Majority Voting with Bidirectional Pre-translation For Bitext Retrieval

Alex Jones
Dartmouth College
alexander.g.jones.23@dartmouth.edu

Derry Tanti Wijaya
Boston University
wijaya@bu.edu

Abstract

Obtaining high-quality parallel corpora is of paramount importance for training NMT systems. However, as many language pairs lack adequate gold-standard training data, a popular approach has been to mine so-called "pseudo-parallel" sentences from paired documents in two languages. In this paper, we outline some drawbacks with current methods that rely on an embedding similarity threshold, and propose a heuristic method in its place. Our method involves *translating both halves of a paired corpus before mining*, and then performing a majority vote on sentence pairs mined in three ways: after translating documents in language $x \rightarrow$ language y, after translating $y \rightarrow x$, and using the original documents in languages x and y. We demonstrate success with this novel approach on the Tatoeba similarity search benchmark in 64 low-resource languages, and on NMT in Kazakh and Gujarati. We also uncover the effect of *resource-related factors* (i.e. how much monolingual/bilingual data is available for a given language) on the *optimal choice of bitext mining method*, demonstrating that there is currently no one-size-fits-all approach for this task. We make the code and data used in our experiments publicly available.[1]

1 Introduction

Mining so-called "pseudo-parallel" sentences from sets of similar documents in different languages ("comparable corpora") has gained popularity in recent years as a means of overcoming the dearth of parallel training data for many language pairs. With increasingly powerful computational resources and highly efficient tools such as Faiss (Johnson et al., 2017) at our disposal, the possibility of mining billions of pseudo-parallel bitexts for thousands of language pairs to the end of training a multilingual NMT system has been realized. For example, Fan et al. (2020) perform global mining over billions of sentences in 100 languages, resulting in a massively multilingual NMT system that supports supervised translation in 2200 directions.

Despite these breakthroughs in high-resource engineering, many questions remain to be answered about bitext mining from a research perspective, with particular attention directed toward the *low-resource engineering case*, i.e. research settings with limited computational resources. While Fan et al. (2020) yield impressive results using hundreds of GPUs, aggressive computational optimization, and a global bitext mining procedure (i.e. searching the entire target corpus for a source sentence match), how these results transfer to the low computational resource case is not clear. Moreover, the effect of circumstantial (e.g. the resources available for a given language or language pair) or linguistic (e.g. typological) factors on bitext mining performance remains highly understudied.

In light of these issues, our contributions are as follows:

- We demonstrate the problematic nature of using similarity-score-based thresholding for mining bitexts, with particular attention given to document-level mining of low-resource languages.
- We propose a novel, heuristic approach for bitext mining that involves translating both halves of a bilingual corpus, mining with three sets of documents (two distinct translated pairs of documents plus the original documents), and then performing a majority vote on the resulting sentence pairs. This approach avoids the pitfalls of laboriously tuning a similarity score threshold, a practice we believe to have been weakly motivated in past studies.

[1] https://github.com/AlexJonesNLP/alt-bitexts

Proceedings of 14th Workshop on Building and Using Comparable Corpora in conjunction with RANLP-2021, pages 46–59, held online, 6 September 2021.
https://doi.org/10.26615/978-954-452-076-2_007

- We show the success of our method on NMT in English-Kazakh and English-Gujarati, and also on the gold-standard bitext retrieval task (“similarity search" on the Tatoeba dataset), and show the optimal choice of mining approach to be partially dependent on the resource availability of the language(s) involved.

2 Related Work

Mining pseudo-parallel sentences from paired corpora for the purpose of training NMT systems is a decades-old problem, and dozens of solutions have been tried, ranging from statistical or heuristic-based approaches (Zhao and Vogel, 2002; Resnik and Smith, 2003; Munteanu et al., 2004; Fung and Cheung, 2004; Munteanu and Marcu, 2006) to similarity-based, rule-based, and hybrid approaches (Azpeitia et al., 2017, 2018; Bouamor and Sajjad, 2018; Hangya et al., 2018; Schwenk, 2018; Ramesh and Sankaranarayanan, 2018; Artetxe and Schwenk, 2019a,b; Hangya and Fraser, 2019; Schwenk et al., 2019a,b; Wu et al., 2019; Keung et al., 2020; Tran et al., 2020; Kvapilíková et al., 2020; Feng et al., 2020; Fan et al., 2020). Benchmarks to measure performance on this task include the BUCC[2] ’17/18 datasets (Zweigenbaum et al., 2017, 2018), whose task involves spotting gold-standard bitexts within comparable corpora, and the Tatoeba dataset (Artetxe and Schwenk, 2019b), whose task involves matching gold-standard pairs in truly parallel corpora.

Relevant to similarity-based mining methods are well-aligned cross-lingual word and sentence embeddings, which are some of the oldest constructs in NLP and have been tackled using hundreds of diverse approaches. Even among relatively recent efforts, these approaches range from static, monolingual embeddings (Pennington et al., 2014; Mikolov et al., 2013; Arora et al., 2017; Kiros et al., 2015) to static, multilingual ones (Klementiev et al., 2012; Ammar et al., 2016; Schwenk and Douze, 2017) to contextualized, monolingual ones (Peters et al., 2018; Subramanian et al., 2018; Devlin et al., 2019; Liu et al., 2019; Conneau et al., 2017; Reimers and Gurevych, 2019) to contextualized, multilingual ones (Song et al., 2019; Conneau et al., 2020; Reimers and Gurevych, 2020; Feng et al., 2020; Wang et al., 2019). In this paper, our approach centers around using *contextualized, multilingual sentence* embeddings for the task of bitext mining.

For low-resource languages where parallel training data is little to none, unsupervised NMT can play a crucial role (Artetxe et al., 2018a, 2019a,b, 2018b; Hoang et al., 2018; Lample et al., 2017, 2018b,c; Pourdamghani et al., 2019; Wu et al., 2019). However, previous works have only focused on high-resource languages and/or languages that are typologically similar to English. Most recently, several works have questioned the universal usefulness of unsupervised NMT and showed its poor results for low-resource languages (Kim et al., 2020; Marchisio et al., 2020). They note the importance of typological similarity between source and target language, in addition to domain proximity and the size and quality of the monolingual corpora involved. They reason that since these conditions can hardly be satisfied in the case of low-resource languages, they result in poor unsupervised performance for these languages. However, recently it has been shown that training a language model on monolingual corpora, followed by training with an unsupervised MT objective, and then training on mined comparable data (Kuwanto et al., 2021) can improve MT performance for low-resource languages. In this work, we explore the usefulness of our mined bitext using a similar pipeline. We show an improvement over using only supervised training data for low-resource MT.

3 Model selection

3.1 Cross-lingual Sentence Embeddings

We initially experiment with XLM-RoBERTa (Conneau et al., 2020) for our bitext mining task, using averaged token embeddings (Keung et al., 2020) or the [CLS] (final) token embedding as makeshift sentence embeddings. However, we replicate results from Reimers and Gurevych (2020) in showing these ad-hoc sentence embeddings to have relatively poor performance on the BUCC ’17/18 EN-FR train data (Zweigenbaum et al., 2017, 2018) compared to bona fide sentence embeddings like LASER (Artetxe and Schwenk, 2019b) and LaBSE (Feng et al., 2020). Thus, we opt to use LaBSE as our sentence embedding model, using its implementation in the Sentence Transformers [3] library. LaBSE performs state-of-the-art (SOTA) or near-SOTA on the BUCC and Tatoeba datasets

[2] Building and Using Comparable Corpora

[3] https://www.sbert.net

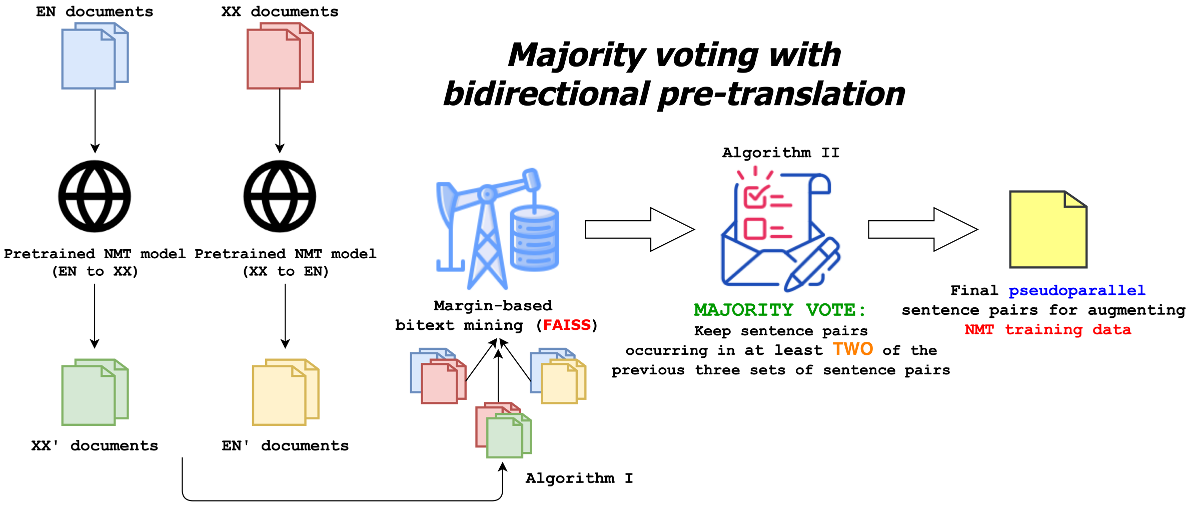
EN documents
XX documents
Majority voting with bidirectional pre-translation
Algorithm II
Pretrained NMT model (EN to XX)
Pretrained NMT model (XX to EN)
Margin-based bitext mining (FAISS)
MAJORITY VOTE: Keep sentence pairs occurring in at least TWO of the previous three sets of sentence pairs
Final pseudoparallel sentence pairs for augmenting NMT training data
XX' documents
EN' documents
Algorithm I

Figure 1: The pipeline we offer for selecting sentence translation pairs from comparable or parallel (e.g. Tatoeba) corpora using a heuristic voting approach. See Algorithms 1 and 2 for further details.

(Artetxe and Schwenk, 2019b)[4], and has demonstrated cross-lingual transfer capabilities for low-resource languages in particular. Moreover, being more recent than LASER, LaBSE has been investigated less thoroughly in the context of the bitext mining task.

4 Methods

An overview of our method for extracting bitexts is given in Figure 1; the processes are sketched in greater detail in Algorithms 1 and 2. The retrieval process begins with a set of English documents and a set of documents in another language XX. Both sets of documents are then translated using a pretrained NMT model to obtain XX' documents (English documents translated to XX) and EN' documents (XX documents translated to English).

We then perform margin-based translation mining (described below in Section 4.1 and in Algorithm 1) on three sets of documents: the original EN-XX documents, the EN-EN' documents, and the XX-XX' documents. Lastly, we perform a majority vote (see Algorithm 2, “majority voting”) on the resulting sentence pairs, keeping any pair that occurs in $\geq$ 2 of the three sets of sentence pairs. If mined from a comparable corpus such as Wikipedia, these pseudoparallel sentence pairs can then be used to augment the training data of the pretrained NMT models, or (help) train an NMT model from scratch, as in Fan et al. (2020).

Alternative methods for filtering an initial set of sentence pairs are also given in Algorithm 2 (see comments in blue). Empirically, we find our majority voting method to be superior when a pretrained NMT model is available for both languages, while vanilla margin-based mining (Artetxe and Schwenk, 2019a) performs best in the absence of a pretrained NMT model. Results are discussed in greater detail in Section 6.

4.1 Primary retrieval procedure: Margin-based Mining

For our primary mining procedure, we use margin-based mining as described in Artetxe and Schwenk (2019a). Seeking to mitigate the hubness problem (Dinu et al., 2014), margin scoring poses an alternative to raw cosine similarity in that it selects the candidate embedding that "stands out" the most from its k nearest neighbors. We use the *ratio* margin score, as described in Artetxe and Schwenk (2019a) and defined below:

$$\text{score}(x, y) = \frac{\cos(x, y)}{\frac{1}{2k}\left(\sum_{z \in NN_k(x)} \cos(x, z) + \sum_{z \in NN_k(y)} \cos(y, z)\right)} \quad (1)$$

As in Artetxe and Schwenk (2019a), we use $k = 4$ for all our mining procedures. We acknowledge that k is indeed a tuneable and important hyperparameter of KNN search, and that higher values of k may work better for bitext mining in certain scenarios, depending on factors such as the size of the search space (Schwenk et al., 2019b).

[4] https://github.com/facebookresearch/LASER/tree/master/data/tatoeba/v1

Algorithm 1: Doc-level margin-based mining

```
Given $\mathcal{X}$, $\mathcal{Y}$, $k$, $t$, JOIN_METHOD
$\mathcal{X}$: Set of sentences in language X. May be grouped
    into documents or standalone sentences.
$\mathcal{Y}$: Set of sentences in language Y that are parallel or
    comparable to those in $\mathcal{X}$.
$k$: Number of neighbors
JOIN_METHOD: Method of combining sentence
    pairs after mining in the forward and backward
    directions. One of either INTERSECT or UNION.
$t$: Margin similarity threshold

MINE SENTENCE PAIRS IN BOTH DIRECTIONS
for document $\mathcal{D} \in \mathcal{X}$ do
    for $x \in \mathcal{D}$ do                                   Mine in the forward direction
        $nn_x \leftarrow NN(x, \mathcal{Y}_{\mathcal{D}}, k)$ ;
          // $NN(x, \mathcal{D}, k)$ := Faiss
          $k$-nearest neighbors search
        $best_y = \text{argmax}_{y \in nn_x}\ score(x, y)$ ;
          // score(x,y) := Eq.(1)
        if $score(x, best_y) > t$ then
            $fwd_D \leftarrow (x, best_y)$
        end
        $fwd \leftarrow fwd_D$
    end
end
for $\mathcal{D} \in \mathcal{Y}$ do
    for $y \in \mathcal{D}$ do                                   Mine in the backward direction
        $nn_y \leftarrow NN(y, \mathcal{X}_{\mathcal{D}}, k)$
          $best_x = \text{argmax}_{x \in nn_y}\ score(y, x)$
        if $score(best_x, y) > t$ then
            $bwd_D \leftarrow (best_x, y)$
        end
        $bwd \leftarrow bwd_D$
    end
end
if INTERSECT then
    $\mathcal{P} \leftarrow \{fwd\} \cap \{bwd\}$
end
else if UNION then
    $\mathcal{P} \leftarrow \{fwd\} \cup \{bwd\}$
end
return $\mathcal{P}$
```

However, we don't make this hyperparameter a focus of this paper, instead addressing the problem of margin score thresholding and its relation to the size of the search space. We leave a thorough examination of k and its effect on bitext mining performance for future work.

4.2 Filtering Procedures

4.2.1 Thresholding

The most straightforward measure for filtering mined sentence pairs after an initial ("primary") mining pass is to set a similarity score threshold, as shown in Artetxe and Schwenk (2019a). Of course, there is a precision-recall trade-off inherent to adjusting this threshold, and we show that simply using a threshold is problematic in two other ways as well: (1) in the case of document-level mining, the size of the search space (document size) is variable, so a threshold that works well for one document may function poorly for another; and (2) when mining bitexts for NMT training, it can be incredibly expensive to tune this threshold as a hyperparameter, as this entails re-training of the NMT system. Our heuristic method outperforms a previously used margin score threshold (Schwenk et al., 2019b,a; Fan et al., 2020) on document-level mining for Kazakh and Gujarati, doesn't require tuning any hyperparameter, and works for any language for which a supervised MT system is available.

Algorithm 2: Secondary retrieval procedures

```
Given $\mathcal{X}, \mathcal{Y}, k, t, \mathcal{M}$, JOIN_METHOD
$t$:    Margin score threshold
$\mathcal{M}$: An NMT model
if TRANSLATE then
    if EN_TO_XX then
        for $x \in \mathcal{X}$ do
            $\mathcal{X}_{trans} \leftarrow \mathcal{M}(x \rightarrow lang_y)$
            $\mathcal{P}_{en\_xx} \leftarrow$
              Algorithm1($\mathcal{X}_{trans}, \mathcal{Y}, k, JOIN\_METHOD, t$)
        end
        if not STRICT_INT or PAIRWISE_INT then
          ;   // EN-to-XX trans. only
        return $\mathcal{P}_{en\_xx}$
    end
    if XX_TO_EN then
        for $y \in \mathcal{Y}$ do
            $\mathcal{Y}_{trans} \leftarrow \mathcal{M}(y \rightarrow lang_x)$
            $\mathcal{P}_{xx\_en} \leftarrow$
              Algorithm1($\mathcal{Y}_{trans}, \mathcal{X}, k, JOIN\_METHOD, t$)
        end
        if not STRICT_INT or PAIRWISE_INT then
          ;  // XX-to-EN trans. only
        return $\mathcal{P}_{xx\_en}$
    end
end
$\mathcal{P}_{orig} \leftarrow$ Algorithm1($\mathcal{X}, \mathcal{Y}, k, JOIN\_METHOD, t$)
  ;             // All-or-nothing voting
if STRICT_INT then
    return $\mathcal{P}_{orig} \cap \mathcal{P}_{en\_xx} \cap \mathcal{P}_{xx\_en}$
end
  ;       // Majority voting (preferred)
else if PAIRWISE_INT then
    return $\mathcal{P}_{orig} \cap \mathcal{P}_{en\_xx} \bigcup \mathcal{P}_{orig} \cap$
      $\mathcal{P}_{xx\_en} \bigcup \mathcal{P}_{en\_xx} \cap \mathcal{P}_{xx\_en}$
end
  ;                          // Vanilla mining
else
    return $\mathcal{P}_{orig}$
end
```

4.2.2 Pre-translation

Our approach capitalizes on multiple similarity-related signals by first translating either the source texts (i.e. en→xx), target texts (xx→en), or both. In our experiments on the Tatoeba dataset (Artetxe and Schwenk, 2019b), we translate with Google

Translate / GNMT (Wu et al., 2016) using Cloud Translation API. However, due to the cost of using this API on large bodies of text, when mining on the English-Kazakh and English-Gujarati comparable corpora, we use an NMT system that we train on WMT'19 data (Barrault et al., 2019), with training corpora sizes given in Table 1. When translating in either direction, we translate the entire corpus, e.g. all English sentences in the Wikipedia corpus are translated to Kazakh.

4.3 Supervised and Unsupervised NMT

We follow the same pipeline for training MT in (Kuwanto et al., 2021) that is based on XLM (Conneau and Lample, 2019). Following their pipeline, we first pretrain a bilingual Language Model (LM) using the Masked Language Model (MLM) objective (Devlin et al., 2019) on the monolingual corpora of two languages (e.g. Kazakh and English for en-kk) obtained from Wikipedia, WMT 2018/2019[5] and Leipzig corpora (2016)[6]. For both the LM pretraining and NMT model fine-tuning, unless otherwise noted, we follow the hyperparameter settings suggested in the XLM repository[7]. For every language pair we extract a shared 60,000 subword vocabulary using Byte-Pair Encoding (BPE) (Sennrich et al., 2016). After pretraining the LM, we train an NMT model in an unsupervised manner following the setup recommended in Conneau and Lample (2019), where both encoder and decoder are initialized using the same pretrained encoder block. For training unsupervised NMT, we use back-translation (*BT*) and denoising auto-encoding (*AE*) losses (Lample et al., 2018a), and the same monolingual data as in LM pretraining. Lastly, to train a supervised MT model using our mined comparable data, we follow *BT+AE* with *BT+MT*, where *MT* stands for supervised machine translation objective for which we use the mined data. We stop training when the validation perplexity (LM pre-training) or BLEU (translation training) was not improved for ten checkpoints. We run all our experiments on 2 GPUs, each with 12GB memory.

We compare the performance in terms of BLEU score of our MT model with a model that follows the same pipeline (LM pre-training, unsupervised MT training, followed by supervised MT training) but that uses gold-standard training data from WMT19 (Table 1). The sizes of the monolingual data we use for LM pretraining are also shown in Table 1.

Train data	**Number of sentences**	
	en-kk	**en-gu**
Monolingual	9.51M	1.36M
Supervised		
WMT'19	222,165	22,321
Comparable		
Doc-level mining, threshold = 1.06	430,762	120,989
Doc-level mining with bidirectional pretranslation → majority voting	154,679	113,955

Table 1: Sizes (in number of sentences) of training corpora used in training supervised and semi-supervised NMT. The comparable/pseudoparallel sentences are mined using margin-based scoring with LaBSE with the indicated secondary retrieval procedures. These procedures are described in Section 4.

5 Experiments

5.1 Gold-standard Bitext Retrieval

In gold-standard bitext retrieval tasks, the goal is to mine gold-standard bitexts from a set of parallel or comparable corpora. We use the common approach of finding k-nearest neighbors for each sentence pair (in both directions, if using INTERSECT in Algorithm 1), then choosing the sentence that maximizes the ratio margin score (Equation 1 in Section 4.1).

Tatoeba Dataset[8] The Tatoeba dataset, introduced by Artetxe and Schwenk (2019b), contains up to 1,000 English-aligned, gold-standard sentence pairs for 112 languages. In light of our focus on lower-resource languages, we experiment only on the languages listed in Table 10 of Reimers and Gurevych (2020), which are languages without parallel data for the distillation process they undertake. This heuristic choice is supported by relative performance against languages *with* parallel data for distillation: the average raw cosine similarity baseline with LaBSE for the latter was 96.3, in contrast with 73.7 for the former. Specifically, the ISO 639-2 codes[9] for the languages we use are as follows:

[5] http://data.statmt.org/news-crawl/

[6] https://wortschatz.uni-leipzig.de/en/download/

[7] http://github.com/facebookresearch/XLM

[8] https://github.com/facebookresearch/LASER/tree/master/data/tatoeba/v1

[9] https://www.loc.gov/standards/iso639-2/php/code_list.php

afr, amh, ang, arq, arz, ast, awa, aze, bel, ben, ber, bos, bre, cbk, ceb, cha, cor, csb, cym, dsb, dtp, epo, eus, fao, fry, gla, gle, gsw, hsb, ido, ile, ina, isl, jav, ksb, kaz, khm, kur, kzj, lat, lfn, mal, mhr, nds, nno, nov, oci, orv, pam, pms, swg, swh, tam, tat, tel, tgl tuk, tzl, uig, uzb, war, wuu, xho, yid.

5.2 Pseudo-parallel Sentences From Comparable Corpora

In addition to gold-standard bitext mining, we also mine pseudo-parallel sentences from comparable corpora. The aim of this task is as follows: given two sets of similar documents in different languages, find sentence pairs that are close enough to being translations to act as training data for an NMT system. Of course, unlike the gold-standard mining task, there are not ground-truth labels present for this task, and so evaluation must be performed on a downstream task like NMT.

Comparable Corpora Our comparable data is mined from comparable documents, which are linked Wikipedia pages in different languages obtained using the langlinks from Wikimedia dumps. For each sentence in a foreign language Wikipedia page, we use all sentences in its corresponding linked English language Wikipedia page as potential comparable sentences.

Pre-processing Since our comparable corpora for both EN-KK and EN-GU are grouped into documents, the most important pre-processing step we perform is eliminating especially short documents before similarity search. The motivation for this is that since we search at document-level, the quality of the resulting pairs could be highly degraded in particularly small search spaces, in a way that neither thresholding nor voting could mitigate. Note that average document length was much shorter for both Gujarati and Kazakh than for English, due simply to shorter Wikipedia articles in those languages. For the EN-KK corpus, we omit any paired documents whose English version was < 30 words or whose Kazakh version was < 8 words, which we determine somewhat arbitrarily by seeing what values allowed for a sufficient number of remaining sentences. For the EN-GU corpus, we take a more disciplined approach and lop off the bottom 35% of shortest document pairs, which happened to be $document\_length = 21$ sentences for English and 5 sentences for Gujarati. This step accounted for the large number of documents in each corpus that contained very few sentences.

5.3 NMT Training Data

We conduct experiments on Kazakh and Gujarati. They are spoken by 22M and 55M speakers worldwide, respectively. Additionally, the languages have few parallel but some comparable and/or monolingual data available, which makes them ideal and important candidates for our low-resource unsupervised NMT research.

Our monolingual data for LM pre-training of these languages (shown in Table 1) are carefully chosen from the same topics (for Wikipedia) and the same domain (for news data). For the news data, we also select data from similar time periods (late 2010s) to mitigate domain discrepancy between source and target languages as per previous research (Kim et al., 2020). We also randomly downsample the English part of WMT NewsCrawl corpus so that our English and the corresponding foreign news data are equal in size.

6 Results & Analysis

6.1 Tatoeba Dataset

We mine bitexts on the Tatoeba test set in 64 generally low-resource languages (listed in Section 5.1) using the primary mining procedure described in Algorithm 1 with *intersection* retrieval, in addition to seven different secondary mining procedures, namely:

1. Cosine similarity (Reimers and Gurevych, 2020)
2. Margin scoring with no threshold
3. Margin scoring, threshold=1.06
4. Margin scoring, threshold=1.20 (shown to be optimal on BUCC mining task [10])
5. Margin scoring using EN sentences translated to XX
6. Margin scoring using XX sentences translated to EN
7. The *strict* intersection of pairs generated by methods 2, 5, and 6
8. The *pairwise* intersection of pairs generated by method 2, 5, and 6 (majority voting)

We report F1 instead of accuracy because the intersection methods (in both primary and secondary procedures) permit less than 100% recall.

The results are broken down across languages by resource availability (as in "high-resource" or "low-

[10] https://www.sbert.net/examples/applications/parallel-sentence-mining/README.html

Procedure	Average gain over baseline (best results only)	Average gain over baseline (all results)	Average gain over baseline (langs with transl. support)	Best results by resource capacity*	Average gain over baseline (by resource capacity)
Margin scoring only (Artetxe and Schwenk, 2019a)	**+6.9**	**+5.2**	+3.6	Level 0: 6 lang. Level 1: 18 lang. Level 2: 2 lang. Level 3: 2 lang. 2†, 6‡	Level 0: +7.2 Level 1: +**5.2** Level 2: +1.8 Level 3: +3.4 Level 4: +1.0
xx-to-en translation → margin scoring	+5.2	+3.3	+3.3	Level 0: 1 lang. Level 1: 7 lang. Level 2: 2 lang. Level 3: 7 lang. Level 4: 1 lang.	Level 0: +3.9 Level 1: +2.8 Level 2: +0.1 Level 3: +**4.3** Level 4: +**1.8**
Bidirectional translation → margin scoring → pairwise intersection of three sets of sentence pairs	+4.6	+4.0	**+4.0**	Level 0: 2 lang. Level 1: 3 lang. Level 2: 2 lang. Level 3: 1 lang.	Level 0: +**7.3** Level 1: +3.9 Level 2: +**2.6** Level 3: +4.0 Level 4: +1.0

* Using resource categorizations from Joshi et al. (2020) † Extinct languages ‡ Constructed (artificial) languages

Table 2: Average gain (F1) over the baseline for each mining method on the low-resource subset of the Tatoeba test data, broken down by several categories. The baseline is the F1 achieved using raw cosine similarity with LaBSE. The "best results" for a given method are those results on which that method achieved superior results compared to all other methods. "All results" refers to all languages in the Tatoeba test set.

Corpus	**Language pair**			
	kk→en	**en→kk**	**gu→en**	**en→gu**
Unsupervised				
Kim et al. (2020)	2.0	0.8	0.6	0.6
Supervised				
WMT'19 (Kim et al., 2020)	10.3	2.4	9.9	3.5
WMT'19 (Tran et al., 2020) Iter 1	9.8	3.4	8.1	8.1
WMT'19 (Tran et al., 2020) Iter 3	**13.2**	4.3	**18.0**	16.9
Google MT (Wu et al., 2016)	**28.9**	**23.1**	**26.2**	**31.4**
Our pipeline: unsup.+sup.				
WMT'19	11.2	7.3	5.7	10.2
Threshold=1.06	6.6	4.1	16.2	19.8
Majority voting	8.6	6.1	16.4	**20.2**
Threshold=1.06+WMT'19	11.8	7.9	15.4	18.5
Majority voting+WMT'19	12.6	**9.0**	15.8	19.1

Table 3: NMT training schemes and corresponding BLEU scores on WMT'19 test set. We train supervised systems with gold-standard data, comparable/pseudoparallel ("silver-standard") data, and combinations of both. We also try supplementing unsupervised training with each of these three types of supervised data. We provide a supervised benchmark from Wu et al. (2016).

resource"), as ranked on a 0-5 scale[11] according to Joshi et al. (2020). These results are summarized in Table 2. We only display results for simple margin scoring (with no threshold), margin scoring with XX-to-EN translation beforehand, and margin scoring with bidirectional pre-translation + majority voting, as these are the best-performing methods for the Tatoeba bitext retrieval task.

Because many of the languages in Table 2 lack support in GNMT, the dominant method overall is simple margin scoring, being the best-performing method on 28/64 languages[12] and seeing an aver-

[11] rb.gy/psmfnz

[12] 6/64 languages lack a resource categorization, so we re-

age gain over the baseline of +5.2 for all languages and +6.9 for languages on which it was the best-performing method. However, for languages with *translation support* (i.e. for which a supervised NMT system is available), the majority voting approach won out, with an average gain over the baseline of +4.0, in contrast to vanilla margin scoring (+3.6). In fact, among these 38 languages, vanilla margin scoring outperformed translation-based or hybrid (intersection) methods on only 11 languages.

Simply translating non-English sentences into English before mining (Method 6) also performed well, netting best results on 18 languages and outperforming other methods on resource level 3 (+4.3 F1 over baseline) and level 4 (+1.8) languages. Meanwhile, pairwise intersection performed best on level 0 (+7.3) and level 2 (+2.6) languages, with vanilla margin scoring outperforming other approaches on level 1 (+5.2).

These results show that the optimal choice of mining approach is very much dependent on the resource availability of the languages involved (most directly, the amount of data available during pre-training), and that if a supervised MT system is already available for a given language, that system can be used for efficient mining of parallel or pseudo-parallel sentences, in tandem with a pre-trained language model like LaBSE. As shown in Table 2, even high-resource (i.e. level 4) languages can be helped by pre-translation of paired corpora.

6.2 NMT

In Table 3, we show the performance in terms of BLEU scores of various NMT training schemes on the same WMT'19 test set. We train the supervised MT part of our pipeline system with gold-standard (WMT'19) data, our mined comparable/pseudoparallel ("silver-standard") data, and combinations of both i.e., training with comparable data followed by training with gold-standard data. We also provide Google massively multilingual MT performance on the same WMT'19 test set (Wu et al., 2016).

As can be seen in Table 3, our method of mining bitext without thresholding results in higher BLEU performance than when using bitexts mined using margin scoring with a threshold of 1.06, which is a commonly used threshold recommended by previous works for margin-based mining (Schwenk et al., 2019b,a). Our preferred method also results in the best en→gu performance, which outperforms previous unsupervised or supervised works. It outperforms the best previous work that uses WMT'19 data and iterative bitext mining by +3.3 BLEU. Since we do not perform iterative mining, if we consider the same previous work without iterative mining i.e., Tran et al. (2020) Iter 1, our approach outperforms that model by +12.1 BLEU in en→gu direction and by +8.3 BLEU in gu→en direction.

When combined with supervised i.e., gold-standard, data for training, our method for mining bitext which does not use any thresholding (majority voting+WMT'19) also outperforms the same model which uses bitext mined using margin scoring with a threshold of 1.06 (Threshold=1.06+WMT'19). Majority voting+WMT'19 also results in the best en→kk performance, which outperforms previous unsupervised or supervised works. It outperforms the best previous work that uses WMT'19 data and iterative bitext mining by +4.7 BLEU. Since we do not perform iterative mining, if we consider the same previous work without iterative mining i.e., Tran et al. (2020) Iter 1, our approach outperforms that model by +5.6 BLEU in the en→kk direction and by +2.8 BLEU in the kk→en direction. It is also worth noting that for training our pipeline model we use the default hyperparameter settings suggested in the XLM repository, while previous works perform extensive hyperparameter tuning. We believe our performance can be improved further by tuning our hyperparameter settings, but for brevity leave this for a future study. These results on low resource MT further demonstrate the superiority of our method for mining bitext without thresholding—compared to margin scoring *with* thresholding—for downstream low-resource MT applications. To our knowledge, we are the first to thoroughly investigate secondary filtering methods for selecting bitexts following a primary, similarity-based mining procedure.

7 Conclusion

We propose a novel method of mining sentence pairs from both comparable and parallel corpora, and demonstrate success on both the Tatoeba gold-standard similarity search task and on mining pseudo-parallel sentences for downstream NMT training. We uncover the problematic nature of setting a similarity score threshold for this task, particularly in the context of document-level min-

port results on the remaining 58

ing. We introduce a heuristic algorithm that filters translations from non-translations by voting on sentence pairs mined in three different ways, which avoids having to laboriously train and re-train NMT systems to tune a similarity score threshold.

References

Waleed Ammar, George Mulcaire, Yulia Tsvetkov, Guillaume Lample, Chris Dyer, and Noah A. Smith. 2016. Massively Multilingual Word Embeddings. *arXiv e-prints, arXiv:1602.01925*.

Sanjeev Arora, Yingyu Liang, and Tengyu Ma. 2017. A Simple but Tough-to-Beat Baseline for Sentence Embeddings. *International Conference on Learning Representations 2017*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Unsupervised Statistical Machine Translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3632–3642, Brussels, Belgium. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019a. An Effective Approach to Unsupervised Machine Translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 194–203, Florence, Italy. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019b. Bilingual Lexicon Induction through Unsupervised Machine Translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5002–5007, Florence, Italy. Association for Computational Linguistics.

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2018b. Unsupervised Neural Machine Translation. In *International Conference on Learning Representations 2018*.

Mikel Artetxe and Holger Schwenk. 2019a. Margin-based parallel corpus mining with multilingual sentence embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3197–3203, Florence, Italy. Association for Computational Linguistics.

Mikel Artetxe and Holger Schwenk. 2019b. Massively Multilingual Sentence Embeddings for Zero-Shot Cross-Lingual Transfer and Beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Adoni Azpeitia, Thierry Etchegoyhen, and Eva Martinez Garcia. 2018. Extracting Parallel Sentences from Comparable Corpora with STACC Variants. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Andoni Azpeitia, Thierry Etchegoyhen, and Eva Martínez Garcia. 2017. Weighted set-theoretic alignment of comparable sentences. In *Proceedings of the 10th Workshop on Building and Using Comparable Corpora*, pages 41–45, Vancouver, Canada. Association for Computational Linguistics.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 conference on machine translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Houda Bouamor and Hassan Sajjad. 2018. H2@BUCC18: Parallel Sentence Extraction from Comparable Corpora Using Multilingual Sentence Embeddings. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Paris, France. European Language Resources Association (ELRA).

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, and Antoine Bordes. 2017. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. *arXiv e-prints*, page arXiv:1705.02364.

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

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

Georgiana Dinu, Angeliki Lazaridou, and Marco Baroni. 2014. Improving Zero-shot Learning by Mitigating the Hubness Problem. *arXiv e-prints*, page arXiv:1412.6568.

Angela Fan, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond English-Centric Multilingual Machine Translation. *arXiv e-prints*, page arXiv:2010.11125.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic BERT Sentence Embedding. *arXiv e-prints*, page arXiv:2007.01852.

Pascale Fung and Percy Cheung. 2004. Multi-level bootstrapping for extracting parallel sentences from a quasi-comparable corpus. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 1051–1057, Geneva, Switzerland. COLING.

Viktor Hangya, Fabienne Braune, Yuliya Kalasouskaya, and Alexander Fraser. 2018. Unsupervised Parallel Sentence Extraction from Comparable Corpora. In *Proceedings of the 15th International Workshop on Spoken Language Translation*, pages 7–13, Bruges, Belgium.

Viktor Hangya and Alexander Fraser. 2019. Unsupervised parallel sentence extraction with parallel segment detection helps machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1224–1234, Florence, Italy. Association for Computational Linguistics.

Vu Cong Duy Hoang, Philipp Koehn, Gholamreza Haffari, and Trevor Cohn. 2018. Iterative back-translation for neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 18–24, Melbourne, Australia. Association for Computational Linguistics.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale Similarity Search with GPUs. *arXiv e-prints*, page arXiv:1702.08734.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Phillip Keung, Julian Salazar, Yichao Lu, and Noah A. Smith. 2020. Unsupervised bitext mining and translation via self-trained contextual embeddings. *Transactions of the Association for Computational Linguistics*, 8:828–841.

Yunsu Kim, Miguel Graça, and Hermann Ney. 2020. When and why is unsupervised neural machine translation useless? In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 35–44, Lisboa, Portugal. European Association for Machine Translation.

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. 2015. Skip-Thought Vectors. *arXiv e-prints*, page arXiv:1506.06726.

Alexandre Klementiev, Ivan Titov, and Binod Bhattarai. 2012. Inducing crosslingual distributed representations of words. In *Proceedings of COLING 2012*, pages 1459–1474, Mumbai, India. The COLING 2012 Organizing Committee.

Garry Kuwanto, Afra Feyza Akyürek, Isidora Chara Tourni, Siyang Li, and Derry Wijaya. 2021. Low-Resource Machine Translation for Low-Resource Languages: Leveraging Comparable Data, Code-Switching and Compute Resources. *arXiv preprint arXiv:2103.13272*.

Ivana Kvapilíková, Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Ondřej Bojar. 2020. Unsupervised multilingual sentence embeddings for parallel corpus mining. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 255–262, Online. Association for Computational Linguistics.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised Machine Translation Using Monolingual Corpora Only. *arXiv preprint arXiv:1711.00043*.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018a. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations*.

Guillaume Lample, Alexis Conneau, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018b. Word translation without parallel data. In *International Conference on Learning Representations*.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018c. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049, Brussels, Belgium. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv e-prints*, page arXiv:1907.11692.

Kelly Marchisio, Kevin Duh, and Philipp Koehn. 2020. When does unsupervised machine translation work? In *Proceedings of the Fifth Conference on Machine Translation*, pages 571–583, Online. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'13, page 3111–3119, Red Hook, NY, USA. Curran Associates Inc.

Dragos Stefan Munteanu, Alexander Fraser, and Daniel Marcu. 2004. Improved machine translation performance via parallel sentence extraction from comparable corpora. In *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics: HLT-NAACL 2004*, pages 265–272, Boston, Massachusetts, USA. Association for Computational Linguistics.

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 44th Annual Meeting of the Association for Computational Linguistics*, pages 81–88, Sydney, Australia. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Nima Pourdamghani, Nada Aldarrab, Marjan Ghazvininejad, Kevin Knight, and Jonathan May. 2019. Translating translationese: A two-step approach to unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3057–3062, Florence, Italy. Association for Computational Linguistics.

Sree Harsha Ramesh and Krishna Prasad Sankaranarayanan. 2018. Neural machine translation for low resource languages using bilingual lexicon induced from comparable corpora. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 112–119, New Orleans, Louisiana, USA. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2020. Making monolingual sentence embeddings multilingual using knowledge distillation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4512–4525, Online. Association for Computational Linguistics.

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

Holger Schwenk. 2018. Filtering and mining parallel data in a joint multilingual space. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 228–234, Melbourne, Australia. Association for Computational Linguistics.

Holger Schwenk, Vishrav Chaudhary, Shuo Sun, Hongyu Gong, and Francisco Guzmán. 2019a. WikiMatrix: Mining 135M Parallel Sentences in 1620 Language Pairs from Wikipedia. *arXiv e-prints*, page arXiv:1907.05791.

Holger Schwenk and Matthijs Douze. 2017. Learning joint multilingual sentence representations with neural machine translation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 157–167, Vancouver, Canada. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, and Armand Joulin. 2019b. CCMatrix: Mining Billions of High-Quality Parallel Sentences on the WEB. *arXiv e-prints*, page arXiv:1911.04944.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked Sequence to Sequence Pre-training for Language Generation. *arXiv e-prints*, page arXiv:1905.02450.

Sandeep Subramanian, Adam Trischler, Yoshua Bengio, and Christopher J Pal. 2018. Learning General Purpose Distributed Sentence Representations via Large Scale Multi-task Learning. *arXiv e-prints*, page arXiv:1804.00079.

Chau Tran, Yuqing Tang, Xian Li, and Jiatao Gu. 2020. Cross-lingual Retrieval for Iterative Self-Supervised Training. *arXiv e-prints*, page arXiv:2006.09526.

Zirui Wang, Jiateng Xie, Ruochen Xu, Yiming Yang, Graham Neubig, and Jaime Carbonell. 2019. Cross-lingual Alignment vs Joint Training: A Comparative Study and A Simple Unified Framework. *arXiv e-prints*, page arXiv:1910.04708.

Lijun Wu, Jinhua Zhu, Di He, Fei Gao, Tao Qin, Jianhuang Lai, and Tie-Yan Liu. 2019. Machine translation with weakly paired documents. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4375–4384, Hong Kong, China. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv e-prints*, page arXiv:1609.08144.

Bing Zhao and Stephan Vogel. 2002. Adaptive Parallel Sentences Mining from Web Bilingual News Collection. In *Proceedings of the 2002 IEEE International Conference on Data Mining*, ICDM '02, page 745, USA. IEEE Computer Society.

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

Pierre Zweigenbaum, Serge Sharoff, and Reinhard Rapp. 2018. Overview of the Third BUCC Shared Task: Spotting Parallel Sentences in Comparable Corpora. In *Workshop on Building and Using Comparable Corpora*, Miyazaki, Japan.

A Appendix

Procedure	**afr**	**amh**	**ang**	**arq**	**arz**	**ast**	**awa**	**aze**	**bel**	**ben**	**ber**	**bos**	**bre**
Raw cosine similarity (_Acc=F1_)	97.4	94	64.2	46.2	78.4	90.6	73.2	96.1	96.2	91.3	10.4	96.2	17.3
Margin scoring, _intersection_, no threshold (_F1_)	98.7	94.2	**73.4**	**57.2**	**84.6**	**94.3**	**83.4**	97.4	97.5	92.4	**14.2**	96.6	**21.5**
Precision	_99.9_	_96.9_	_88.4_	_80.0_	_93.6_	_98.3_	_95.5_	_99.3_	_99.1_	_96.6_	_30.9_	_98.0_	_38.5_
Recall	_97.6_	_91.7_	_62.7_	_44.5_	_77.1_	_90.6_	_74.0_	_95.6_	_95.9_	_88.5_	_9.2_	_95.2_	_14.9_
Margin scoring, _intersection_, threshold = 1.06 (_F1_)	98.2	94.5	72.9	56.0	84.0	94.2	80.5	97.2	97.3	91.8	13.4	96.4	21.3
Precision	_100_	_97.5_	_90.1_	_85.0_	_95.7_	_99.1_	_97.0_	_99.3_	_99.1_	_96.9_	_44.4_	_98.0_	_54.1_
Recall	_96.5_	_91.7_	_61.2_	_41.7_	_74.8_	_89.8_	_68.8_	_95.3_	_95.6_	_87.3_	_7.9_	_94.9_	_13.3_
Margin scoring, _intersection_, threshold = 1.20 (_F1_)	89.5	82.5	59.1	43.6	76.9	92.4	57.2	89.6	94.8	78.6	11.8	90.5	13.4
Precision	_100_	_100_	_96.6_	_97.3_	_98.1_	_99.1_	_98.9_	_99.8_	_99.5_	_99.1_	_90.0_	_99.0_	_92.3_
Recall	_81.0_	_70.2_	_42.5_	_28.1_	_63.3_	_86.6_	_40.3_	_81.4_	_90.5_	_65.1_	_6.3_	_83.3_	_7.2_
Margin scoring, _intersection_, en-xx (_F1_)	98.4	93.2	*	*	*	*	*	96.7	97.6	91.8	*	96.3	*
Precision	_99.6_	_96.8_	*	*	*	*	*	_98.6_	_99.1_	_96.5_	*	_98.2_	*
Recall	_97.3_	_89.9_	*	*	*	*	*	_94.9_	_96.1_	_87.6_	*	_94.4_	*
Margin scoring, _intersection_, xx-en (_F1_)	**99.0**	**95.7**	*	*	*	*	*	**97.6**	97.6	92.0	*	97.3	*
Precision	_99.8_	_98.1_	*	*	*	*	*	_99.0_	_99.1_	_96.3_	*	_98.8_	*
Recall	_98.2_	_93.5_	*	*	*	*	*	_96.3_	_96.1_	_88.0_	*	_95.8_	*
Margin scoring, _intersection_, strict intersection (_F1_)	98.1	93.7	*	*	*	*	*	96.2	96.9	89.8	*	96.0	*
Precision	_100_	_100_	*	*	*	*	*	_99.8_	_99.8_	_99.3_	*	_100_	*
Recall	_96.2_	_88.1_	*	*	*	*	*	_92.8_	_94.2_	_82.0_	*	_92.4_	*
Margin scoring, _intersection_, majority vote (_F1_)	98.9	95.4	*	*	*	*	*	97.5	**97.9**	**93.0**	*	97.1	*
Precision	_99.9_	_98.7_	*	*	*	*	*	_99.3_	_99.6_	_97.9_	*	_98.8_	*
Recall	_97.9_	_92.3_	*	*	*	*	*	_95.9_	_96.2_	_88.6_	*	_95.5_	*

Procedure	**cbk**	**ceb**	**cha**	**cor**	**csb**	**cym**	**dsb**	**dtp**	**epo**	**eus**	**fao**	**fry**	**gla**
Raw cosine similarity (_Acc=F1_)	82.5	70.9	39.8	12.8	56.1	93.6	69.3	13.3	98.4	95.8	90.6	89.9	88.8
Margin scoring, _intersection_, no threshold (_F1_)	**89.5**	79.3	**49.3**	**18.8**	**69.5**	96.2	**80.7**	**18.8**	**99.0**	96.8	**94.9**	93.7	91.9
Precision	_96.7_	_91.1_	_65.9_	_45.2_	_86.5_	_98.9_	_94.7_	_37.5_	_99.7_	_98.4_	_98.0_	_96.9_	_97.1_
Recall	_83.2_	_70.2_	_39.4_	_11.9_	_58.1_	_93.6_	_70.4_	_12.5_	_98.4_	_95.2_	_92.0_	_90.8_	_87.3_
Margin scoring, _intersection_, threshold = 1.06 (_F1_)	87.1	78.5	47.8	16.2	68.0	95.6	79.1	18.5	99.0	96.4	93.4	93.1	91.2
Precision	_97.8_	_93.3_	_75.0_	_64.1_	_90.2_	_99.1_	_95.6_	_56.1_	_99.9_	_98.5_	_98.7_	_97.5_	_97.3_
Recall	_78.6_	_67.7_	_35.0_	_9.3_	_54.5_	_92.3_	_67.4_	_11.1_	_98.2_	_94.4_	_88.5_	_89.0_	_85.8_
Margin scoring, _intersection_, threshold = 1.20 (_F1_)	71.5	67.4	44.3	9.0	54.2	86.0	93.4	15.2	97.9	92.6	84.5	89.5	80.3
Precision	_99.6_	_98.7_	_85.4_	_100_	_95..0_	_99.3_	_99.6_	_87.4_	_99.9_	_99.2_	_99.0_	_99.3_	_98.9_
Recall	_55.7_	_51.2_	_29.9_	_4.7_	_37.9_	_75.8_	_46.6_	_8.3_	_96.0_	_86.8_	_73.7_	_81.5_	_67.6_
Margin scoring, _intersection_, en-xx (_F1_)	*	78.6	*	15.0	*	96.3	76.2	*	98.5	96.4	*	**96.4**	92.6
Precision	*	_90.6_	*	_36.0_	*	_98.9_	_95.0_	*	_99.5_	_98.6_	*	_98.8_	_97.1_
Recall	*	_69.3_	*	_9.5_	*	_93.9_	_63.7_	*	_97.6_	_94.3_	*	_94.2_	_88.4_
Margin scoring, _intersection_, xx-en (_F1_)	*	**86.1**	*	17.3	*	**97.3**	67.3	*	98.9	**97.6**	*	95.6	**93.9**
Precision	*	_94.2_	*	_41.8_	*	_98.9_	_85.5_	*	_99.6_	_98.8_	*	_97.6_	_97.5_
Recall	*	_79.2_	*	_10.9_	*	_95.7_	_55.5_	*	_98.3_	_96.4_	*	_93.6_	_90.6_
Margin scoring, _intersection_, strict intersection (_F1_)	*	77.3	*	13.0	*	95.2	63.0	*	98.5	96.2	*	93.9	89.9
Precision	*	_99.2_	*	_68.6_	*	_100_	_99.1_	*	_100_	_99.5_	*	_98.7_	_99.3_
Recall	*	_63.3_	*	_7.2_	*	_90.8_	_46.1_	*	_97.1_	_93.1_	*	_89.6_	_82.1_
Margin scoring, _intersection_, majority vote (_F1_)	*	81.8	*	18.7	*	96.7	79.4	*	98.8	96.8	*	95.8	93.5
Precision	*	_96.0_	*	_47.9_	*	_99.1_	_97.3_	*	_99.6_	_98.6_	*	_98.8_	_98.0_
Recall	*	_71.3_	*	_11.6_	*	_94.4_	_67.0_	*	_98.1_	_95.2_	*	_93.1_	_89.4_

Procedure	**gle**	**gsw**	**hsb**	**ido**	**ile**	**ina**	**isl**	**jav**	**kab**	**kaz**	**khm**	**kur**	**kzj**
Raw cosine similarity (_Acc=F1_)	95.0	52.1	71.2	90.9	87.1	95.8	96.2	84.4	6.2	90.5	83.2	87.1	14.2
Margin scoring, _intersection_, no threshold (_F1_)	96.6	62.0	81.6	**95.1**	**93.0**	**97.4**	**97.9**	**92.2**	**7.7**	92.6	86.8	92.1	**20.8**
Precision	_98.7_	_85.1_	_94.6_	_98.7_	_98.4_	_99.0_	_99.4_	_98.9_	_19.4_	_96.8_	_93.0_	_98.1_	_41.3_
Recall	_94.6_	_48.7_	_71.8_	_91.7_	_88.1_	_95.9_	_96.4_	_86.3_	_4.8_	_88.7_	_81.3_	_86.8_	_13.9_
Margin scoring, _intersection_, threshold = 1.06 (_F1_)	95.9	60.2	79.7	94.1	91.7	96.9	97.5	91.6	7.3	92.2	86.4	91.4	20.0
Precision	_98.9_	_89.8_	_94.9_	_99.0_	_99.0_	_99.0_	_99.4_	_99.4_	_31.3_	_96.9_	_94.7_	_98.3_	_55.2_
Recall	_93.1_	_45.3_	_68.7_	_89.7_	_85.4_	_95.0_	_95.7_	_84.9_	_4.1_	_87.8_	_79.5_	_85.4_	_12.2_
Margin scoring, _intersection_, threshold = 1.20 (_F1_)	84.7	43.7	67.8	88.5	77.9	94.5	91.0	83.6	5.0	85.7	76.4	82.9	15.1
Precision	_100_	_97.1_	_99.6_	_99.9_	_99.8_	_99.4_	_99.9_	_99.3_	_78.8_	_99.1_	_98.7_	_99.7_	_94.3_
Recall	_73.5_	_28.2_	_51.3_	_79.5_	_63.8_	_90.0_	_83.6_	_72.2_	_2.6_	_75.5_	_62.3_	_71.0_	_8.2_
Margin scoring, _intersection_, en-xx (_F1_)	96.9	58.7	76.6	80.4	76.4	96.3	91.9	*	*	92.6	87.3	92.0	*
Precision	_98.8_	_80.6_	_92.9_	_91.8_	_90.1_	_99.4_	_96.4_	*	*	_97.0_	_93.9_	_97.5_	*

Recall	*95.2*	*46.2*	*65.2*	*71.6*	*66.3*	*93.5*	*87.8*	*	*	*88.7*	*81.6*	*97.1*	*
Margin scoring, *intersection*, xx-en (*F1*)	97.7	59.3	80.0	82.1	78.7	95.8	80.8	*	*	**93.5**	87.5	**95.6**	*
Precision	*99.0*	*83.1*	*93.1*	*95.4*	*93.0*	*98.6*	*93.8*	*	*	*96.8*	*93.5*	*99.2*	*
Recall	*96.4*	*46.2*	*70.2*	*72.0*	*68.2*	*93.2*	*71.0*	*	*	*90.4*	*82.1*	*92.2*	*
Margin scoring, *intersection*, strict intersection (*F1*)	95.6	55.1	74.7	73.2	67.0	94.9	78.2	*	*	91.2	85.6	90.3	*
Precision	*99.6*	*91.2*	*96.1*	*100*	*99.8*	*99.7*	*99.8*	*	*	*99.2*	*98.2*	*99.4*	*
Recall	*92.0*	*39.3*	*61.2*	*57.7*	*50.4*	*90.6*	*64.3*	*	*	*84.3*	*75.9*	*82.7*	*
Margin scoring, *intersection*, majority vote (*F1*)	**97.8**	**62.3**	**81.7**	91.1	88.4	97.1	96.6	*	*	93.1	**87.8**	94.0	*
Precision	*99.3*	*86.4*	*94.8*	*99.5*	*99.3*	*99.1*	*99.3*	*	*	*97.5*	*94.9*	*99.2*	*
Recall	*73.5*	*28.2*	*51.3*	*79.5*	*63.8*	*90.0*	*83.6*	*72.2*	*2.6*	*75.5*	*62.3*	*71.0*	*8.2*

Procedure	**lat**	**lfn**	**mal**	**mhr**	**nds**	**nno**	**nov**	**oci**	**orv**	**pam**	**pms**	**swg**	**swh**
Raw cosine similarity (*Acc=F1*)	82.0	71.2	98.9	19.2	81.2	95.9	78.2	69.9	46.8	13.6	67.0	65.2	88.6
Margin scoring, *intersection*, no threshold (*F1*)	**89.0**	**80.7**	**99.3***	26.3	**89.0**	97.5	**85.4**	**78.7**	**57.4**	**17.9**	**78.9**	**80.4**	93.2
Precision	*96.8*	*93.4*	*99.7*	*46.0*	*96.9*	*99.4*	*93.5*	*90.6*	*78.6*	*34.6*	*92.8*	*95.1*	*97.7*
Recall	*82.4*	*71.0*	*98.8*	*18.4*	*82.2*	*95.7*	*78.6*	*69.6*	*45.3*	*12.1*	*68.6*	*69.6*	*89.0*
Margin scoring, *intersection*, threshold = 1.06 (*F1*)	87.2	79.4	**99.3***	**26.3**	87.6	97.2	83.0	77.7	55.9	17.4	76.3	77.0	92.5
Precision	*97.6*	*94.7*	*99.7*	*59.3*	*98.3*	*99.5*	*94.5*	*93.1*	*83.6*	*50.2*	*94.4*	*96.0*	*98.8*
Recall	*78.7*	*68.4*	*98.8*	*16.9*	*79.1*	*95.1*	*73.9*	*66.6*	*42.0*	*10.5*	*64.0*	*64.3*	*86.9*
Margin scoring, *intersection*, threshold = 1.20 (*F1*)	72.6	68.8	96.4	18.0	74.8	92.1	77.3	65.8	37.0	11.7	63.0	72.3	81.8
Precision	*99.5*	*98.5*	*99.7*	*90.1*	*99.3*	*99.9*	*98.8*	*98.8*	*96.5*	*85.1*	*98.4*	*98.5*	*100*
Recall	*57.2*	*52.9*	*93.3*	*10.0*	*60.0*	*85.5*	*63.4*	*49.3*	*22.9*	*6.3*	*46.3*	*57.1*	*69.2*
Margin scoring, *intersection*, en-xx (*F1*)	83.5	*	98.0	*	86.0	97.3	*	*	*	*	*	*	94.9
Precision	*95.1*	*	*99.5*	*	*97.5*	*99.3*	*	*	*	*	*	*	*98.6*
Recall	*74.4*	*	*96.5*	*	*76.9*	*95.4*	*	*	*	*	*	*	*91.5*
Margin scoring, *intersection*, xx-en (*F1*)	86.1	*	98.2	*	83.8	97.7	*	*	*	*	*	*	95.3
Precision	*95.6*	*	*99.6*	*	*95.2*	*99.4*	*	*	*	*	*	*	*98.1*
Recall	*78.3*	*	*96.9*	*	*74.9*	*96.1*	*	*	*	*	*	*	*92.6*
Margin scoring, *intersection*, strict intersection (*F1*)	81.7	*	97.1	*	80.1	96.6	*	*	*	*	*	*	92.1
Precision	*98.2*	*	*100*	*	*99.3*	*99.8*	*	*	*	*	*	*	*100*
Recall	*69.9*	*	*94.3*	*	*67.2*	*93.7*	*	*	*	*	*	*	*85.4*
Margin scoring, *intersection*, majority vote (*F1*)	88.8	*	99.2	*	88.4	**97.8**	*	*	*	*	*	*	**95.5**
Precision	*97.1*	*	*99.9*	*	*98.3*	*99.6*	*	*	*	*	*	*	*99.4*
Recall	*81.7*	*	*98.5*	*	*80.4*	*96.0*	*	*	*	*	*	*	*91.2*

Procedure	**tam**	**tat**	**tel**	**tgl**	**tuk**	**tzl**	**uig**	**uzb**	**war**	**wuu**	**xho**	**yid**	
Raw cosine similarity (*Acc=F1*)	90.7	87.9	98.3	97.4	80.0	63.0	93.7	86.8	65.3	90.3	91.9	91.0	*
Margin scoring, *intersection*, no threshold (*F1*)	93.0	92.0	**99.1***	98.6	86.8	**71.0**	95.4	91.1	**75.8**	**94.8**	94.2	95.2	*
Precision	*97.5*	*97.4*	*99.6*	*99.7*	*95.8*	*82.3*	*98.3*	*96.8*	*89.5*	*98.8*	*97.7*	*98.7*	*
Recall	*88.9*	*87.1*	*98.7*	*97.6*	*79.3*	*62.5*	*92.7*	*86.0*	*65.7*	*91.1*	*90.8*	*92.0*	*
Margin scoring, *intersection*, threshold = 1.06 (*F1*)	92.8	91.3	**99.1***	98.4	87.3	70.9	95.1	90.7	73.8	94.0	94.2	94.3	*
Precision	*97.8*	*97.9*	*99.6*	*99.8*	*99.4*	*87.3*	*98.3*	*97.1*	*93.5*	*99.0*	*97.7*	*99.1*	*
Recall	*88.3*	*85.5*	*98.7*	*97.1*	*77.8*	*59.6*	*92.2*	*85.0*	*60.9*	*89.4*	*90.8*	*90.0*	*
Margin scoring, *intersection*, threshold = 1.20 (*F1*)	88.9	83.9	97.1	93.3	58.8	56.0	91.5	85.8	57.6	86.6	87.6	87.6	*
Precision	*98.8*	*98.9*	*100*	*100*	*98.8*	*91.3*	*99.6*	*99.4*	*99.8*	*99.5*	*97.4*	*99.5*	*
Recall	*80.8*	*72.8*	*94.4*	*87.5*	*41.9*	*40.4*	*84.6*	*75.5*	*40.5*	*76.7*	*79.6*	*78.2*	*
Margin scoring, *intersection*, en-xx (*F1*)	93.0	89.8	98.5	97.5	85.9	*	94.8	93.5	*	*	92.9	93.6	*
Precision	*98.2*	*95.4*	*99.1*	*99.2*	*95.8*	*	*98.2*	*98.7*	*	*	*98.4*	*98.2*	*
Recall	*88.3*	*84.8*	*97.9*	*95.8*	*77.8*	*	*91.6*	*88.8*	*	*	*88.0*	*89.5*	*
Margin scoring, *intersection*, xx-en (*F1*)	**93.7**	**93.9**	97.6	**99.4**	**97.0**	*	**95.5**	**95.2**	*	*	**97.2**	**97.2**	*
Precision	*97.5*	*97.7*	*99.1*	*99.9*	*99.5*	*	*98.6*	*97.8*	*	*	*97.9*	*98.8*	*
Recall	*90.2*	*90.4*	*96.2*	*98.9*	*94.6*	*	*92.5*	*92.8*	*	*	*96.5*	*95.8*	*
Margin scoring, *intersection*, strict intersection (*F1*)	92.0	89.9	97.4	97.6	79.9	*	93.7	91.2	*	*	91.3	92.7	*
Precision	*99.2*	*99.5*	*99.6*	*100*	*100*	*	*99.7*	*100*	*	*	*98.4*	*99.6*	*
Recall	*85.7*	*81.9*	*95.3*	*95.3*	*66.5*	*	*88.5*	*83.9*	*	*	*85.2*	*86.7*	*
Margin scoring, *intersection*, majority vote (*F1*)	93.7	92.5	**99.1***	98.8	94.0	*	95.4	93.6	*	*	95.7	95.9	*
Precision	*98.6*	*97.9*	*99.6*	*100*	*100*	*	*98.7*	*99.2*	*	*	*98.5*	*99.1*	*
Recall	*89.3*	*87.6*	*98.7*	*97.6*	*88.7*	*	*92.3*	*88.6*	*	*	*93.0*	*92.8*	*

Table 4: Tatoeba test set results for a subset of low-resource, English-aligned language pairs, broken down by the mining method used. These language pairs are ones *without* parallel data for the multilingual distillation process described in Reimers and Gurevych (2020) (cf. Table 10 in that paper). Note that LaBSE has training data for most of these languages. Descriptions of the various mining methods are found in Section 4.