate of $5e-5$ for one epoch. We sample equal-sized parallel sentences from each language pair, do a forward pass accumulating losses for each language pair and then backpropagate based on combined loss from all language pairs. We use Geomloss

[8] We investigated few values for $\epsilon$. The default value $\epsilon = 0.05$ in Geomloss (Feydy et al., 2019) provides the best results.

for Sinkhorn divergence with its default parameter values (Feydy et al., 2019). We empirically chose batch size of 24 and gradient accumulation step of 2 to balance between speed, memory, and model accuracy.[9] Having smaller batch sizes or updating the gradients too frequently slightly hurt the performance and may lead to over-fitting the contextualized LM to noisy parallel sentences or irregular patterns.

### 5.3 Evaluation

We evaluate our proposed method for two tasks provided by XTREME benchmarks (Hu et al., 2020): XNLI for textual entailment where the task is to classify the entailment relationship between a given pair of sentences into entailment/neutral/contradiction (Conneau et al., 2018; Williams et al., 2018); XQuAD for question answering where the task is to identify the answer to a question as a span in the corresponding paragraph (Artetxe et al., 2020; Rajpurkar et al., 2016).[10] These tasks evaluate zero shot transferability and hence we train all tasks using English labelled data with cross-entropy loss and test on the target languages. More details about the task settings can be found in Appendix B. To measure the improvements, we use F1 score for textual entailment; F1 and EM (Exact Match) scores for question answering which reflect the partial and exact matches between the prediction and ground truth, respectively.

### 5.4 Models Comparison

In addition to mBERT, we compare our approach to the following baselines: 1. XLM (Lample and Conneau, 2019) which use similar objective as mBERT with a larger model and vocabulary, 2. L2 (Cao et al., 2020) which uses L2 based alignment objective, 3. AMBER (Hu et al., 2021a) for XNLI which uses a combination of MLM, TLM, word alignment and sentence alignment objectives,[11] 4. MAD-X (Pfeiffer et al., 2020b) for XQuAD which leverages language and task adapters for efficient cross lingual transfer.[12] We also compare how our model performs with respect to current state-of-the-art model i.e. XLMR (Conneau et al., 2020) which is same as XLM but trained on much more data.

| Model | XNLI F1 | XQuAD F1 | XQuAD EM |
|---|---|---|---|
| mBERT | 71.9 | 73.1 | 57.0 |
| XLM | 74.6 | 66.5 | 50.2 |
| AMBER | **76.4** | - | - |
| mBERT† | 73.5 | 73.4 | 57.8 |
| L2† | 74.6 | 68.0 | 51.6 |
| MAD-X$^{\text{mBERT}}$† | - | 70.2 | 53.8 |
| WordOT (Ours) | 75.4 | **74.7** | **59.0** |

Table 1: Averaged scores for XNLI and XQuAD benchmarks across three runs compared to baselines in seen languages. **Bold** scores are the highest in the respective columns. † refers to internal benchmarking, where we either obtained the models from the authors or implemented internally.

## 6 Results and Discussion

Table 1 shows the performance of our proposed method (WordOT) averaged for languages that are seen during OT fine-tuning. We compare that to the baselines and state-of-the-art approaches in the respective evaluation tasks (XNLI and XQuAD). We run all tasks for three seeds for each considered language and report the average scores for experiments that we run internally. More detailed results per language can be seen in Appendix C.

Compared to the baseline mBERT, we obtain +1.9% and +1.3% F1 scores on average in XNLI and XQuAD, respectively. Compared to L2 (Cao et al., 2020), we obtain an average improvement of +0.8% for XNLI and +6.7% for XQuAD in F1 scores. In XNLI, we obtain comparable F1 score (-1.0%) to the more recent model - AMBER (Hu et al., 2021a). This could be attributed to the TLM (Lample and Conneau, 2019) objective used in AMBER which provides additional cross-lingual signal and hence, further boosts the performance. In XQuAD, we obtain better F1 score (+4.5%) than the more recent work - MAD-X (Pfeiffer et al., 2020b) - showing the effectiveness of our method.

**More languages during optimization:** In our previous results, we fine-tuned mBERT with parallel sentences drawn from a set of 5 languages (refer to Section 5.1). We also investigate whether adding more languages during fine-tuning with OT

[9] Roughly, batch size = 1 takes at least 5 days to complete fine-tuning while batch size = 24 takes around 8 hours on a single NVIDIA V100 GPU.

[10] Refer to (Hu et al., 2020) for more details regarding these benchmarks. We use XTREME open source code implementation - https://github.com/google-research/xtreme

[11] We compare our model with the published AMBER variant that does not use sentence alignment as that is most comparable to our settings.

[12] We internally reproduce MAD-X scores with mBERT as the main model to show fair comparison with our method. MAD-X$^{\text{base}}$ and MAD-X$^{\text{mBERT}}$ refers to MAD-X architectures with XLMR-Base and mBERT as main model respectively.

**XNLI**

| Model | en | bg | de | el | es | fr | ar | hi | ru | sw | th | tr | ur | vi | zh | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mBERT | **82.6** | 69.3 | 72.0 | 67.7 | 75.2 | 74.4 | 66.0 | 60.8 | 69.4 | 51.0 | 55.3 | 62.9 | 58.5 | 71.0 | 69.9 | 73.5/67.1 |
| L2 | 81.0 | 73.3 | 72.8 | **70.9** | 74.9 | 74.4 | 62.4 | 59.2 | 67.3 | 42.8 | 48.5 | 54.6 | 56.3 | 69.7 | 69.5 | 74.6/65.2 |
| WordOT | 82.1 | **73.6** | **73.7** | 70.6 | **76.7** | **75.9** | 66.3 | 61.3 | 69.7 | 49.8 | 54.8 | 61.7 | 59.4 | 70.9 | 70.5 | **75.4**/67.8 |
| WordOT* | 81.5 | 73.5 | 73.3 | 70.8 | 76.3 | 75.3 | 66.0 | 61.9 | 69.7 | 48.3 | 55.7 | 61.3 | 59.7 | 71.1 | **71.2** | 75.1/67.7 |
| LargeWordOT | 81.8 | 72.6 | 73.1 | 69.8 | 76.1 | 75.3 | **66.9** | **64.8** | **70.4** | **62.2** | **60.1** | **68.6** | **60.6** | **72.2** | 70.4 | 74.8/**69.7** |

**XQuAD**

| Model | Metric | en | de | el | es | ar | hi | ru | th | tr | vi | zh | Average |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| mBERT | F1 | 83.7 | 72.0 | 62.3 | **75.6** | **61.5** | 57.3 | **70.7** | **42.2** | 54.1 | 68.6 | **59.3** | 73.4/**64.3** |
| L2 | F1 | 81.4 | 67.5 | 56.6 | 66.2 | 48.0 | 42.6 | 62.6 | 25.3 | 39.0 | 59.8 | 48.4 | 68.0/54.3 |
| WordOT | F1 | **84.2** | **73.6** | **65.6** | 75.5 | 58.6 | 55.7 | 68.6 | 42.1 | 51.8 | 69.0 | 57.3 | **74.7**/63.8 |
| WordOT* | F1 | 83.5 | 72.7 | 64.7 | 74.2 | 58.3 | 53.6 | 68.4 | 42.0 | 50.7 | 68.2 | 56.8 | 73.8/63.0 |
| LargeWordOT | F1 | 83.4 | 71.8 | 63.2 | 73.8 | 55.9 | **59.5** | 68.9 | 38.9 | **59.2** | **70.2** | 51.4 | 73.1/63.3 |
| mBERT | EM | **72.4** | 55.9 | 45.3 | **57.5** | **45.5** | 44.1 | **53.9** | 32.6 | 39.5 | **49.7** | **49.7** | 57.8/**49.6** |
| L2 | EM | 69.4 | 51.3 | 40.3 | 45.4 | 29.9 | 28.4 | 44.1 | 17.8 | 25.5 | 41.2 | 40.0 | 51.6/39.4 |
| WordOT | EM | **72.4** | **57.8** | **48.5** | 57.1 | 40.9 | 40.8 | 51.3 | 32.9 | 37.1 | 49.1 | 48.6 | **59.0**/48.8 |
| WordOT* | EM | 71.7 | 56.9 | 47.6 | 55.5 | 40.4 | 38.8 | 50.5 | **33.4** | 36.1 | 48.5 | 48.2 | 57.9/48.0 |
| LargeWordOT | EM | 71.8 | 56.4 | 46.5 | 55.6 | 37.1 | **44.9** | 50.8 | 30.1 | **44.4** | 49.3 | 43.1 | 57.6/48.2 |

Table 2: F1 score in XNLI and (F1 / EM) scores in XQuAD for each language across three runs. **Bold** scores are the highest in the respective column and metric. *Average* starts with the average score for the 5 seen languages (separated by vertical bar) in L2, *WordOT* and *WordOT** followed by the average score for the 15 languages seen during optimization in *LargeWordOT*, separated by /.

(LargeWordOT) would help improve the performance. We expanded the set of languages to all 15 languages as described in Section 5.1 (also Table 6 in Appendix A). As a result of computational complexity of OT, we instead used batch size of 8 and gradient accumulation step of 3 to overcome memory overhead. We also re-trained the model with previous 5 languages using new hyper-parameter settings (WordOT*) to have a fair comparison between both models.

Table 2 shows the results for XNLI and XQuAD, respectively. In XNLI, we obtain 2.6% improvements with LargeWordOT compared to mBERT. We do not observe improvements on average for XQuAD benchmark for LargeWordOT. This could be a byproduct of fine-tuning mBERT with parallel texts of different languages, exposing their similarities as well as their differences to the whole network. XQuAD, being a difficult task compared to XNLI is impacted more by these differences in languages' properties (language family, writing script, word order etc.). Moreover, we observe that adding more languages during fine-tuning slightly decreases the average score for the 5 languages seen as in WordOT*. Looking at scores for each language individually, we gain significant improvements for hi, sw, and tr across the two tasks. Note that the monolingual data available in Wikipedia is scarce for sw, hi, tr, and ar.

We also examine the impact of OT fine-tuning on unseen languages from the performance of WordOT*. We notice similar or better performance compared to LargeWordOT on average for all languages for both tasks, thereby showing that the performance on remaining languages on average is comparable. In addition, Table 2 shows that WordOT performs overall better than its counterpart (WordOT*) both of which differ in the batch size (24 vs. 8) and the number of gradient accumulation steps (2 vs. 3). Hence, we presumably would obtain better scores with higher batch size for OT if the implementation is optimized for memory efficiency.

**Notes on OT Efficiency:** To examine the efficiency of our proposed method, we computed the time taken by one epoch of fine-tuning mBERT with five language pairs (250K parallel sentences for each pair). On a single NVIDIA V100 GPU, it took approximately 8 hours to complete one epoch, which is relatively 30% higher compared to L2 based alignment method which took approximately 6 hours with the same settings. This increase is expected as our method considers every combination of words from source to target in order to find OT mapping with minimum cost for each step of fine-tuning. Hence, it performs at least $O(n * m)$ operations, where $n$ and $m$ are the number of words in source and target languages, respectively. On the other hand, L2 based alignment considers only precomputed one-to-one mapping which speeds up the process. This is a trade-off between time and accuracy where OT outperforms L2 in both tasks for seen and unseen languages in terms of accuracy. The time complexity only impacts the model

during fine-tuning which is done once.

**Word vs. subword alignments:** As discussed in Section 4.1, OT is flexible to align different textual units. We compare between fine-tuning at word level (WordOT) and subword level (SubOT). Table 3 shows that SubOT slightly improves the scores in XNLI and slightly decreases the scores in XQuAD. We observe individual improvements for some languages with subword level alignment. In XNLI, the F1 scores for Greek and German slightly increased by 0.65% and 0.47% respectively with subword information. Both languages exhibit compounding structure as opposed to remaining languages seen during training in which the benefit is less observed (<0.29%). For XQuAD, we observe slight drop in overall performance with subword information.[13] This can be attributed to the nature of XQuAD task in which a span of information is identified. We believe that the difference between word and subword can be more pronounced when we construct language specific vocabulary and/or increase the vocabulary capacity.

| **Model** | **XNLI** | | **XQuAD** | |
|---|---|---|---|---|
| | seen | all | seen | all |
| WordOT | 75.4 | 67.8 | **74.7/59.0** | **63.8/48.8** |
| SubOT | **75.7** | **67.9** | 74.0/58.3 | 63.5/48.5 |

Table 3: Scores (F1 for XNLI F1 / EM for XQuAD) for SubOT vs. WordOT. *"All"* represents the average of both seen and unseen languages during optimization.

**Impact of amount of parallel data:** In all previous experiments, we used 250k parallel sentences (upsampled if needed). Adding more language pairs during training with OT increases the fine-tuning time thus limiting the scalability of our proposed approach. In addition, the impact of OT if we have limited amount of parallel data for a low resource language is not clear.[14] To address the aforementioned two points, we investigate the impact of reducing the amount of available parallel data. These experiments were performed using Large-WordOT. We can see from Table 4 that for XNLI, we can achieve comparable performance (-0.4% absolute) with as low as 50k sentences, i.e. one-fifth of the data. Similar experiments for XQuAD can be found in the Appendix D. This shows that alignment using OT is robust to low data scenarios, especially for languages where huge amounts of parallel data might not be available.

| | **XNLI** | | | | | | |
|---|---|---|---|---|---|---|---|
| Model | en | bg | de | el | es | fr | Avg |
| mBERT | **82.6** | 69.3 | 72.0 | 67.7 | 75.2 | 74.4 | 73.5 |
| 1k | 82.3 | 69.9 | 72.5 | 67.3 | 75.0 | 74.6 | 72.7 |
| 10k | 81.7 | 71.9 | 72.2 | 68.5 | 75.5 | 74.6 | 74.1 |
| 50k | 81.4 | **72.7** | 72.7 | 69.2 | 75.8 | 74.7 | 74.4 |
| 250k | 81.8 | 72.6 | **73.1** | **69.8** | **76.1** | **75.3** | **74.8** |

Table 4: XNLI F1 scores for different amounts of parallel data. mBERT represents the case where we have no parallel datasets

**State-of-the-art Comparison:** We compare our method to the state-of-the-art model XLMR which has a larger capacity in terms of model and/or training data sizes. Due to efficiency reasons, we apply OT on XLMR$^{base}$ which has similar model size compared to mBERT but is trained on significantly larger amount of data (2.5TB) and larger vocabulary.[15] As shown in Table 5, we observe comparable or slightly lower results when we apply OT on XLMR$^{base}$. Hence, explicit alignment objective with OT as our proposed method did not help further boost the performance; this is in line with the findings of Wu and Dredze (2020) which show improvements for different alignment objectives over mBERT but not XLMR.

We speculate that the robustness of XLMR over alignment objectives can be attributed to the large amount of data used for pre-training even for low resource languages. Hence, to further boost the performance, there must be consideration for the amount of data used for alignment in correlation with the pretrained data (e.g. mBERT shows benefits from our method with even smaller size of data, i.e. 50K samples). In addition, the definition of alignment objective is a determining factor. For example, Chi et al. (2021) showed improvements when they used OT based alignment as self-labels to minimize the loss between predicted masked word and the corresponding aligned word. Note that Chi et al. (2021) also uses large amount of data for training.

## 7 Qualitative Analysis for OT

Our objective is not to obtain explicit word alignment but rather compute the cost of transferring both distributions to each other and use this cost to guide the fine-tuning process. We examine the

[13] We observe benefits for some low resource languages such as th which improved +2.4% F1 and +1.6% EM.

[14] Low resource languages here refer to languages covered by mBERT vocabulary but has limited amount of parallel data when using our approach.

[15] OT can also be applied in XLMR$^{large}$; however, this would require parameter tuning to overcome memory issues.

| Model | XNLI F1 | XQuAD F1 | XQuAD EM |
|---|---|---|---|
| XLMR | **84.1** | **82.2** | **66.0** |
| XLMR$^{base}$ | 77.5 | 77.0 | 61.4 |
| WordOT$^{base}$ | 77.6 | 76.4 | 60.8 |

Table 5: Comparison with state-of-the-art (XLMR). In WordOT$^{base}$, we apply OT based fine-tuning on XLMR$^{base}$. **Bold** scores are the highest in the respective column. All results were obtained internally and are averaged across three runs. For learning rate, we use $5e-6$ for XLMR evaluation benchmarks.

obtained alignments during fine-tuning for two language pairs (German-English and Arabic-English) to inspect potential errors. We found that alignments are capable to include word relationships other than one-to-one mapping. For instance, the German compound nouns *"Vorsichtsprinzip"* and *"Rahmengesetzgebung"* are correctly aligned to *"precautionary approach"* and *"framework legislation"*. In addition, alignments do not necessarily include semantics but also highlight similar or dependent words in context, thus capturing contextual alignments. For instance, in the Arabic phrase التدخل العسكري, the first word التدخل is aligned with its literal translation *"intervention"* while العسكري is aligned with the phrase *"armed intervention"*, where *"arms"* is the literal translation while *"intervention"* is the dependent word. More examples in Tables 9 and 10 in Appendix F.

OT as an unsupervised aligner generates incorrect alignments for some cases which could be related to quality of parallel sentences or limitations of the OT variant that we used. Some parallel sentences are not translations of each other (refer to Table 11 in Appendix F) which has a negative impact on OT especially given that we use uniform distribution which leads to finding at least one target word for each word in the source sentence. For the OT limitations, the alignments happen at the point level regardless of the word order or syntactic structure of the sentence. This indicates that a word in the source language may be aligned with more than one occurrence of the same word. For instance, the Arabic word ساعة is mapped to the two occurrences of *"hours"* in the target neglecting the clause structure. This also led the model to align different morphological variants to the same instance. For example, the Arabic word تيسير is aligned with both *"facilitate"* and *"Facilitating"* in the corresponding English sentence.

## 8 Conclusion

In this paper, we investigated OT to align the space of contextualized embeddings of a source and a target sentence in order to improve contextualized word embeddings for cross-lingual settings. We trained an independent OT per language pair and used the resultant cost as a guidance to fine-tune the contextualized LM and encourage the alignment of the corresponding contextual embeddings. We obtain improvements in sentence level evaluation tasks: XNLI and XQuAD. As an improvement for our proposed technique, we intend to use different variants of OT such as Goromov-Wasserstein which performs the same logic presented in this paper in addition to its ability to align embeddings of different spaces, mapping both geometry and points of different embedding spaces. We would also like to combine more cross-lingual objectives using additional signals and perform evaluation on more tasks and languages.

## Acknowledgements

We would like to thank the Lex Science team at Amazon Web Services AI for the helpful discussions; and the reviewers for providing insightful feedback. We would also like to express gratitude to Steven Cao for sharing his implementation and model in order to help us build upon his work; and clarifying our queries.

## References

Mokhtar Z. Alaya, M. Bérar, G. Gasso, and A. Rakotomamonjy. 2019. Screening sinkhorn algorithm for regularized optimal transport. In *NeurIPS*.

Hanan Aldarmaki and Mona T. Diab. 2019. Context-aware cross-lingual mapping. In *NAACL*.

Hanan Aldarmaki, Mahesh Mohan, and Mona Diab. 2018. Unsupervised word mapping using structural similarities in monolingual embeddings. *Transactions of the Association for Computational Linguistics*, 6:185–196.

David Alvarez-Melis and T. Jaakkola. 2018. Gromov-wasserstein alignment of word embedding spaces. In *EMNLP*.

David Alvarez-Melis, S. Jegelka, and T. Jaakkola. 2019. Towards optimal transport with global invariances. In *AISTATS*.

David Alvarez-Melis, Youssef Mroueh, and T. Jaakkola. 2020. Unsupervised hierarchy matching with optimal transport over hyperbolic spaces. In *AISTATS*.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *ACL*.

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.

Gary B'ecigneul, Octavian-Eugen Ganea, Benson Chen, R. Barzilay, and T. Jaakkola. 2020. Optimal transport graph neural networks. *ArXiv*, abs/2006.04804.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. In *International Conference on Learning Representations*.

Liqun Chen, Yizhe Zhang, Ruiyi Zhang, Chenyang Tao, Zhe Gan, Haichao Zhang, Bai Li, Dinghan Shen, Changyou Chen, and Lawrence Carin. 2019. Improving sequence-to-sequence learning via optimal transport. In *International Conference on Learning Representations*.

Zewen Chi, Li Dong, Bo Zheng, Shaohan Huang, Xian-Ling Mao, Heyan Huang, and Furu Wei. 2021. Improving pretrained cross-lingual language models via self-labeled word alignment. *arXiv preprint arXiv:2106.06381*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, E. Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *ACL*.

Alexis Conneau, Guillaume Lample, Ruty Rinott, Adina Williams, Samuel R. Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. Xnli: Evaluating cross-lingual sentence representations. In *EMNLP*.

Marco Cuturi. 2013. Sinkhorn distances: Lightspeed computation of optimal transportation distances.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Zi-Yi Dou and Graham Neubig. 2021. Word alignment by fine-tuning embeddings on parallel corpora. In *EACL*.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013a. 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.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013b. A simple, fast, and effective reparameterization of ibm model 2. In *HLT-NAACL*.

Andreas Eisele and Yu Chen. 2010. MultiUN: A multilingual corpus from united nation documents. In *Proceedings of the Seventh International Conference on Language Resources and Evaluation (LREC'10)*, Valletta, Malta. European Language Resources Association (ELRA).

Kilian Fatras, Younes Zine, Rémi Flamary, Rémi Gribonval, and Nicolas Courty. 2020. Learning with minibatch wasserstein : asymptotic and gradient properties.

Kilian Fatras, Younes Zine, Szymon Majewski, Rémi Flamary, Rémi Gribonval, and Nicolas Courty. 2021. Minibatch optimal transport distances; analysis and applications.

Fangxiaoyu Feng, Yin-Fei Yang, Daniel Matthew Cer, N. Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *ArXiv*, abs/2007.01852.

Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouve, and Gabriel Peyré. 2019. Interpolating between optimal transport and mmd using sinkhorn divergences. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2681–2690.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z. Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, Léo Gautheron, Nathalie T.H. Gayraud, Hicham Janati, Alain Rakotomamonjy, Ievgen Redko, Antoine Rolet, Antony Schutz, Vivien Seguy, Danica J. Sutherland, Romain Tavenard, Alexander Tong, and Titouan Vayer. 2021. Pot: Python optimal transport. *Journal of Machine Learning Research*, 22(78):1–8.

Edouard Grave, Armand Joulin, and Quentin Berthet. 2018. Unsupervised alignment of embeddings with wasserstein procrustes.

A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and Alex Smola. 2006. A kernel method for the two-sample-problem. In *NIPS*.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *ACL*.

Junjie Hu, M. Johnson, Orhan Firat, Aditya Siddhant, and Graham Neubig. 2021a. Explicit alignment objectives for multilingual bidirectional encoders. In *NAACL*.

Junjie Hu, M. Johnson, Orhan Firat, Aditya Siddhant, and Graham Neubig. 2021b. Explicit alignment objectives for multilingual bidirectional encoders. In *NAACL*.

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 *ICML*, pages 4411–4421.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*, volume 5, pages 79–86. Citeseer.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic. Association for Computational Linguistics.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The IIT Bombay English-Hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France. PMLR.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. In *NeurIPS*.

Pierre Lison and Jörg Tiedemann. 2016. OpenSubtitles2016: Extracting large parallel corpora from movie and TV subtitles. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 923–929, Portorož, Slovenia. European Language Resources Association (ELRA).

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013. Exploiting similarities among languages for machine translation. *ArXiv*, abs/1309.4168.

M. Nagata, Katsuki Chousa, and Masaaki Nishino. 2020. A supervised word alignment method based on cross-language span prediction using multilingual bert. In *EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*.

G. Peyré and Marco Cuturi. 2019. Computational optimal transport. *Found. Trends Mach. Learn.*, 11:355–607.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020a. Adapterhub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 46–54.

Jonas Pfeiffer, Ivan Vuli'c, Iryna Gurevych, and Sebastian Ruder. 2020b. Mad-x: An adapter-based framework for multi-task cross-lingual transfer. In *EMNLP*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *EMNLP*.

Aaditya Ramdas, Nicolás García Trillos, and Marco Cuturi. 2017. On wasserstein two-sample testing and related families of nonparametric tests. *Entropy*, 19:47.

Sebastian Ruder, Ivan Vulic, and Anders Søgaard. 2019. A survey of cross-lingual word embedding models. *J. Artif. Intell. Res.*, 65:569–631.

Filippo Santambrogio. 2015. Optimal transport for applied mathematicians. *Birkäuser, NY*, 55(58-63):94.

Morgan A. Schmitz, Matthieu Heitz, Nicolas Bonneel, Fred Ngolè, David Coeurjolly, Marco Cuturi, Gabriel Peyré, and Jean-Luc Starck. 2018. Wasserstein dictionary learning: Optimal transport-based unsupervised nonlinear dictionary learning. *SIAM Journal on Imaging Sciences*, 11(1):643–678.

Bernhard Schmitzer. 2019. Stabilized sparse scaling algorithms for entropy regularized transport problems.

Tal Schuster, Ori Ram, R. Barzilay, and A. Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *NAACL-HLT*.

Sidak Pal Singh, Andreas Hug, Aymeric Dieuleveut, and Martin Jaggi. 2019. Context mover's distance & barycenters: Optimal transport of contexts for building representations. In *DGS@ICLR*.

Samuel L. Smith, David H. P. Turban, Steven Hamblin, and Nils Y. Hammerla. 2017. Offline bilingual word vectors, orthogonal transformations and the inverted softmax.

Jörg Tiedemann. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey. European Language Resources Association (ELRA).

Titouan Vayer, N. Courty, R. Tavenard, L. Chapel, and Rémi Flamary. 2019. Optimal transport for structured data with application on graphs. In *ICML*.

Yuxuan Wang, Wanxiang Che, Jiang Guo, Yijia Liu, and Ting Liu. 2019. Cross-lingual bert transformation for zero-shot dependency parsing.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*.

Shijie Wu and Mark Dredze. 2020. Do explicit alignments robustly improve multilingual encoders?

L. Xu, Han Sun, and Y. Liu. 2019. Learning with batch-wise optimal transport loss for 3d shape recognition. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3328–3337.

Ruochen Xu, Yiming Yang, Naoki Otani, and Yuexin Wu. 2018. Unsupervised cross-lingual transfer of word embedding spaces. In *EMNLP*.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Earth mover's distance minimization for unsupervised bilingual lexicon induction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1934–1945, Copenhagen, Denmark. Association for Computational Linguistics.

Pierre Zweigenbaum, S. Sharoff, and R. Rapp. 2017. Overview of the second bucc shared task: Spotting parallel sentences in comparable corpora. In *BUCC@ACL*.

## A Parallel Corpus Details

| Data Source | Lang | #Pairs | Data Source | Lang | #Pairs |
|---|---|---|---|---|---|
| Europarl corpora | bg-en | 371K | IIT Bombay | hi-en | 618K |
| | de-en | 1M | OpenSubtitles | th-en | 278K |
| | el-en | 157K | | tr-en | 9M |
| | es-en | 990K | | vi-en | 608K |
| | fr-en | 1M | Tanzil | ur-en | 422K |
| MultiUN | ar-en | 2M | Tanzil | sw-en | 128K |
| | ru-en | 9M | Global-Voices | | |
| | zh-en | 3M | | | |

Table 6: The data source and number of parallel sentences in each pair of languages. Overall 32M parallel sentences combined

## B Task Hyperparameter Settings

We benchmarked the performance of our model and baselines with XNLI and XQuAD datasets using the same settings as XTREME (Hu et al., 2020). However, for internally implemented MAD-X using XLMR-base or mBERT as the base model, we followed the XQuAD scripts as in (Pfeiffer et al., 2020a) because of incompatibility in versions of certain packages between XTREME[16] and Adapters[17] libraries. We used learning rate of 1e-4 for adapters and trained on XQuAD task for 4 epochs with a batch size of 4 and gradient accumulation steps of 4. Rest of the settings were similar to as mentioned in Pfeiffer et al. (2020b), i.e., adapter sizes correspond to reductions of 2 for language adapters, 2 for invertible adapters, and 16 for task adapters.

## C Detailed Results Per Language

Table 7 shows comparison of our method with baselines and state-of-the-art approaches per language (average numbers across 3 runs).

## D Impact of Amount of Parallel Data for XQuAD

Table 8 shows the impact of amount of parallel sentence pairs used during fine-tuning with OT for XQuAD benchmark. From the XQuAD results, we don't see a clear trend of decreasing performance with the decrease in parallel data used for OT fine-tuning. Results are more or less comparable to the baseline, with surprisingly best performance being seen with only 1k parallel sentence pairs. This

[16] https://github.com/google-research/xtreme

[17] https://github.com/Adapter-Hub/adapter-transformers

| XNLI | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Model** | **en** | **bg** | **de** | **el** | **es** | **fr** | **Avg** |
| mBERT | 80.8 | 68.0 | 70.0 | 65.3 | 73.5 | 73.4 | 71.9 |
| XLM | 82.8 | 71.9 | 72.7 | 70.4 | 75.5 | 74.3 | 74.6 |
| XLMR | **88.7** | **83.0** | **82.5** | **80.8** | **83.7** | **82.2** | **81.6** |
| AMBER | 84.1 | 73.9 | 74.7 | 71.6 | 76.6 | 77.7 | 76.4 |
| mBERT† | 82.6 | 69.3 | 72.0 | 67.7 | 75.2 | 74.4 | 73.5 |
| L2† | 81.0 | 73.3 | 72.8 | 70.9 | 74.9 | 74.3 | 74.6 |
| **Ours** | | | | | | | |
| WordOT | 82.1 | 73.6 | 73.7 | 70.6 | 76.7 | 75.9 | 75.4 |

| XQuAD | | | | | |
|---|---|---|---|---|---|
| **Model** | **en** | **de** | **el** | **es** | **Avg** |
| mBERT | 83.5/72.2 | 70.6/54.0 | 62.6/44.9 | 75.5/56.9 | 73.1/57.0 |
| XLM | 74.2/62.1 | 66.0/49.7 | 57.5/39.1 | 68.2/49.8 | 66.5/50.2 |
| XLMR | **86.5/75.7** | **80.4/63.4** | **79.8/61.7** | **82.0/63.9** | **82.2/66.2** |
| MAD-X$^{\text{base}}$ | 83.5/72.6 | 72.9/56 | 72.9/54.6 | 75.9/56.9 | 76.3/60.0 |
| mBERT† | 83.7/72.4 | 72.0/55.9 | 62.3/45.3 | 75.6/57.5 | 73.4/57.8 |
| L2† | 81.4/69.4 | 67.5/51.3 | 56.6/40.3 | 66.2/45.4 | 68.0/51.6 |
| MAD-X$^{\text{base}}$† | 82.0/71.1 | 72.1/54.5 | 71.7/53.7 | 74.3/55.7 | 75.0/58.8 |
| MAD-X$^{\text{mBERT}}$† | 81.7/69.7 | 68.6/52.1 | 58.6/41.3 | 71.8/51.9 | 70.2/53.8 |
| **Ours** | | | | | |
| WordOT | 84.2/72.4 | 73.6/57.8 | 65.6/48.5 | 75.5/57.1 | 74.7/59.0 |

Table 7: Averaged F1 and F1/EM scores for XNLI and XQuAD benchmarks across three runs in seen languages. **Bold** scores are the highest in the respective columns. † refers to internal benchmarking, where we either obtained the models from the authors or implemented internally.

could be attributed to the fact that these experiments were run using LargeWordOT that utilized 15 languages and with additional parallel data or more fine-tuning, XQuAD is being impacted negatively by the differences in these 15 languages.

| XQuAD | | | | | |
|---|---|---|---|---|---|
| Model | en | de | el | es | Avg |
| mBERT | 83.7/72.4 | 72.0/55.9 | 62.3/45.3 | 75.6/57.5 | 73.4/57.8 |
| 1k | 84.4/73.3 | 72.7/56.4 | 63.9/46.8 | 75.2/56.6 | 74.1/58.3 |
| 10k | 84.1/73.1 | 72.2/56.6 | 61.9/44.6 | 75.4/57.0 | 73.4/57.8 |
| 50k | 84.0/72.2 | 72.2/57.1 | 63.3/45.4 | 74.7/56.3 | 73.6/57.8 |
| 250k | 83.4/71.8 | 71.9/56.5 | 63.2/46.5 | 73.8/55.6 | 73.1/57.6 |

Table 8: XQuAD (F1/EM) scores for different amounts of parallel data. Experiments were run with LargeWordOT. mBERT represents the case where we have no parallel datasets

## E Shuffling different languages in one OT process

In all our experiments, we trained an independent OT per language pair. We additionally examined the impact of combining more than one non-English language in the same OT optimization versus learning independent OT per language. Hence, in each batch, we have pairs of sentences (non-English to their equivalents in English) drawn equally from all languages seen during training; remaining parameters are the same hence we back-propagate the loss values with the same number of computations. Combining sentences from different languages in one OT optimization leads to soft aligning all seen languages at once minimizing the cost of transferring knowledge from source to target. We observe consistent significant drop across languages in XNLI. The performance dropped for approximately 5.1% for de, 3.8% for es, 5.1% for fr, and 9.1% for bg. As we conflate sentences from different languages, the OT alignment optimization becomes harder especially that we follow batching strategy and languages can differ at different linguistic properties (e.g. syntactic structure ... etc).

## F Examples

| |
|---|
| وأود في هذا الصدد أن أتناول بوجه خاص قضية مدينة القدس ، وهي قضية أساسية بالنسبة لجميع أعضاء المجموعة العربية |
| In this regard , I wish to address in specific the issue of the City of Jerusalem , a central issue for all of the Members of the Arab Group |
| وأود — I wish |
| في — In,in |
| هذا — this |
| الصدد — regard |
| أن — to |
| أتناول — address |
| بوجه — **in the** |
| خاص — specific |
| قضية — **the,issue,of** |
| مدينة — **the** |
| القدس — City of Jerusalem |
| قضية — issue |
| أساسية — Central |
| بالنسبة — for,**of** |
| لجميع — all, **the** |
| أعضاء — **of**, Members,**the** |
| المجموعة — **Members**,the,**Arab**,Group |
| العربية — Arab,**Group** |
| وبذلك اختتمت اللجنة مناقشتها العامة لهذا البند من جدول الأعمال . |
| The Committee thus concluded its general discussion on this agenda item . |
| وبذلك — thus |
| اختتمت — concluded |
| اللجنة — Committee |
| مناقشتها — its,**general** |
| العامة — **discussion** |
| لهذا — this |
| البند — agenda,item |
| من — on |
| جدول — **The,this** |
| الأعمال — **discussion**,agenda |
| أربعة أعضاء من دول أوروبا الغربية ودول أخرى . |
| Four members from Western European and other States . |
| أربعة — Four |
| أعضاء — members |
| من — from |
| دول — **Western** |
| أوروبا — European |
| الغربية — **European,States** |
| ودول — and,States |
| أخرى — other |

Table 9: Alignment examples in Arabic. Words in **bold** are either errors or not direct alignment.

| Zunächst wurde die für die Beitreibung der traditionellen Eigenmittel bzw. Zölle und Agrarausgleichsbeträge zu erhebende Prämie auf 25 Prozent erhöht<br>First , the premium paid for the collection of traditional own resources , i.e. customs duty and agricultural levies , was increased to 25 % | |
|---|---|
| Zunächst | First |
| wurde | , was |
| die | the |
| für | for |
| die | the |
| Beitreibung | premium,collection |
| der | of |
| traditionellen | collection,traditional,own,agricultural |
| Eigenmittel | resources |
| bzw. | i.e. |
| Zölle | customs,duty,agricultural,levies |
| und | and |
| Agrarausgleichsbeträge | duty,levies |
| zu | paid |
| erhebende | paid |
| Prämie | premium |
| auf | to |
| 25 | 25 |
| Prozent | % |
| erhöht | increased |
| Mit anderen Worten : Die tschechische Rahmengesetzgebung in diesem Bereich muß an die der Europäischen Union angepaßt und auch praktisch umgesetzt werden<br>In other words , the Czech framework legislation in this area must be adapted and to all all intents and purposes converted to that of the European Union | |
| Mit | In |
| anderen | other |
| Worten | words |
| : | , |
| Die | the |
| tschechische | Czech |
| Rahmengesetzgebung | framework,legislation |
| in | in |
| diesem | this |
| Bereich | area |
| muß | must |
| an | to,to,of |
| die | that,of |
| der | the |
| Europäischen | European |
| Union | Union |
| angepaßt | adapted,converted |
| und | and,and |
| auch | **all,intents,purposes** |
| praktisch | all,purposes |
| umgesetzt | adapted |
| werden | be |
| Mit der Einführung des Euro ist das Wechselkursrisiko verschwunden<br>The exchange rate risk has disappeared with the advent of the euro | |
| Mit | with |
| der | the |
| Einführung | advent,of |
| des | the |
| Euro | euro |
| ist | has |
| das | The |
| Wechselkursrisiko | exchange,rate,risk |
| verschwunden | disappeared |

Table 10: Alignment examples in German. Words in **bold** are either errors or not direct alignment.

| |
|---|
| ar: الاتفاقية الإطارية |
| en: *“FCCC / SBSTA / 2002 / L.23 / Add.1 Article 6 of the Convention”* |
| de: *“Herr Präsident !”* |
| en: *“Mr President , we are dealing here with sectors which have been excluded for a long time .”* |
| de: *“Im übrigen wurden die Abhängigkeitsverhältnisse eher verstärkt , als daß die Schuldenprobleme wirklich geklärt worden wären”* |
| en: *“An analysis of the situation would seem to be more of a diagnosis as the details available and the same explanatory statement leave several signs of this imbalance the world is suffering”* |

Table 11: Examples of incorrect pairs in parallel corpus.