Massively Multilingual Document Alignment with Cross-lingual Sentence-Mover's Distance

Ahmed El-Kishky
Facebook AI
ahelk@fb.com

Francisco Guzmán
Facebook AI
fguzman@fb.com

Abstract

Document alignment aims to identify pairs of documents in two distinct languages that are of comparable content or translations of each other. Such aligned data can be used for a variety of NLP tasks from training cross-lingual representations to mining parallel data for machine translation. In this paper we develop an unsupervised scoring function that leverages cross-lingual sentence embeddings to compute the semantic distance between documents in different languages. These semantic distances are then used to guide a document alignment algorithm to properly pair cross-lingual web documents across a variety of low, mid, and high-resource language pairs. Recognizing that our proposed scoring function and other state of the art methods are computationally intractable for long web documents, we utilize a more tractable greedy algorithm that performs comparably. We experimentally demonstrate that our distance metric performs better alignment than current baselines outperforming them by 7% on high-resource language pairs, 15% on mid-resource language pairs, and 22% on low-resource language pairs.

1 Introduction

While the Web provides a large amount of monolingual text, cross-lingual parallel data is more difficult to obtain. Despite its scarcity, parallel cross-lingual data plays a crucial role in a variety of tasks in natural language processing such as machine translation. Previous works have shown that training on sentences extracted from parallel or comparable documents mined from the Web can improve machine translation models (Munteanu and Marcu, 2005) or learning word-level translation lexicons (Fung and Yee, 1998; Rapp, 1999). Other tasks that leverage these parallel texts include cross-lingual information retrieval, document classification, and multilingual representations such as XLM (Lample and Conneau, 2019). Document alignment is a method for obtaining cross-lingual parallel data that seeks to pair documents in different languages such that pairs are translations or near translations of each other. As seen in Figure 1, this involves a one-to-one pairing of documents in a source language with documents in a target language.

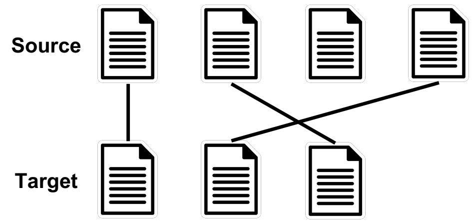


Figure 1: Documents in a source and target langauge in the same web-domain. Solid lines indicate cross-lingual document pairs.

To automate and scale the process of identifying these documents pairs, we introduce an approach to accurately mine comparable web documents across a variety of low, mid, and high-resource language directions. Previous approaches have been applied to homogeneous corpora, however mining the Web involves analyzing a variety of heterogeneous data sources (Koehn et al., 2002). Other approaches rely on corpus-specific features such as metadata and publication date which can be inconsistent and unreliable (Munteanu and Marcu, 2005; AbduI-Rauf and Schwenk, 2009). Related methods utilize document structure when calculating document similarity (Resnik and Smith, 2003; Chen and Nie, 2000). However, when mining large, unstructured collections of web documents these features are often missing or unreliable. As such, we introduce an approach that aligns documents based solely on semantic distances between their textual content.

For our approach, we first decompose documents into sentences, and encode each sentence into a cross-lingual semantic space yielding a bag-

Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, pages 616–625

of-sentences representation. Utilizing the dense, cross-lingual representation of sentences, we then compute document distances using a variant of earth mover's distance where probability mass is moved from the source document to the target document. We then leverage these document distances as a guiding metric for identifying cross-lingual document pairs and demonstrate experimentally that our proposed method outperforms state-of-the-art baselines that utilize cross-lingual document representations.

2 Related Works

Crawling and mining the web for parallel data has been previously explored by Resnik (1999) where the focus is on identifying parallel text from multilingual data obtained from a single source. For example, parallel corpora were curated from the United Nations General Assembly Resolutions (Rafalovitch et al., 2009; Ziemski et al., 2016) and from the European Parliament (Koehn, 2005). However, curating from homogeneous sources by deriving domain-specific rules does not generalize to arbitrary web-domains.

Other approaches rely on metadata for mining parallel documents in unstructured web corpora. Some methods leveraged publication date and other temporal heuristics to identifying parallel documents (Munteanu and Marcu, 2005, 2006; Udupa et al., 2009; Do et al., 2009; AbdulI-Rauf and Schwenk, 2009). However, temporal features are often sparse, noisy, and unreliable. Another class of alignment methods rely on document structure (Resnik and Smith, 2003; Chen and Nie, 2000) yet these structure signals can be sparse and may not generalize to new domains.

In the WMT-2016 bilingual document alignment shared task (Buck and Koehn, 2016a), many techniques were proposed to retrieve, score, and align cross-lingual document pairs. However this shared task only considered English to French – a high-resource direction and the proposed techniques were not readily extendable to more languages.

Several approaches translate the target corpus into the source language, then apply retrieval and matching approaches on translated 2-grams and 5-grams to query, retrieve, and align documents (Dara and Lin, 2016; Gomes and Lopes, 2016). These methods rely on high-quality translation systems to translate, however such models may not exist, especially for low-resource language directions. Additionally, these methods leverage rare n-grams to identify likely candidates, yet low-frequency words and phrases that are likely to be mistranslated by machine translation systems.

In the shared task, many document similarity measures were investigated for use in aligning English to French web documents. One method utilized a phrase table from a phrase-based statistical machine translation system to compute coverage scores, based on the ratio of phrase pairs covered by a document pair (Gomes and Lopes, 2016). Other methods utilize the translated content of the target (French) document, and find the source (English) corresponding document based on n-gram matches in conjunction with a heuristic document length ratio (Dara and Lin, 2016; Shchukin et al., 2016). Other methods translate the target documents into the source language and apply cosine similarity between tf/idf weighted vectors on unigrams and n-grams (Buck and Koehn, 2016b; Medveď et al., 2016; Jakubina and Langlais, 2016). Finally, several methods were introduced that score pairs using metadata in each document such as links to documents, URLs, digits, and HTML structure (Esplà-Gomis et al., 2016; Papavassiliou et al., 2016).

Recently, the use of neural embedding methods has been explored for bilingual alignment of text at the sentence and document level. One method proposes using hierarchical document embeddings, constructed from sentence embeddings, for bilingual document alignment (Guo et al., 2019). Another method leverages a multilingual sentence encoder to embed individual sentences from each document, then performs a simple vector average across all sentence embeddings to form a dense document representation with cosine similarity guiding document alignment (El-Kishky et al., 2019).

Word mover's distance (WMD) is an adaptation of earth mover's distance (EMD) (Rubner et al., 1998) that has been recently used for document similarity and classification (Kusner et al., 2015; Huang et al., 2016; Atasu et al., 2017). Other methods have leveraged the distance for cross-lingual document retrieval (Balikas et al., 2018). However these methods treat individual words as the base semantic unit for comparison which are intractable for large web-document alignment.

Finally, sentence mover's similarity has been proposed for automatically evaluating machine-generated texts outperforming ROUGE (Clark et al., 2019). This method is purely monolingual and sentence representations are constructed by summing individual word embeddings.

3 Problem Definition

Given a set of source documents, D_s and a set of target documents D_t, there exist $|D_s| \times |D_t|$ potential pairs of documents of the form (d_s, d_t). Let $\mathcal{P}$ be the set of all candidate pairs ($D_s \times D_t$). Then cross-lingual document alignment aims to find the largest mapping from source documents to target documents, $\mathcal{P}' \subset \mathcal{P}$, s.t. given an D_s and D_t where, without a loss of generality, $|D_s| \leq |D_t|$, the largest *injective function mapping* between D_s and D_t:

$$\forall a, b \in D_s, (a, c) \in \mathcal{P}' \land (b, c) \in \mathcal{P}' \implies a = b$$

In other words, each source document and target document can only be used in at most a single pair. This can be seen in Figure 1 where within the same web-domain, given source and target documents, the task is to match each source document to a unique target document where possible.

To find the best possible mapping between D_s and D_t we require two components: 1) a similarity function $\phi(d_s, d_t)$ which is used to score a set of candidate document pairs according to their semantic relatedness; and 2) an alignment or matching algorithm which uses the scores for each of the pairs in $D_s \times D_t$ to produce an alignment of size $min(|D_s|, |D_t|)$ representing the best mapping according to $\phi(d_s, d_t)$.

4 Cross-Lingual Sentence Mover's Distance

WMD fails to generalize to our use case for two reasons: (1) it relies on monolingual word representations which fail to capture the semantic distances between different language documents (2) intractability due to long web documents or lack word boundaries in certain languages.

To address this, we introduce cross-lingual sentence mover's distance (SMD) and show that representing each document as a bag-of-sentences (BOS) and leveraging recent improvements in multilingual sentence representations, SMD can better identify cross-lingual document pairs.

4.1 Cross-Lingual Sentence Mover's Distance

Our proposed SMD solves the same optimization problem as WMD, but utilizes cross-lingual sentence embeddings instead of word embeddings as the base semantic. In particular, we utilize LASER sentence representations (Artetxe and Schwenk, 2019). LASER learns to simultaneously embed 93 languages covering 23 different alphabets into a joint embedding space by training a sequence-to-sequence system on many language pairs at once using a shared encoder and a shared byte-pair encoding (BPE) vocabulary for all languages. Utilizing LASER, each sentence is encoded using an LSTM encoder into a fixed-length dense representation.

We adapt EMD to measure the distance between two documents by comparing the distributions of sentences within each document. More specifically, SMD represents each document as a *normalized bag-of-sentences* (nBOS) where each sentence has associated with it some probability mass. As distances can be computed between dense sentence embeddings, the overall document distance can then be computed by examining how close the distribution of sentences in the source document is to sentences in the target document. We formulate this distance as the minimum cost of transforming one document into the other.

For our basic formulation of SMD, each document is represented by the relative frequencies of sentences, i.e., for the i_{th} sentence in the document,

$$d_{A,i} = cnt(i)/|A| \tag{1}$$

where $|A|$ is the total number of sentence in document A, and $d_{B,i}$ is defined similarly for document B. Under this assumption, each individual sentence in a document is equally important and probability mass is allocated uniformly to each sentence. Later, we will investigate alternative schemes to allocating probability mass to sentences.

Now let the i_{th} sentence be represented by a vector $v_i \in R^m$. This length-m dense embedding representation for each sentence allows us to define distances between the i_{th} and j_{th} sentences. We denote $\Delta(i, j)$ as the distance between the i_{th} and j_{th} sentences and let V denote the vocabulary size where the vocabulary is the unique set of sentences within a document pair. We follow previous works (Kusner et al., 2015) and use the Euclidean distance, $\Delta(i, j) = ||v_i - v_j||$. The SMD between a document pair is then the solution to the linear program:

$$SMD(A, B) = \min_{T \geq 0} \sum_{i=1}^{V} \sum_{j=1}^{V} T_{i,j} \times \Delta(i, j) \tag{2}$$

subject to:

$$\forall i \sum_{j=1}^{V} T_{i,j} = d_{A,i}$$
$$\forall j \sum_{i=1}^{V} T_{i,j} = d_{B,j}$$

Where $T \in R^{V \times V}$ is a nonnegative matrix, where each $T_{i,j}$ denotes how much of sentence i in document A is assigned to sentences j in document B, and constraints ensure the flow of a given sentence cannot exceed its allocated mass. Specifically, SMD ensures the the entire outgoing flow from sentence i equals $d_{A,i}$, i.e. $\sum_j T_{i,j} = d_{A,i}$. Additionally, the amount of incoming flow to sentence j must match $d_{B,j}$, i.e., $\sum_i T_{i,j} = d_{B,j}$.

4.2 Alternative Sentence Weighting Schemes

In Equation 1, each document is represented as a normalized bag-of-sentences (nBOS) where sentences are equally weighted. However, we posit that some sentences may be more semantically important than others.

Sentence Length Weighting The first insight we investigate is that documents will naturally be segmented into sentences of different lengths based on the language, content, and choice of segmentation. While Equation 1, treats each sentence equally, we posit that longer sentences should be assigned larger weighting than shorter sentences.

As such, we weight each sentence by the number of tokens in the sentence relative to the total number of tokens in the entire document, i.e., for the i_{th} sentence in the document A, we compute the weighting SL(i) as follows:

$$d_{A,i} = cnt(i) \cdot |i| / \sum_{s \in A} cnt(s) \cdot |s| \tag{3}$$

where $|i|$ and $|s|$ indicate the number of tokens in sentences i and s respectively. As such, longer sentence receive larger probability mass than shorter sentences.

IDF Weighting The second insight we investigate is that text segments such as titles and navigation text is ubiquitous in crawled data yet less semantically informative. Based on this insight, we apply a variant of inverse document frequency (IDF) – a weighting scheme common in the information retrieval space – to individual sentences (Robertson, 2004). Under this scheme, the more common a sentence is within a webdomain, the less mass the sentence will be allocated.

For sentence i in a web-domain D, we compute IDF(i) as follows:

$$d_{A,i} = 1 + \log \frac{|D|}{|\{d \in D : i \in d\}|} \tag{4}$$

where $|\{d \in D : s \in d\}|$ is the number of documents where the sentence s occurs and smoothing by 1 is performed to prevent 0 IDF.

SLIDF Weighting Finally, we propose combining both sentence length and inverse document frequency into a joint weighting scheme:

$$d_{A,i} = SL(i) \cdot IDF(i) \tag{5}$$

In this scheme, each sentence is weighted proportionally to the number of tokens it contains as well as by the IDF of the sentence within the domain. This weighting scheme is reminiscent of the use of tf-idf to determine word relevance (Ramos et al., 2003), but instead sentence length and idf are used to determine sentence importance.

4.3 Fast Distance Approximation

While EMD and other variants have demonstrated superior performance in many retrieval and classification tasks, they have also been shown to suffer from high computational complexity $\mathcal{O}(p^3 \log p)$, where p denotes the number of unique semantic units in a document pair. As such, we investigate techniques to speed up this computation.

Relaxed SMD Given the scalability challenges for computing WMD, simplified version of WMD was proposed that relaxes one of the two constraints in the original formulation (Kusner et al., 2015). Applying the same principle to SMD, we formulate:

$$SMD(A,B) = \min_{T \geq 0} \sum_{i=1}^{V} \sum_{j=1}^{V} T_{i,j} \times \Delta(i,j)$$

subject to: $\forall i \sum_{j=1}^{V} T_{i,j} = d_{A,i}$. Analogous to the relaxed-WMD, this relaxed problem yields a lower-bound to the SMD as every SMD solution satisfying both constraints remains a feasible solution if one constraint is removed. The optimal solution can be found by simply allocating the mass in each source sentence to the closest sentence in the target document.

The same computation can be performed in the reverse direction by removing the second constraint: $\forall j \sum_{i=1}^{V} T_{i,j} = d_{B,j}$. Similarly, the optimal solution allocates the mass sentences in the target document to the closest sentence in the source document. Both these distances can be calculated by computing the distance matrix between all pairs of sentences in $\mathcal{O}(p^2)$ time. For a tighter estimate of distance, the maximum of the two resultant distances can be used.

Greedy Mover's Distance We introduce an alternative to the relaxed-EMD variant wherein we keep both constraints in the transportation problem, but identify an approximate transportation scheme. This greedy mover's distance (GMD) finds the closest sentence pair between the source and target and moves as much mass between the two sentences as possible; the algorithm moves to the next closest until all mass has been moved while maintaining both constraints.

Algorithm 1: Greedy Mover's Distance

Input: d_s, d_t, w_s, w_t
Output: $\Delta(d_s, d_t)$

$pairs \leftarrow \{(s_s, s_t)$ for $s_s, s_t \in d_s \times d_t\}$ in ascending order by $\|s_s - s_t\|$
distance $\leftarrow 0.0$
for $s_s, s_t \in pairs$ **do**
  flow $\leftarrow min(w_s[s_s], w_t[s_t])$
  $w_s[s_s] \leftarrow w_s[s_s]$ − flow
  $w_t[s_t] \leftarrow w_t[s_t]$ − flow
  distance $\leftarrow$ distance + $\|s_s - s_t\| \times$ flow
end
return total

As seen in Algorithm 1, the algorithm takes a source document (d_s) and a target document (d_t) as well as the probability mass for the sentences in each: respectively w_s and w_t. The algorithm first computes the euclidean distance between each sentence pair from source to target and sorts these pairs in ascending order by their euclidean distance. The algorithm then iteratively chooses the closest sentence pair and moves the mass of the smallest sentence from the source to the target and subtracting this moved math from both. The algorithm terminates when all moveable mass has been moved. Unlike the exact solution to EMD, the runtime complexity is a more tractable $\mathcal{O}(|d_s||d_t| \times \log(|d_s||d_t|))$ which is dominated by the cost of sorting all candidate pairs. Unlike the relaxation, both constraints are satisfied but the transport is not necessarily optimal. As such, GMD yields an upper-bound to the exact computation.

We experimentally compare the effect of both approximation strategies on downstream document alignment in Section 7.

5 Document Matching Algorithm

In addition to a distance metric (i.e. SMD), we need a document matching algorithm to determine the best mapping between documents in two languages.

In our case, this works as follows: for any given webdomain, each document in the source document set, D_s is paired with each document in the target set, D_t, yielding $|D_s \times D_t|$ scored pairs – a fully connected bipartite graph representing all candidate pairings. Similar to previous works (Buck and Koehn, 2016b), the expected output assumes that each webpage in the non-dominant language has a translated or comparable counterpart. As visualized in Figure 1, this yields a $min(|D_s|, |D_t|)$ expected number of aligned pairs.

While an optimal matching maximizing scoring can be solved using the Hungarian algorithm (Munkres, 1957), the complexity of this algorithm is $\mathcal{O}(max(|D_s||D_t|)^3)$ which is intractable to even moderately sized web domains. As such, similar to the work in (Buck and Koehn, 2016b), a one-to-one matching between English and non-English documents is enforced by applying, competitive matching, a greedy bipartite matching algorithm.

Algorithm 2: Competitive Matching

Input: $P = \{(d_s, d_t) | d_s \in D_s, d_t \in D_t\}$
Output: $P' = \{(d_{s,i}, d_{t,i}), ...\} \subset P$

$scored \leftarrow \{(p, score(p))$ for $p \in P\}$
$sorted \leftarrow sort(scored)$ in ascending order
aligned $\leftarrow \varnothing$
$S_s \leftarrow \varnothing$
$S_t \leftarrow \varnothing$
for $d_s, d_t \in sorted$ **do**
  if $d_s \notin S_s \wedge d_t \notin S_t$ **then**
    $aligned \leftarrow aligned \cup \{(d_s, d_t)\}$
    $S_s \leftarrow S_s \cup d_s$
    $S_t \leftarrow S_t \cup d_t$
end
return aligned

In Algorithm 2, the algorithm first scores each candidate document pair using a distance function and then sorts pairs from closest to farthest. The algorithm then iteratively selects the closest document pair as long as the d_s and d_t of each pair have not been used in a previous (closer) pair. The algorithm terminates when $min(|D_s|, |D_t|)$ pairs have been selected. Unlike the Hungarian algorithm, the runtime complexity is a more tractable $\mathcal{O}(|D_s||D_t| \times \log(|D_s||D_t|))$ which is dominated by the cost of sorting all candidate pairs.

6 Experiments and Results

In this section, we explore the question of whether SMD can be used as a dissimilarity metric for the document alignment problem. Moreover, we explore which sentence weighting schemes yield the best results.

6.1 Experimental Setup

Dataset We evaluate on the test set from the URL-Aligned CommonCrawl dataset (El-Kishky et al., 2019) across 47 language directions.

Baseline Methods For comparison, we implemented two existing and intuitive document scoring baselines from (El-Kishky et al., 2019). The direct embedding (DE), directly embeds the entire content of a document using LASER. The second method sentence averaging (SA) embeds all sentences in a document using LASER and averages all embeddings to get a document representation. Cosine similarity on the embedded representation is used to compare documents.

SMD Weightings We evaluate four weighting schemes for SMD: (1) vanilla SMD with each sentence equally weighted(2) weighting by sentence length (SL) where SMD is computed under a scheme where each sentence is weighted by its length (number of tokens) normalized by the length of the entire document (3) weighting by inverse document frequence (IDF) where SMD is computed under a scheme where each sentence is weighted by the idf of the sentence (4) computing SMD under a scheme where each sentence is weighted by both sentence length and inverse document frequency (SLIDF). Under all these schemes, all weights are normalized to unit measure.

Distance approximation We use the greedy mover's distance approximation for all variants reported. In Section 7 we further explore the performance of the full distance computation and relaxed variants that were described in Section 4.3.

Evaluation Metric for Document Alignment Because the ground-truth document pairs only reflect a high-precision set of web-document pairs that are translations or of comparable content, there may be many other valid cross-lingual document pairs within each web-domain that are not included in the ground truth set. As such, we evaluate each method's generated document pairs solely on the recall (i.e. what percentage of the aligned pages in the test set are found) from the ground truth pairs.

For each scoring method, we score document pairs from the source and target languages within the same web-domain using the proposed document distance metrics described above. For the alignment, we report the performance for each distance metric after applying the competitive matching alignment algorithm as described in Algorithm 2.

6.2 Results

In Table 1, we first notice that constructing document representations by directly embedding (DE) the entire content of each document and computing document similarity using cosine similarity of the representation severely under-performs compared to individually embedding sentences and constructing the document representations by averaging the individual sentence representations within the document (SA). This is intuitive as LASER embeddings were trained on parallel sentences and embedding much larger documents directly using LASER results in poorer representations than by first embedding smaller sentences and combining them into the final document representation.

Comparing the basic SMD to the best performing baseline (SA), we see a 4%, 12%, and 20% improvement across high, mid, and low-resource directions respectively. This improvement suggests that summing sentence embeddings into a single document representation degrades the quality of the resultant document distances over computing document distances by keeping all sentence representations separate and computing distances between individual sentence pairs and combining these distances into a final document distance. This is more pronounced in lower-resource over higher-resource pairs which may be due to poorer lower-resource embeddings due to LASER being trained on fewer low-resource sentence pairs. As such averaging is more destructive to these representations while SMD avoids this degradation.

Further analysis verified the intuition that different sentences should be allocated different weighting in SMD. Assigning mass proportional to the

Language	Recall DE	Recall SA	Recall SMD	Recall SL	Recall IDF	Recall SLIDF
French	0.39	0.84	0.81	0.84	0.83	**0.85**
Spanish	0.34	0.53	0.59	0.63	0.62	**0.64**
Russian	0.06	0.64	0.69	0.69	0.70	**0.71**
German	0.52	0.74	**0.78**	0.76	0.77	0.77
Italian	0.22	0.47	0.55	0.56	0.56	**0.59**
Portuguese	0.17	0.36	0.39	**0.41**	0.38	0.40
Dutch	0.28	0.49	0.54	0.54	0.54	**0.56**
Indonesian	0.11	0.47	0.49	0.52	0.51	**0.53**
Polish	0.17	0.38	0.45	0.45	**0.46**	**0.46**
Turkish	0.12	0.38	0.52	0.56	0.57	**0.59**
Swedish	0.19	0.40	0.44	0.44	**0.46**	0.45
Danish	0.27	0.62	0.63	**0.69**	0.65	**0.69**
Czech	0.15	0.40	0.43	**0.44**	**0.44**	0.43
Bulgarian	0.07	0.43	0.52	0.54	**0.55**	0.52
Finnish	0.06	0.47	0.51	0.51	**0.54**	0.52
Norwegian	0.13	0.33	0.37	0.39	**0.42**	0.41
AVG	0.20	0.50	0.54	0.56	0.56	**0.57**

(a) High-resource directions.

Language	Recall DE	Recall SA	Recall SMD	Recall SL	Recall IDF	Recall SLIDF
Romanian	0.15	0.40	0.44	0.43	**0.45**	0.43
Vietnamese	0.06	0.28	0.29	0.29	0.29	**0.32**
Ukrainian	0.05	0.68	0.67	0.78	0.78	**0.82**
Greek	0.05	0.31	0.47	0.48	**0.49**	**0.49**
Korean	0.06	0.34	0.60	0.54	**0.61**	0.60
Arabic	0.04	0.32	0.63	0.59	**0.65**	0.61
Croatian	0.16	0.37	0.40	0.40	**0.41**	0.40
Slovak	0.20	0.41	0.46	**0.46**	**0.46**	0.44
Thai	0.02	0.19	0.41	0.33	**0.47**	0.41
Hebrew	0.05	0.18	0.39	**0.43**	0.41	0.41
Hindi	0.04	0.27	0.34	**0.54**	0.52	0.53
Hungarian	0.15	0.49	0.50	**0.54**	0.51	**0.54**
Lithuanian	0.11	0.73	0.79	0.79	**0.80**	**0.80**
Slovenian	0.13	0.33	0.34	0.35	**0.36**	**0.36**
Persian	0.06	0.32	0.56	0.57	0.53	**0.59**
AVG	0.09	0.37	0.49	0.50	**0.52**	**0.52**

(b) Mid-resource directions.

Language	Recall DE	Recall SA	Recall SMD	Recall SL	Recall IDF	Recall SLIDF
Estonian	0.28	0.52	0.69	0.66	**0.74**	0.72
Bengali	0.05	0.32	0.78	0.72	0.77	**0.79**
Albanian	0.23	0.56	**0.66**	0.65	0.65	**0.66**
Macedonian	0.02	0.33	0.32	0.36	**0.38**	0.33
Urdu	0.06	0.22	**0.60**	**0.60**	0.49	0.56
Serbian	0.06	0.59	**0.75**	0.74	0.74	0.71
Azerbaijani	0.08	0.34	0.74	0.74	**0.75**	0.74
Armenian	0.02	0.18	0.32	0.35	0.34	**0.38**
Belarusian	0.07	0.47	0.67	0.69	**0.73**	0.71
Georgian	0.06	0.24	0.46	**0.48**	0.45	0.45
Tamil	0.02	0.20	0.51	0.45	0.51	**0.53**
Marathi	0.02	0.11	0.43	**0.46**	0.33	0.39
Kazakh	0.05	0.31	0.44	**0.46**	0.45	0.45
Mongolian	0.03	0.13	0.18	0.22	0.21	**0.23**
Burmese	0.01	0.10	0.26	0.33	**0.46**	**0.46**
Bosnian	0.18	0.64	0.61	0.69	0.65	**0.72**
AVG	0.08	0.33	0.53	0.54	0.54	**0.55**

(c) Low-resource directions.

Table 1: Alignment recall on URL-aligned CommonCrawl dataset.

number of tokens in the sentence (SL), we see a 2%, 1% and 1% absolute improvement in recall in high, mid, and low-resource directions over assigning equal probability mass. This supports the claim that longer sentences should be allocated higher importance weight over shorter sentences as they contain more semantic content. The second assumption we investigated is that sentences that are common within a webdomain have less semantic importance and should be allocated less probability mass when computing SMD. After computing SMD with each sentence allocated mass according to inverse document frequency (IDF) and normalized to unit measure, we see a 2%, 3%, and 1% improvement over SMD for high, mid, and low-resource directions. Finally, when combining both sentence length and inverse document frequency (SLIDF) and normalizing to unit measure, we see a 3%, 3% and 2% absolute improvement in recall for high, mid, and low-resource directions. Overall, our SMD with SLIDF weighting scheme outperforms the sentence averaging baseline by 7% on high-resource directions, 15% on mid-resource directions, and 22% on low-resource directions.

7 Discussion

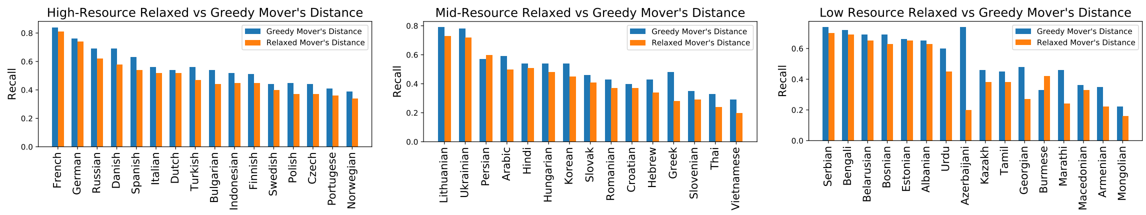
Although using sentences over words as the base semantic unit drastically reduces the overall cost of computing EMD-based metrics, the cubic computation still prohibits its use as a fast distance metric for large-scale alignment efforts. As such, in Section 4.3 we described two faster approximations to EMD computation: (1) a relaxation of constraints resulting in a lower bound and (2) a greedy algorithm for computing assigning transport representing an upper bound. We first analyze and compare the distances from each approximation scheme to the exact SMD computation.

Method	Tau	Recall	MAE	Runtime (s)
Exact-SMD	1.00	0.69	0.000	0.402
Relaxed-SMD	0.70	0.58	0.084	0.031
Greedy-SMD	0.98	0.69	0.010	0.107

Table 2: Comparing exact SMD computation to approximation schemes for computing SMD on 10 webdomains.

In Figure 3, we see that the distance computations for exact SMD and the greedy SMD approximation are highly correlated with small variance, while the relaxed approximation is less so with high variance. Additionally, as discussed in Section 4.3, the visualizations empirically suggest that our greedy approximation is a fairly tight upper bound while the relaxed approximation is a looser lower bound.

In Table 2, we compare quantitative metrics for the relaxed and greedy approximations to the exact solution of SMD on ten webdomains. Our first evaluation investigates how the approximate computation of distances affects the resultant ordering of document pairs. For the ten selected webdomains, we sort the document pairs in order by their computed distances and compare the ordering to the ordering induced by the exact computation of SMD. We evaluate the orderings using the Kendall-Tau metric (Kendall, 1938) which measures the agreement between the two rankings; if the agreement between the two rankings is perfect (i.e., the two



(a) High-resource directions. (b) Mid-resource directions. (c) Low-resource directions.

Figure 2: Document alignment results for different distance approximation techniques.

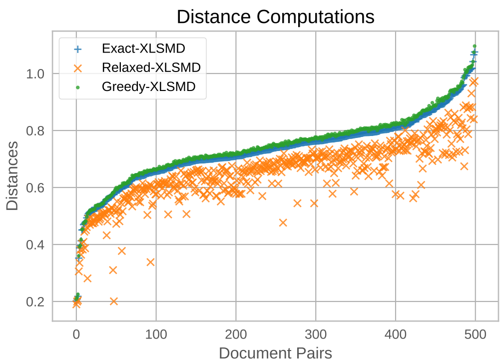


Figure 3: Exact, relaxed, and greedy-SMD distances sorted by Exact-SMD for a random selection of document pairs.

rankings are the same) the coefficient has value 1 and if the disagreement between the two rankings is perfect (i.e., one ranking is the reverse of the other) the coefficient has value -1. Intuitively, we would like the distances computed by an approximation to induce a similar ordering to the ordering by the exact distance computation. Comparing the Kendall-Tau for the relaxed and greedy approximations in relation to the exact computation shows that the order induced by the greedy approximation is very similar to the ordering induced by the exact computation while the relaxed approximation varies considerably. Additionally, the relaxed approximation demonstrates fairly high mean absolute error (MAE) and results in lower document alignment recall when compared to the exact computation of SMD, while our greedy approximation performs comparably and shows insignificant MAE. Finally, while the runtime of the relaxed computation is the fastest at 13 times faster than the exact computation, our greedy algorithm is approximately 4 times faster while delivering comparable document alignment performance to the exact computation and superior performance to the relaxed computation.

To ensure that the greedy algorithm consistently outperforms the relaxed algorithm on document alignment, we investigate the effect of using each approximation method on the downstream document alignment performance across 47 language pairs of varying resource availability.

Approximation	**Low**	**Mid**	**High**	**All**
Relaxed-SMD	0.44	0.43	0.50	0.46
Greedy-SMD	0.54	0.50	0.56	0.54

Table 3: Document alignment performance of fast methods for approximating the same variant of SMD.

As seen in Figure 2, in 45 of the 47 evaluated language pairs, our proposed Greedy Mover's Distance approximation yielded higher downstream recall in our alignment task over using the relaxed distance proposed for use in WMD (Kusner et al., 2015). In Table 3, we see a 10%, 7%, and 6% improvement in downstream recall across low, mid, and high-resource directions respectively. These results indicate that relaxing one of the two constraints in EMD is too lax for measuring an accurate distance. We posit this is because there are many sentences that can be considered "hubs" that are semantically close to many other sentences. These sentences can have a lot of probability mass allocated to them, resulting in a lower approximate EMD. Our greedy approximation ensures that both constraints are maintained even if the final result does not reflect the optimal transport.

8 Conclusion

In this paper, we introduce SMD a cross-lingual sentence mover's distance metric for automatically assessing the semantic similarity of two documents in different languages. We leverage state-of-the-art multilingual sentence embeddings and apply SMD to the task of cross-lingual document alignment. We demonstrate that our new metric outperforms other unsupervised metrics by a margin, especially in medium and low-resourced conditions.

References

Sadaf AbduI-Rauf and Holger Schwenk. 2009. On the use of comparable corpora to improve smt performance. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 16–23. Association for Computational Linguistics.

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.

Kubilay Atasu, Thomas Parnell, Celestine Dünner, Manolis Sifalakis, Haralampos Pozidis, Vasileios Vasileiadis, Michail Vlachos, Cesar Berrospi, and Abdel Labbi. 2017. Linear-complexity relaxed word mover's distance with gpu acceleration. In *2017 IEEE International Conference on Big Data (Big Data)*, pages 889–896. IEEE.

Georgios Balikas, Charlotte Laclau, Ievgen Redko, and Massih-Reza Amini. 2018. Cross-lingual document retrieval using regularized wasserstein distance. In *European Conference on Information Retrieval*, pages 398–410. Springer.

Christian Buck and Philipp Koehn. 2016a. Findings of the wmt 2016 bilingual document alignment shared task. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 554–563.

Christian Buck and Philipp Koehn. 2016b. Quick and reliable document alignment via tf/idf-weighted cosine distance. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 672–678.

Jiang Chen and Jian-Yun Nie. 2000. Parallel web text mining for cross-language ir. In *Content-Based Multimedia Information Access-Volume 1*, pages 62–77. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

Elizabeth Clark, Asli Celikyilmaz, and Noah A Smith. 2019. Sentence mover's similarity: Automatic evaluation for multi-sentence texts. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2748–2760.

Aswarth Abhilash Dara and Yiu-Chang Lin. 2016. Yoda system for wmt16 shared task: Bilingual document alignment. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 679–684.

Thi-Ngoc-Diep Do, Viet-Bac Le, Brigitte Bigi, Laurent Besacier, and Eric Castelli. 2009. Mining a comparable text corpus for a vietnamese-french statistical machine translation system. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 165–172. Association for Computational Linguistics.

Ahmed El-Kishky, Vishrav Chaudhary, Francisco Guzman, and Philipp Koehn. 2019. A massive collection of cross-lingual web-document pairs. *arXiv preprint arXiv:1911.06154*.

Miquel Esplà-Gomis, Mikel Forcada, Sergio Ortiz Rojas, and Jorge Ferrández-Tordera. 2016. Bitextor's participation in wmt'16: shared task on document alignment. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 685–691.

Pascale Fung and Lo Yuen Yee. 1998. An ir approach for translating new words from nonparallel, comparable texts. In *COLING 1998 Volume 1: The 17th International Conference on Computational Linguistics*.

Luís Gomes and Gabriel Pereira Lopes. 2016. First steps towards coverage-based document alignment. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 697–702.

Mandy Guo, Yinfei Yang, Keith Stevens, Daniel Cer, Heming Ge, Yun-hsuan Sung, Brian Strope, and Ray Kurzweil. 2019. Hierarchical document encoder for parallel corpus mining. In *Proceedings of the Fourth Conference on Machine Translation*, pages 64–72, Florence, Italy. Association for Computational Linguistics.

Gao Huang, Chuan Guo, Matt J Kusner, Yu Sun, Fei Sha, and Kilian Q Weinberger. 2016. Supervised word mover's distance. In *Advances in Neural Information Processing Systems*, pages 4862–4870.

Laurent Jakubina and Phillippe Langlais. 2016. Bad luc@ wmt 2016: a bilingual document alignment platform based on lucene. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 703–709.

Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

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

Philipp Koehn et al. 2002. Europarl: A multilingual corpus for evaluation of machine translation.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *International conference on machine learning*, pages 957–966.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Marek Medveď, Miloš Jakubícek, and Vojtech Kovár. 2016. English-french document alignment based on keywords and statistical translation. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 728–732.

James Munkres. 1957. Algorithms for the assignment and transportation problems. *Journal of the society for industrial and applied mathematics*, 5(1):32–38.

Dragos Stefan Munteanu and Daniel Marcu. 2005. Improving machine translation performance by exploiting non-parallel corpora. *Computational Linguistics*, 31(4):477–504.

Dragos Stefan Munteanu and Daniel Marcu. 2006. Extracting parallel sub-sentential fragments from non-parallel corpora. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, pages 81–88. Association for Computational Linguistics.

Vassilis Papavassiliou, Prokopis Prokopidis, and Stelios Piperidis. 2016. The ilsp/arc submission to the wmt 2016 bilingual document alignment shared task. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 733–739.

Alexandre Rafalovitch, Robert Dale, et al. 2009. United nations general assembly resolutions: A six-language parallel corpus. In *Proceedings of Machine Translation Summit XII*.

Juan Ramos et al. 2003. Using tf-idf to determine word relevance in document queries. In *Proceedings of the first instructional conference on machine learning*, volume 242, pages 133–142. Piscataway, NJ.

Reinhard Rapp. 1999. Automatic identification of word translations from unrelated english and german corpora. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 519–526. Association for Computational Linguistics.

Philip Resnik. 1999. Mining the web for bilingual text. In *Proceedings of the 37th annual meeting of the Association for Computational Linguistics on Computational Linguistics*, pages 527–534. Association for Computational Linguistics.

Philip Resnik and Noah A Smith. 2003. The web as a parallel corpus. *Computational Linguistics*, 29(3):349–380.

Stephen Robertson. 2004. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520.

Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. 1998. A metric for distributions with applications to image databases. In *Sixth International Conference on Computer Vision (IEEE Cat. No. 98CH36271)*, pages 59–66. IEEE.

Vadim Shchukin, Dmitry Khristich, and Irina Galinskaya. 2016. Word clustering approach to bilingual document alignment (wmt 2016 shared task). In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 740–744.

Raghavendra Udupa, K Saravanan, A Kumaran, and Jagadeesh Jagarlamudi. 2009. Mint: A method for effective and scalable mining of named entity transliterations from large comparable corpora. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, pages 799–807. Association for Computational Linguistics.

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