Data Filtering using Cross-Lingual Word Embeddings

Christian Herold **Jan Rosendahl** **Joris Vanvinckenroye** **Hermann Ney**
Human Language Technology and Pattern Recognition Group
Computer Science Department
RWTH Aachen University
D-52056 Aachen, Germany
{surname}@i6.informatik.rwth-aachen.de

Abstract

Data filtering for machine translation (MT) describes the task of selecting a subset of a given, possibly noisy corpus with the aim to maximize the performance of an MT system trained on this selected data. Over the years, many different filtering approaches have been proposed. However, varying task definitions and data conditions make it difficult to draw a meaningful comparison. In the present work, we aim for a more systematic approach to the task at hand. First, we analyze the performance of language identification, a tool commonly used for data filtering in the MT community and identify specific weaknesses. Based on our findings, we then propose several novel methods for data filtering, based on cross-lingual word embeddings. We compare our approaches to one of the winning methods from the WMT 2018 shared task on parallel corpus filtering on three real-life, high resource MT tasks. We find that said method, which was performing very strong in the WMT shared task, does not perform well within our more realistic task conditions. While we find that our approaches come out at the top on all three tasks, different variants perform best on different tasks. Further experiments on the WMT 2020 shared task for parallel corpus filtering show that our methods achieve comparable results to the strongest submissions of this campaign.

1 Introduction

In recent years, neural machine translation (NMT) systems have greatly improved the quality of automatically generated translations, some argue even to the point of human parity (Hassan et al., 2018). While there most definitely have been advancements in designing the NMT system architectures (Bahdanau et al., 2015; Vaswani et al., 2017), arguably the best (and easiest) way to improve an NMT system is to use more training data. With an ever increasing amount of parallel data for NMT training, which often comes from web-crawling[1] and is quite 'noisy', the task of data filtering becomes increasingly important (Khayrallah and Koehn, 2018).

Data filtering in the context of machine translation (MT) describes a collection of approaches which select a subset of a given, possibly noisy corpus with the aim to maximize the performance of an MT system trained on this data. There exist very simple approaches, the most prominent being based on language identification tools, to detect certain types of noise, e.g. sentences that are from a wrong language. However, other types of noise are much harder to detect, for example when both source and target sentence are well formulated and in the correct language but are not translations of one another.

In some formulations of the data filtering task, for example in the WMT shared task for parallel corpus filtering (Koehn et al., 2018, 2019, 2020), the assumption is that there already exists a large amount of 'clean' data which can be used to detect bad training samples in a separated 'noisy' corpus. However, such an assumption does typically not hold true in real-life scenarios. Therefore, in this work, we make no such distinction between 'known-to-be-clean' and 'noisy' data. We present novel approaches that use all the available data to filter that very same data in order to improve translation performance.

In the proposed methods, we use the structure of cross-lingual word embeddings to compare the words in a given source-target sentence pair to determine if the pair is of 'good' quality. This is done in a variety of ways, including nearest neighbor search in the embedding space and an explicit calculation of alignment scores. All proposed methods are specifically designed to detect the types of noise which cannot be detected by language identification tools. Furthermore, we design our approaches

[1] http://opus.nlpl.eu

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 162–172
June 6–11, 2021.

to not rely on the quality of the sentence pair alignments between the source and the target side of the data, since this information might be highly unreliable in a 'noisy' corpus.

The main contributions of this paper are summarized below:

- We perform a systematic analysis of 'noise-types' for a commonly used MT task and identify specific weaknesses of the commonly used filtering by language identification.
- Building on our findings, we propose novel data filtering approaches using cross-lingual word embeddings.
- We compare our approaches to other strong filtering systems from the literature on three real-life, high resource MT tasks and the WMT 2020 task on parallel corpus filtering.

2 Related Work

Recently, a number of shared tasks for data filtering have been held, giving a good overview of current state-of-the-art methods. Best known is the WMT shared task for parallel corpus filtering, which was held in 2018 (Koehn et al., 2018), 2019 (Koehn et al., 2019) and 2020 (Koehn et al., 2020) respectively. In these tasks, the participants are asked to provide scores for every sentence pair in a noisy corpus. Afterwards, a fixed amount of sentence pairs is selected according to that score.

The best performing submissions from past years use language identification tools as the first part of their setup (Junczys-Dowmunt, 2018; Chaudhary et al., 2019; Lu et al., 2020), removing sentence pairs where the language of either source or target sentence does not match the expectation. Rossenbach et al. (2018) and Junczys-Dowmunt (2018) use a combination of language model and translation model scores to sort the sentence pairs by quality. Chaudhary et al. (2019) use the cosine distance between cross-lingual sentence embeddings of source and target sentence as score. Wang et al. (2017) estimate the quality of a sentence pair using the euclidean distance between each sentence vector and two vectors representing in-domain and out-domain data. Hangya and Fraser (2018) score the similarity between source and target sentence by averaging the word-pair similarity, which is calculated from cross-lingual word embeddings.

Since the above mentioned methods are evaluated on different tasks with very different data conditions, one can not easily make a statement about which approach works best. However, all approaches have in common that they use 'known-to-be-clean' parallel data in order to train the models of their filtering pipeline.

Creating cross-lingual word embeddings from parallel and/or monolingual data is an active field of research (Ruder et al., 2019). In addition to capturing semantic relationships within each language, these representations should be aligned in such a way that the embeddings of the same word in different languages are close together in the embedding space. The standard approach for creating such embeddings is to first train embeddings for each language pair separately (Mikolov et al., 2013; Pennington et al., 2014) and then projecting them into the same vector space (Conneau et al., 2017; Artetxe et al., 2018), which is possible with or without the help of parallel data.

Word alignments between a source and a target sentence were an integral part in count-based statistical machine translation systems (Brown et al., 1993; Koehn et al., 2007) and it has been shown that they can be used to help certain aspects of NMT systems as well (Alkhouli et al., 2018). For a long time, IBM-model-based frameworks like GIZA++ (Och and Ney, 2003) or fastalign (Dyer et al., 2013) produced the best word alignments. However, recently Sabet et al. (2020) report equally good results by using a word similarity matrix calculated from cross-lingual word embeddings.

3 Detecting Different Types of Noise

Applying language identification (language ID) is a well established first step in most high performing data filtering approaches. During this step, all sentence pairs for which either the source or target sentence is not mapped to the correct language are discarded. It can be argued that this step does not only remove sentence pairs in the wrong language, but also that language-agnostic noise, e.g. sequences of numbers, is almost completely removed.

In order to evaluate the effectiveness of the filtering by language ID approach, we decide to test the method on the popular De→En data filtering task. By manually checking the noisy corpus (see Section 5.1 for details) we find different types of 'noise patterns'. For each of these 'noise patterns', we create a synthetic corpus (50k lines each), only consisting of sentence pairs with this specific noise.

We find/create the following 'noise patterns':

trg to src: The source and target side of a valid sentence pair are swapped.

trg to trg: Both source and target side contain different sentences from the target language.

src to src: Both source and target side contain different sentences from the source language.

src to other: The sentence on the source side is from the correct language. The sentence on the target side is a random sentence from a third language.

other to trg: The sentence on the source side is a random sentence from a third language. The sentence on the target side is from the correct language.

other to other: Both sentences on the source and target side are random sentences from a third language.

sentence misalign: Both sentences on the source and target side are from the correct language, but they are not translations of one another.

overtranslation: Both sentences on the source and target side are from the correct language and translations of one another, but parts of the source sentence are missing.

undertranslation: Both sentences on the source and target side are from the correct language and translations of one another, but parts of the target sentence are missing.

random digits: The source and target sentences each consist of random number sequences.

For the unrelated third language (other) we choose French.

Next, we use the `langid.py` toolkit (Lui and Baldwin, 2012) to filter each of these synthetic corpora and check which percentage of noise (ideally 100.0%) gets removed. The results are shown in Table 1.

We find that the language identification filtering approach does an outstanding job in detecting noise that comes from wrong language alignment. Furthermore it also removes basically all of the random noise, represented by the **random digits** corpus. However, we also see where this approach fails: it can not detect noise resulting from a semantic mismatch between source and target sentence.

Noise Type	Percentage removed
trg to src	100.0%
trg to trg	100.0%
src to src	100.0%
src to other	99.5%
other to trg	99.8%
other to other	100.0%
sentence misalign	0.0%
overtranslation	7.8%
undertranslation	6.7%
random digits	100.0%

Table 1: Removal rate of different noise types by the language identification filtering method.

Two conclusions can be drawn from this experiment: First, the filtering methods applied after language identification filtering can be language-agnostic, since all types of noise which originate from wrong languages can be detected by language identification very reliably. Second, downstream filtering methods should focus on the alignment between source and target sentence, since this is where language identification filtering predictably fails.

4 Data Filtering Methods

Intuitively a bilingual sentence pair is appropriate for training if a) both the source and the target sentence belong to the corresponding languages and b) they are translations of each other. We rely on established language identification methods (see Section 5.1) to verify the first condition. Following state of the art filtering systems (Junczys-Dowmunt, 2018; Chaudhary et al., 2019) we predict the language for source and target sentence and keep the sentence only if both match the requirements of the task. To check whether the sentences of a training pair (f_1^J, e_1^I) are indeed translations of each other we propose several approaches based on cross-lingual word embeddings. For the details of how the cross-lingual word embeddings are constructed we refer to Section 5.1. Here we assume that we are given a cross-lingual word embedding $E : V_{\text{src}} \cup V_{\text{trg}} \rightarrow \mathbb{R}^{d_{\text{embd}}}$ that maps each word from the source vocabulary V_{src} or the target vocabulary V_{trg} to a joint space $\mathbb{R}^{d_{\text{embd}}}$ with a similarity measure ρ. For convenience we use $E_w := E(w)$. In practice all embedding vectors are length normalized,

i.e. $||E_w|| = 1$.

4.1 Nearest Neighbour based

Many works investigate distances in the embedding space as an indicator of relatedness between words of the same language. However we are interested in the relation between the words of the source sentence and the target sentence. Specifically, we want to know whether the two sentences are translations of each other. We assume a source word f is explained by a word e in the target sentence, if $E(f)$ is one of the k nearest neighbours of $E(e)$ i.e. if:

$$\rho\left(E_f, E_e\right) \geq \text{max-k}\left\{\rho\left(E_{\hat{f}}, E_e\right) \;\middle|\; \hat{f} \in V_{\text{src}}\right\}$$

where max-k yields the k-th biggest value. Note that we only consider the source nearest neighbourhood around e. To score a sentence pair (f_1^J, e_1^I) we calculate:

$$\text{explain}(f_1^J \mid e_1^I) := \left|\{f_j \mid \exists e_i : e_i \text{ explains } f_j\}\right|.$$

For data filtering we consider different variants of combining the forward and backward score:

Accumulated Explanation Score:

$$\frac{\text{explain}(e_1^I|f_1^J) + \text{explain}(f_1^J|e_1^I)}{I+J}$$

Explanation Disagreement Score: Note that being nearest neighbours in a multilingual embedding space is not a symmetric relation. We compute the agreement of the forward and the backward score:

$$\left|\frac{\text{explain}(e_1^I|f_1^J)}{I} - \frac{\text{explain}(f_1^J|e_1^I)}{J}\right|$$

Explanation Disagreement + Pre-Filtering: A sentence pair is removed if its score for either direction falls below a threshold γ:

$$\min\{\text{explain}(e_1^I|f_1^J), \text{explain}(f_1^J|e_1^I)\} < \gamma$$

the remaining sentences are scored via explanation disagreement score

As similarity measure ρ we choose cross-domain-similarity-scaling (CSLS) (Conneau et al., 2017):

$$\begin{aligned}\text{CSLS}(E_f, E_e) = 2\cos(E_f, E_e) \\ - \frac{1}{n}\sum_{f' \in N_f(e,n)} \cos(E_{f'}, E_e) \\ - \frac{1}{n}\sum_{e' \in N_e(f,n)} \cos(E_f, E_{e'})\end{aligned}$$

where $N_f(e,n)$ is the neighborhood of size n across the word e in the space of the language of f.

4.2 Source ↔ Target Embedding Similarity

The methods described so far are based on the neighbourhood of size k around each word to create a source→target and a distinct target→source alignment. Alternatively we consider the source↔target similarity matrix:

$$A_{i,j} := A(f_1^J, e_1^I)_{i,j} := E_{e_i}^\intercal E_{f_j}$$

where each entry expresses the similarity of a word pair from the source and target sentence. Note that due to the construction of the cross-lingual word embeddings (see Section 5.1) all word embeddings are normalized. This means that the scalar product above is equivalent to the cosine similarity. We consider several options to compute a source↔target similarity score:

Argmax Agreement: Considers alignment points where src→trg and the trg→src argmax are the same:

$$\begin{aligned}M := \{(i,j) \mid\; & i = \text{argmax}_{\hat{i}}\, A_{\hat{i},j} \text{ and} \\ & j = \text{argmax}_{\hat{j}}\, A_{i,\hat{j}}\}\end{aligned}$$

and sums up the corresponding weights

$$\frac{1}{\max\{I,J\}}\sum_{(i,j)\in M} A_{i,j}.$$

Maximum Matching (Score): On the complete bipartite graph induced from the similarity matrix A, i.e. the bipartite graph with vertices $V := f_1^J \dot\cup e_1^I$ and edge weight function $f := I \times J \to \mathbb{R} : (i,j) \mapsto A_{i,j}$. We use the total weight of the maximum-weight matching divided by $\max\{I,J\}$ as a score.

Maximum Matching (Count): We construct a maximum-weight matching on the bipartite graph with vertices V and edge weights f however we prune the edges if the corresponding word similarity is below a threshold t, keeping only the edges

$$\mathcal{E} := \{(i,j) \in I \times J \mid A_{i,j} \geq t\}.$$

The number of matching points divided by $\max\{I,J\}$ is used as score for the sentence pair.

Average similarity: The score is defined as the average over the similarity matrix, i.e.

$$\frac{1}{I \cdot J}\sum_{I\times J} A_{i,j}.$$

We would like to point out that parallel to the present work, Sabet et al. (2020) also introduced the first two of the four methods. Since they aim to extract an explicit alignment between source and target they do not construct a score for a sentence pair and do not consider the use in a data filtering task.

Since we are interested in aligning the source and target sentence to obtain a score for data filtering we also use the IBM4 alignment scores provided from GIZA++ (Och and Ney, 2003) for filtering as a comparison.

4.3 Data Selection and Score Transformation

We consider different ways to select training data given a noisy corpus where each sentence pair (f_1^J, e_1^I) has an associated score $s(f_1^J, e_1^I) \in \mathbb{R}$:

(1) **Top X%:** Selecting the $X\%$ sentence pairs with the best score s.

(2) **Top X% Transformed:** Selecting the $X\%$ sentence pairs with the best transformed score:

$$s_{\text{t}}(f_1^J, e_1^I) = \left| s(f_1^J, e_1^I) - \sum_{(F,E)\in\text{dev}} \frac{s(F,E)}{|\text{dev}|} \right|.$$

(3) **Dev set distribution**: We score the dev set using s. Empirically this yields a Gaussian distribution where some scores are more frequent than others. We fit a Gaussian distribution and select a lower and an upper threshold such that 95% of the dev set distribution are selected. All sentence pairs from the training corpus whose score falls between the two thresholds are selected.

We introduce Variants (2) and (3) since we observe that often the best scored sentence pairs exhibit a pattern that is easy to learn but not representative for translation at all, e.g. sentence pairs that are dominated by long dates on both sides, etc. In particular the sentence pairs from the dev set are our best approximation of what 'valid training data' should look like. A sentence pair that scores significantly better than the dev set is just as suspicious than one that scores significantly worse.

	# trg tokens	# lines
De→En	743M	37M
En→Tr	332M	50M
En→Cs	668M	57M

Table 2: Training data size of the three translation tasks.

5 Experiments

5.1 Experimental Setup

We evaluate the performance of the data filtering systems on three high-resource tasks, namely German→English, English→Turkish and English→Czech. The De→En training data consists of the corpora Commoncrawl, Europarl, Rapid and ParaCrawl from the WMT 2019 news translation task[2]. We use the czeng 1.7 corpus[3] from the WMT 2018 news translation task for En→Cs. For En→Tr we test our systems on a real world corpus with a focus on the entertainment domain provided by a company. We select these three data conditions because they provide high resource data that originates from very different sources and, hence, should express rather different data biases and noise patterns. We choose to test the proposed methods in two settings of the WMT news translation task and not in the conditions defined by the WMT parallel corpus filtering task because we experienced in the past, that performance gains from data filtering on the very noisy corpora of the data filtering task do not carry over to the news translation task. For the corpus data statistics, please refer to Table 2.

Following state of the art filtering systems (Junczys-Dowmunt, 2018), we use the `langid.py` toolkit (Lui and Baldwin, 2012) as the first step in our filtering pipeline by removing source and target sentences where at least one side is not classified to be the correct language. In order to obtain cross-lingual word embeddings we follow the method proposed by Artetxe et al. (2018). In particular we first train GloVe Word Embeddings (Pennington et al., 2014) with a fixed vector size of 300 on the respective monolingual corpora after applying `langid.py`. From these we select the embeddings of the 200k most common words in each language. They form the base

[2] http://www.statmt.org/wmt19/translation-task.html

[3] https://ufal.mff.cuni.cz/czeng/czeng17

Filter Method	Data Selection Method	Training Data #trg tokens	#sent. pairs	dev BLEU	test BLEU
De→En Accu. Expl. Scores	Top 50	374M	18M	33.1	34.1
	Top 50 Transformed	360M	14M	34.2	35.6
	Dev Set Distribution	612M	26M	34.0	35.4
De→En Matching (score) (BPE)	Top 50	366M	16M	33.7	34.9
	Top 50 Transformed	366M	17M	33.7	34.8
	Dev Set Distribution	559M	25M	33.8	34.8
En→Tr Matching (score) (BPE)	Top 50	164M	21M	14.8	14.7
	Top 50 Transformed	162M	19M	17.6	20.5
	Dev Set Distribution	273M	35M	15.0	15.3

Table 3: Comparison of different data selection methods and the resulting translation performance. As test set we use: newstest2017 (De→En) and newstest2018 (En→Tr). BLEU and TER are reported in percentage.

for the cross-lingual word embeddings, also with a fixed vector size of 300, which are created using the VecMap toolkit (Artetxe et al., 2018). All of the cross-lingual word embeddings are normalized. To be consistent with our filtering task definition, we do not use an initial seed dictionary to train the cross-lingual word embeddings. For nearest neighbor search we set k equal to five and use cross-domain-similarity-scaling (Conneau et al., 2017) as the distance metric when computing the sentence pair scores. The threshold γ is set to 0.1 for the pre-filtering step of the explanation disagreement score. We compare our methods to another strong filtering method, that scores all sentence pairs by averaging the log probabilities of two language models (LMs) and two translation models (TMs) (Rossenbach et al., 2018). Each method creates a subset from the corpus, which is used to train a base transformer model (Vaswani et al., 2017) with six encoder and decoder layers implemented using the RETURNN toolkit (Zeyer et al., 2018). Machine translation performance is measured using BLEU scores (Papineni et al., 2002) and TER scores (Snover et al., 2006) using the MtEval tool from the Moses toolkit (Koehn et al., 2007). The development sets we use are newstest2015 for De→En, newstest2016 for En→Cs and a concatenation of development sets from multiple domains for En→Tr.

5.2 Experimental Results

In a first step we investigate the data selection strategies described in Section 4.3. We consider two variants that select a fixed amount of training data plus an additional variant where the amount of selected data is dynamically determined in an automatic way. Note that the amount of data is measured in target positions on the raw text. However since for each MT training we train and apply a new subword splitting, the amount of target subwords in training varies slightly (we observe changes of less than 5%). Results for the different data selection schemes can be found in Table 3. We observe that transforming the scores can be extremely helpful to get good filtering performance. Selecting based on a dev set distribution yields similar strong results but is not as stable. We select data corresponding to the Top 50% of target tokens according to the transformed score except for the GIZA method where we use the non-transformed score because the transformation resulted in unreliable scores due to precision issues.

German→English

First we consider the De→En WMT 2019 news translation task. Note that most of the training data comes from the news translation task ParaCrawl corpus which is smaller and of better quality than the ParaCrawl corpus used in the WMT 2018 parallel corpus filtering task. We start with all the training data and apply language ID as initial filtering, i.e. if either the source or the target sentence of a training pair is not classified with the correct language we drop the sentence pair. The result of this filtering can be seen in Table 4, Line 2. All further filtering methods are trained and applied on this pre-filtered corpus.

It is interesting to point out that the LM & TM comparison system does not even beat the language identification baseline. For LM & TM we employ a slight simplification of a system that improved

Filter Method	Training Data Ratio	dev (newstest2015) BLEU	dev (newstest2015) TER	newstest2017 BLEU	newstest2017 TER
None (baseline)	1.00	33.5	53.3	34.6	52.7
Language ID	0.89	33.7	53.0	35.0	52.0
LM & TM (Rossenbach et al., 2018)	0.49	33.6	53.7	34.5	52.9
Accum. Expl. Scores	0.48	**34.2**	**52.6**	**35.6**	**51.6**
Expl. Disagreement Score	0.50	33.5	53.3	35.1	52.1
+ pre-filtering	0.49	33.6	52.8	35.2	51.8
Argmax Agreement	0.49	34.1	52.9	35.2	52.0
Maximum Matching (score)	0.49	33.9	53.2	35.1	52.2
+ BPE level	0.49	33.7	53.2	34.8	52.7
Maximum Matching (count)	0.49	33.8	53.0	34.9	52.2
+ BPE level	0.50	33.5	53.8	34.9	52.4
Average similarity	0.49	33.9	52.8	35.2	52.2
GIZA	0.50	32.6	53.9	33.5	53.1

Table 4: Comparing filtering methods on De→En WMT 2019 news translation task. All filtering methods are trained and applied on a corpus that is pre-filtered with language identification (Line 2). Amount of training data is given as ratio of the original corpus. BLEU and TER are reported in percentage.

translation performance by more than 8.0 BLEU and performed among the best on the WMT 2018 data filtering task (Rossenbach et al., 2018). There are two crucial differences to consider: (1) We train the filtering system on the same data that it needs to filter afterwards. This means the filtering pipeline might learn typical patterns from the data that are not actually relevant for translation, like copying the input sentence. (2) The ParaCrawl corpus used here is a newer version of better quality and we add the established training data for the WMT news translation task so that the complete training data is generally of significantly higher quality. Note that the ParaCrawl corpus still provides 80% of the training data and the benefits of doing data filtering diminish quite clearly. We conclude that it is highly important how exactly the data filtering task is phrased.

The best performance on the De→En WMT task is achieved by the 'Accumulated Explanation Scores' method which yields an average improvement of 0.5% with respect to both BLEU and TER across the dev and test set. All other methods except for 'GIZA' are on par with the language identification baseline, however they achieve a significant reduction of the training data. We experiment with a variant of the Maximum Matching method for scores and counts that is built on top of cross-lingual subword embeddings without any effect in translation performance.

English→Turkish

The behaviour of the filtering systems is quite different for the company data set of the En→Tr task. We report results on three openly available test sets from different domains. In this scenario language identification helps quite clearly on two out of three data sets while LM & TM data filtering significantly reduces the translation performance.

With our methods, we observe very clear improvements on the TED test set as well as newstest2018. The Explanation Disagreement Score with pre-filtering gains an average of 0.7 $\text{BLEU}^{[\%]}$ over the language identification filtering. If we apply Maximum Matching filtering on BPE level we even observe improvements of 2.2 and 5.1 $\text{BLEU}^{[\%]}$ on TED and newstest2018, however we lose 0.9 $\text{BLEU}^{[\%]}$ and 0.7 $\text{TER}^{[\%]}$ on the OpenSubtitles test set. In practice, this minor degradation is out weighted by the significantly stronger performance on the other domains, proofing the usefulness of data filtering in this scenario.

The scores based on GIZA alignments result in a very poor performance on all domains except subtitles. By analyzing the selected data, we find that the 'GIZA' method selects on average shorter sequences than other methods which is detrimental for the news and talks domain but not so much for subtitles.

Filter Method	Training Data Ratio	TED BLEU	TED TER	newstest2018 BLEU	newstest2018 TER	Opensubtitles BLEU	Opensubtitles TER
None (baseline)	1.00	14.8	77.2	16.2	73.5	**20.9**	**71.4**
Language ID	0.84	16.1	76.1	19.3	70.6	20.4	74.6
LM & TM (Rossenbach et al., 2018)	0.50	10.4	82.1	9.2	83.5	19.4	76.1
Accum. Expl. Scores	0.48	15.6	75.6	22.1	66.6	19.0	76.1
Expl. Disagreement Score	0.48	16.8	73.9	21.8	67.6	19.3	76.4
+ pre-filtering	0.48	17.3	73.1	22.7	65.9	20.0	75.6
Maximum Matching (score)	0.48	16.3	74.8	20.5	68.7	19.5	76.0
+ BPE level	0.52	**18.2**	**71.2**	**24.4**	**64.1**	19.5	75.3
Maximum Matching (count)	0.48	16.1	74.0	19.4	69.5	18.7	77.8
+ BPE level	0.49	14.7	75.3	15.7	73.7	18.6	77.7
GIZA	0.50	7.4	82.9	5.9	84.6	18.7	76.7

Table 5: Comparing filtering methods on En→Tr WMT 2019 news translation task. All filtering methods are trained and applied on a corpus that is pre-filtered with language identification (Line 2). Amount of training data is given as ratio of the original corpus. BLEU and TER are reported in percentage.

Filter Method	Training Data Ratio	dev (newstest2016) BLEU	dev (newstest2016) TER	newstest2019 BLEU	newstest2019 TER
None (baseline)	1.00	25.7	63.3	22.3	66.8
Language ID	0.90	**25.9**	63.4	22.7	66.7
LM & TM (Rossenbach et al., 2018)	0.50	24.7	64.7	21.3	68.4
Accum. Expl. Scores	0.49	25.7	63.4	**23.0**	**66.3**
Expl. Disagreement Score	0.49	25.7	**63.2**	22.3	66.9
Maximum Matching (score)	0.48	25.6	63.5	22.3	66.9
+ BPE level	0.48	25.5	63.7	22.5	66.7
GIZA	0.50	24.8	64.0	20.6	68.4

Table 6: Comparing filtering methods on En→Cs WMT 2019 news translation task. All filtering methods are trained and applied on a corpus that is pre-filtered with language identification (Line 2). Amount of training data is given as ratio of the original corpus. BLEU and TER are reported in percentage.

English→Czech

For the En→Cs task we observe no significant improvement with any of the methods over even the training on the full training data, even though 10% of the data is removed by simple language identification filtering. Here we observe that LM & TM filtering becomes actively hurtful to the translation performance while the methods proposed in this paper reduce the training data by a factor of two without losing in translation performance. The proposed filtering methods all provide very similar filtering performances except for the scores based on GIZA alignments which decrease the system performance by more than one BLEU[%].

5.3 WMT 2020: Khmer→English

As an additional experiment, we also test our methods on the WMT 2020 shared task for parallel corpus filtering in the Khmer→English setting. Although some conditions of this task are quite artificial as discussed before, it provides the opportunity to compare different filtering approaches in the same framework.

The task consists of selecting sentence pairs that amount to 5.0M English words from a noisy parallel corpus with a total of 58.3M English words. The quality of the selected data is evaluated by training an NMT system (Ott et al., 2019) on this data and evaluating the system on unseen test sets labeled 'devt' and 'test' (Koehn et al., 2020). For

Filter Method	BLEU	
	devt	test
LASER (2019 winner)	7.1	8.4
Alibaba system (2020 winner)	8.9	**11.0**
Maximum Matching (score)	8.2	10.3
Accum. Expl. Scores	**9.0**	10.9

Table 7: Final performance of NMT systems trained on the selected data (5.0M English tokens) of the WMT 2020 Khmer→English data filtering task.

training the filtering system, around 123k clean parallel sentences are given as well as large monolingual corpora for both languages (14M sentences for Khmer and 1.9B sentences for English).

As a first step, we apply filtering using language identification as described in Section 3 to sort out sentence pairs with wrong language on source and/or target side. Based on the previous findings, we use our 'Accum. Expl. Scores' and our 'Maximum Matching (score)' methods on the BPE level for scoring. Since the parallel data is very small and of questionable quality we only use the monolingual data for the training of our word embeddings. We use all the available monolingual Khmer data while subsampling 14M English sentences. We use the polyglot tokenizer[4] on the Khmer data and train BPE models for Khmer and English separately. The performance of the resulting NMT system is shown in Table 7.

Also shown in the table are the results of the LASER filtering system (Chaudhary et al., 2019) which won the WMT 2019 data-filtering evaluation as well of the Alibaba filtering system (Lu et al., 2020) which won the WMT 2020 data-filtering evaluation for Khmer→English. We find that our filtering methods performs strongly on this task as well, with our 'Accum. Expl. Scores' method performing on par with the strongest submission of the latest WMT campaign while not relying on any parallel data.

6 Conclusion

In this work we focus on data filtering for machine translation. We define this task as the selection of a subset of a given, possibly noisy corpus, without the help of additional large-scale 'clean' corpora. In order to develop a helpful filtering method, we first analyze the commonly used 'filtering by language identification' approach by applying it to synthetically generated noisy data. We find that while 'filtering by language identification' does an outstanding job in detecting noise that comes from wrong language alignment, it fails to detect noise resulting from a semantic mismatch between source and target sentence.

Building on these findings, we develop several approaches - based on cross-lingual word embeddings - specifically targeting the word alignments between source and target sentence. Furthermore, we conduct a systematic comparison on data selection methods in an effort to uncouple the scoring and selection parts of any data filtering pipeline. We compare our approaches to one of the winning methods from the WMT 2018 shared task on parallel corpus filtering on three real-life, high resource tasks as well as on the recent WMT 2020 shared task on parallel corpus filtering. We find that the existing approach does not perform well in our more realistic scenario, leading to a degradation in performance in most cases. Our methods result in improvements over the baseline on all three three tasks. However, different variants of our methods perform best on different tasks and we can not identify a single best approach.

Finally, we compare our methods to state-of-the-art data-filtering systems on the WMT 2020 shared task on parallel corpus filtering. Here, our proposed approaches yield comparable results to aforementioned state-of-the-art methods while not relying on any parallel training data.

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No 694537, project "SEQCLAS"). The work reflects only the authors' views and the European Research Council Executive Agency (ERCEA) is not responsible for any use that may be made of the information it contains.

References

Tamer Alkhouli, Gabriel Bretschner, and Hermann Ney. 2018. On the alignment problem in multi-head attention-based neural machine translation. In *Pro-*

[4] https://github.com/aboSamoor/polyglot

ceedings of the Third Conference on Machine Translation: Research Papers, pages 177–185.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018. A robust self-learning method for fully unsupervised cross-lingual mappings of word embeddings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 789–798. Association for Computational Linguistics.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

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

Vishrav Chaudhary, Yuqing Tang, Francisco Guzmán, Holger Schwenk, and Philipp Koehn. 2019. Low-resource corpus filtering using multilingual sentence embeddings. In *Proceedings of the Fourth Conference on Machine Translation, WMT 2019, Florence, Italy, August 1-2, 2019 - Volume 3: Shared Task Papers, Day 2*, pages 261–266.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *CoRR*, abs/1710.04087.

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

Viktor Hangya and Alexander M. Fraser. 2018. An unsupervised system for parallel corpus filtering. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 882–887.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. 2018. Achieving human parity on automatic chinese to english news translation. *CoRR*, abs/1803.05567.

Marcin Junczys-Dowmunt. 2018. Dual conditional cross-entropy filtering of noisy parallel corpora. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 888–895.

Huda Khayrallah and Philipp Koehn. 2018. On the impact of various types of noise on neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation, NMT@ACL 2018, Melbourne, Australia, July 20, 2018*, pages 74–83. Association for Computational Linguistics.

Philipp Koehn, Vishrav Chaudhary, Ahmed El-Kishky, Naman Goyal, Peng-Jen Chen, and Francisco Guzmán. 2020. Findings of the WMT 2020 shared task on parallel corpus filtering and alignment. In *Proceedings of the Fifth Conference on Machine Translation*, pages 726–742.

Philipp Koehn, Francisco Guzmán, Vishrav Chaudhary, and Juan Pino. 2019. Findings of the WMT 2019 shared task on parallel corpus filtering for low-resource conditions. In *Proceedings of the Fourth Conference on Machine Translation, WMT 2019, Florence, Italy, August 1-2, 2019 - Volume 3: Shared Task Papers, Day 2*, pages 54–72.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *ACL 2007, Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics, June 23-30, 2007, Prague, Czech Republic*. The Association for Computational Linguistics.

Philipp Koehn, Huda Khayrallah, Kenneth Heafield, and Mikel L Forcada. 2018. Findings of the WMT 2018 shared task on parallel corpus filtering. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 726–739.

Jun Lu, Xin Ge, Yangbin Shi, and Yuqi Zhang. 2020. Alibaba submission to the WMT20 parallel corpus filtering task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 979–984.

Marco Lui and Timothy Baldwin. 2012. langid.py: An off-the-shelf language identification tool. In *Proceedings of the ACL 2012 System Demonstrations*, pages 25–30, Jeju Island, Korea. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

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

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Jeffrey Pennington, Richard Socher, and Christopher D 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.

Nick Rossenbach, Jan Rosendahl, Yunsu Kim, Miguel Graça, Aman Gokrani, and Hermann Ney. 2018. The RWTH Aachen University filtering system for the WMT 2018 parallel corpus filtering task. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 946–954, Belgium, Brussels. Association for Computational Linguistics.

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

Masoud Jalili Sabet, Philipp Dufter, François Yvon, and Hinrich Schütze. 2020. Simalign: High quality word alignments without parallel training data using static and contextualized embeddings. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 1627–1643. Association for Computational Linguistics.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*. Cambridge, MA.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Rui Wang, Andrew M. Finch, Masao Utiyama, and Eiichiro Sumita. 2017. Sentence embedding for neural machine translation domain adaptation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 560–566.

Albert Zeyer, Tamer Alkhouli, and Hermann Ney. 2018. RETURNN as a generic flexible neural toolkit with application to translation and speech recognition. In *Annual Meeting of the Assoc. for Computational Linguistics*, volume abs/1805.05225.