Subword Mapping and Anchoring across Languages

Giorgos Vernikos[1,2]
[1]HEIG-VD
Yverdon-les-Bains, Switzerland
`georgios.vernikos@heig-vd.ch`

Andrei Popescu-Belis[1,2]
[2]EPFL School of Engineering
Lausanne, Switzerland
`andrei.popescu-belis@heig-vd.ch`

Abstract

State-of-the-art multilingual systems rely on shared vocabularies that sufficiently cover all considered languages. To this end, a simple and frequently used approach makes use of subword vocabularies constructed jointly over several languages. We hypothesize that such vocabularies are suboptimal due to *false positives* (identical subwords with different meanings across languages) and *false negatives* (different subwords with similar meanings). To address these issues, we propose Subword Mapping and Anchoring across Languages (SMALA), a method to construct bilingual subword vocabularies. SMALA extracts subword alignments using an unsupervised state-of-the-art mapping technique and uses them to create cross-lingual anchors based on subword similarities. We demonstrate the benefits of SMALA for cross-lingual natural language inference (XNLI), where it improves zero-shot transfer to an unseen language without task-specific data, but only by sharing subword embeddings. Moreover, in neural machine translation, we show that joint subword vocabularies obtained with SMALA lead to higher BLEU scores on sentences that contain many false positives and false negatives.

1 Introduction

NLP systems that operate in more than one language have been proven effective in tasks such as cross-lingual natural language understanding and machine translation (Devlin et al., 2019; Conneau et al., 2020a; Aharoni et al., 2019). The performance of such systems is strongly connected to their use of an input space that can sufficiently represent all the considered languages (Sennrich et al., 2016; Wu and Dredze, 2019; Conneau et al., 2020a). Conceptually, an effective cross-lingual input space should exploit latent similarities between languages.

State-of-the-art multilingual systems take advantage of cross-lingual similarities in their input spaces through the use of a shared vocabulary of subwords. This vocabulary is learned on the concatenation of multilingual training corpora, using heuristic subword segmentation algorithms (Sennrich et al., 2016; Schuster and Nakajima, 2012; Kudo, 2018), which handle the open vocabulary problem by identifying tokens at multiple granularity levels, based on character n-gram frequencies. Therefore, the embeddings of subwords that appear in several languages act as anchors between these languages and, thus, provide implicit cross-lingual information that leads to improved performance (Conneau and Lample, 2019; Pires et al., 2019; Conneau et al., 2020b).

Cross-lingual transfer in joint subword models may be limited by false positives, i.e. identical subwords with different meanings in two languages, a phenomenon also known as ‘oversharing’ (Wang et al., 2020b; Dhar and Bisazza, 2021). Moreover, they do not benefit from false negatives, i.e. different subwords with identical meanings. Examples of false positives are: *die*, a definite article in German and a verb in English; *also*, meaning ‘so’ or ‘therefore’ in German, not ‘as well’ as in English; or *fast*, which in German means ‘almost’, not ‘quick’. Examples of false negatives are *and* and *und*, *very* and *sehr*, *people* and *Menschen* – all pairs being near synonyms that could benefit from a unique embedding rather than two. A unique embedding would not constrain the models to always represent or translate them in the same way, as representations are highly contextualized.

In this paper, we address the problem of false positives and negatives by employing *subword similarity to create cross-lingual anchors*. Specifically, using cross-lingual mapping, we determine subword alignments for a set of subwords, and then share their representations. In this way, we relax the requirements for isomorphism and common scripts between languages on which previous studies rely. We demonstrate that this can improve both

Findings of the Association for Computational Linguistics: EMNLP 2021, pages 2633–2647
November 7–11, 2021. ©2021 Association for Computational Linguistics

cross-lingual transfer of language models and machine translation (MT). Our contributions are the following:

1. We propose a method for subword mapping and anchoring across two languages (SMALA), with no constraints on the availability of parallel data or the similarity of scripts (Section 3).
2. We show how SMALA can be used to extend an existing monolingual vocabulary and facilitate cross-lingual transfer of a pre-trained language model to an unseen language under a limited parameter budget (Section 4).
3. We demonstrate experimentally the benefits of SMALA for cross-language natural language inference (XNLI) (Section 5).
4. We demonstrate how SMALA can be used to build a shared vocabulary for MT, and bring experimental evidence of its benefits (Section 6).

We release our code online[1].

2 Related Work

Cross-lingual representations. A large body of work has attempted to harness the similarities of languages via cross-lingual word embeddings, i.e. continuous word vectors that can represent multiple languages in a shared vector space. A first approach to obtain these embeddings is offline mapping of pre-trained monolingual embeddings, where the mapping can be learned using supervision in the form of lexicons (Mikolov et al., 2013b; Xing et al., 2015; Joulin et al., 2018), or by leveraging weak supervision in the form of identical seed words (Artetxe et al., 2017; Søgaard et al., 2018), or in an unsupervised way (Artetxe et al., 2018; Lample et al., 2018a). A second approach to obtain cross-lingual embeddings is joint training from scratch, by combining monolingual language modeling objectives with a cross-lingual objective – with either strong, or weak, or no supervision (see respectively Luong et al., 2015; Duong et al., 2016; Lample et al., 2018b).

Despite their success, both approaches have certain limitations. On the one hand, alignment methods assume that the monolingual embedding spaces have comparable structures, i.e., that they are isomorphic to a certain extent. However, this assumption has been challenged, especially for etymologically distant languages, but also for related ones (Søgaard et al., 2018; Patra et al., 2019; Ormazabal et al., 2019). Unsupervised joint training, on the other hand, relies on the assumption that identical tokens carry the same information across languages, which is not always true.

To address the limitations of alignment and joint training (the isomorphism assumption and requirement for common script), combinations of the two methods have been proposed. Wang et al. (2020b) jointly train embeddings on concatenated monolingual corpora and then "unshare" identical words across languages, reallocating the overshared word embeddings and subsequently aligning them. Ormazabal et al. (2021) find word alignments that are used as anchors to create cross-lingual representations with a modified version of Skip-gram (Mikolov et al., 2013a). Our approach shares a similar motivation, but instead of directly creating cross-lingual representations, we shape the input space (i.e. the vocabulary) of multilingual systems in a way that facilitates cross-lingual transfer.

Subword vocabularies. Recently, multilingual language models have superseded cross-lingual word embeddings, not only because they produce contextualized representations, but also because they can handle the open vocabulary problem through the use of subwords as tokens (Sennrich et al., 2016; Schuster and Nakajima, 2012; Kudo, 2018). Multilingual subword vocabularies are simply obtained by learning the subwords on the concatenation of all used languages. Since each subword is assigned to a unique embedding, identical subwords that appear in several languages serve as anchors between languages, providing implicit cross-lingual information (Wu and Dredze, 2019; Pires et al., 2019; Conneau et al., 2020b). Parameter sharing across languages make subword models particularly suitable for multilingual NLP and machine translation.

The number of shared tokens in multilingual vocabularies highly depends on the similarities of script between languages. When this is not the case, transliteration can be applied (Nguyen and Chiang, 2017; Müller et al., 2020; Amrhein and Sennrich, 2020). In addition, shared subword vocabularies often produce inconsistent segmentations across languages that can hurt cross-lingual transfer. Regular-

[1] https://github.com/GeorgeVern/smala

ization techniques that introduce randomness in the tokenization process (Kudo, 2018; Provilkov et al., 2020) can partially address this problem, or consistency between the different segmentations can be otherwise enforced (Wang et al., 2021). Still, there is no guarantee that shared (sub)words have identical meanings (false positives are not excluded) and, conversely, subwords with identical meanings but different spellings (false negatives) are missed.

Cross-lingual LM transfer. The success of pretrained monolingual and multilingual language models raises the question of whether these models can be transferred to unseen languages. To transfer such a model, it is mostly necessary to add language-specific parameters in the form of a subword embedding layer, which can be learned from scratch (Artetxe et al., 2020; de Vries and Nissim, 2021). Alternatively, offline mapping can be used to initialize the new embedding layer, for faster convergence and improved zero-shot performance (Tran, 2020). Another option, which reduces the computational cost of this transfer but assumes similarity of scripts, is to leverage common subwords between languages (Chronopoulou et al., 2020; Wang et al., 2020a). Our proposal combines the two approaches without the requirement for a common script.

Recent work has shown that cross-lingual transfer can still be achieved in the absence of anchors (i.e. subwords shared between languages), although the existence of anchors contributes to performance gains (Artetxe et al., 2020; Conneau et al., 2020b; Aji et al., 2020). Specifically, Conneau et al. (2020b) have shown that performance increases with the number of available anchors. However, these studies do not discuss the quality of anchors, or how they can be obtained, which is the main focus of our work.

3 SMALA: Subword Mapping and Anchoring across Languages

Our motivation is to create cross-lingual vocabularies that are parameter-efficient and exploit the similarity of concepts between different languages. We propose a method for Subword Mapping and Anchoring across Languages (SMALA), which combines the powerful initialization of mapping methods with the anchoring properties of joint training, while attempting to alleviate the limitations of both methods. We first learn subwords separately for each language and then train the corresponding embeddings. We then apply a mapping method to obtain similarity scores between the embeddings, which we use to extract alignments between subwords of the two languages. We finally tie the parameters of the aligned subwords to create anchors during training. We describe hereafter in detail the two main components of our approach.

3.1 Subword Mapping

As a first step, we aim to find subwords that have similar meanings or functions (morphological or syntactic) between different languages, i.e. to extract subword alignments. To this end, we first learn separate subword vocabularies for each language from monolingual data using one of the existing subword segmentation algorithms (specified below for each series of experiments). Since we argue against using identical subwords as anchors between languages, we employ a distributional method to find the alignments: we obtain subword representations for each language from monolingual data from FastText embeddings (Bojanowski et al., 2017)[2] and then align them using a state-of-the-art unsupervised alignment approach, VecMap (Artetxe et al., 2018).

Our method can also exploit parallel data, when it is available. In this case, we tokenize both sides of the bitext with language-specific subwords and then use FastAlign (Dyer et al., 2013) to estimate the alignment, similar to Tran (2020). Implementation details can be found in Appendix A.1.

3.2 Anchoring of Similar Subwords

After the mapping step, we apply cosine similarity[3] to compute a similarity matrix S: each of its coefficients $S_{i,j}$ is the cosine similarity between the embeddings of the i^{th} subword of language $\mathcal{L}_1$ and of the j^{th} subword of language $\mathcal{L}_2$.

We use the similarity matrix S to identify alignments between subwords in a completely unsupervised way. We extract the aligned subword alignments using the *Argmax* method of Jalili Sabet et al. (2020), as follows. A subword $w_i^{L_1}$ from the $\mathcal{L}_1$ vocabulary is aligned to a subword $w_j^{L_2}$ from the $\mathcal{L}_2$ vocabulary, if and only if $w_j^{L_2}$ is the most similar subword to $w_i^{L_1}$ and vice versa:

$$i = \arg\max_l(S_{l,j}) \text{ and } j = \arg\max_l(S_{i,l}) \quad (1)$$

[2] The use of subword co-occurrence and PCA appeared to underperform with respect to FastText.

[3] We also experimented with CSLS retrieval (Lample et al., 2018a) but it produced more alignments of lower quality.

Each pair of subwords that satisfies the above consistency condition forms an alignment, to which we assign a score: the average similarity $(S_{i,j}+S_{j,i})/2$. This will be used as a threshold to select a subset of all alignments. We thus obtain a dictionary D of aligned subwords that will function as anchors between languages during training, by tying their embeddings.

The above definition implies that the aligned subwords are translations of one another. Although this might seem quite limiting, the same issue arises for joint vocabulary construction, with the difference being the criterion according to which we choose to share subwords. We argue that our similarity is a more expressive criterion than the raw surface form. Our approach does not rely on the surface form for cross-lingual anchors and additionally removes the requirement for a common script. Furthermore, it prevents sharing subwords that are identical but differ in meaning (false positives) and allows sharing subwords that are spelled differently but are close to synonyms (false negatives). The (sub)words aligned by our method may or not be identical, as long as they satisfy Equation 1.

4 Language Model Transfer with SMALA

For the first set of experiments, we attempt to transfer a pretrained Language Model (LM) from one language ($\mathcal{L}_1$) to another language ($\mathcal{L}_2$), by leveraging the linguistic knowledge that was implicitly encoded in $\mathcal{L}_1$'s embedding layer. Following previous work (Artetxe et al., 2020; Tran, 2020), we create an embedding layer for $\mathcal{L}_2$ and initialize it by sharing parameters using SMALA. In this way, we aim to reduce the computational budget of cross-lingual transfer via parameter sharing without sacrificing performance, but removing the need for a common script and the pitfalls of false positives and false negatives.

We transfer the model following the same steps as Tran (2020). We start from a pretrained LM that we continue training on masked language modeling (MLM) using monolingual data from both the original and the target languages ($\mathcal{L}_1$ and $\mathcal{L}_2$). The bilingual model has two *separate embedding layers*, one for $\mathcal{L}_1$ and one for $\mathcal{L}_2$, while the rest of the encoder is common to $\mathcal{L}_1$ and $\mathcal{L}_2$. Each language-specific embedding layer is used both as the first and last layer (tied embeddings). During this training phase, we keep including monolingual data from $\mathcal{L}_1$ to avoid degradation in performance in the original language and maximize cross-lingual transfer (Pires et al., 2019; Conneau et al., 2020b). We update the weights of the whole model during this phase, since updating only the embeddings would not significantly reduce computation time (due to the need to calculate all activations for backpropagation) and has actually a negative impact on performance, as we observed in our initial experiments. At this stage, the transferred model could be used for any cross-lingual natural language understanding task (Hu et al., 2020) or for unsupervised machine translation (Conneau and Lample, 2019; Chronopoulou et al., 2020; Liu et al., 2020).

In a second stage, we fine-tune the model for XNLI (Conneau et al., 2018) on labeled data in $\mathcal{L}_1$ (English), using $\mathcal{L}_1$ embeddings and freezing the embedding layer. Finally, we zero-shot transfer the model to $\mathcal{L}_2$ data by simply changing the language-specific embedding layer.

5 Experiments with XNLI

5.1 Models

We compare several models in our experiments on cross-lingual natural language inference (textual entailment) with the XNLI dataset (Conneau et al., 2018). We note that all models, with the exception of mBERT, follow the pipeline from the previous section to transfer the pretrained LM to a new language. The only difference between these models is the way the new embedding layer is created.

JOINT. A system that employs parameter sharing based on surface form, that is, the union of the two language-specific vocabularies, similar to joint tokenization. The embeddings for the tokens that are not shared with the original embedding layer are initialized randomly.

This model allows for a comparison between anchoring identical vs. semantically similar subwords identified by SMALA, as an inductive bias for cross-lingual vocabularies. Although this is not exactly the same as joint tokenization, previous works have suggested that performance is similar (Aji et al., 2020; Conneau et al., 2020b) and that a language-specific embedding layer and tokenizer can have a positive impact on performance (Rust et al., 2021; Pfeiffer et al., 2020).

OURS. Our approach (SMALA) leverages similarity to find alignments between subwords. The parameters of the subwords are then tied, as explained above. Our system is directly comparable to JOINT, since we only use monolingual data to find the alignments, and the non-aligned subwords are randomly initialized.

OURS+ALIGN. Random initialization of the non-aligned subwords requires more computation to reach convergence (Artetxe et al., 2020) and/or can lead to subpar performance[4] (Tran, 2020; Aji et al., 2020). Therefore, we also propose a system which initializes the non-aligned subwords using the similarity matrix S from which we calculated the subword alignments. Following Tran (2020), we use *sparsemax* (Martins and Astudillo, 2016) to initialize the non-shared $\mathcal{L}_2$ subwords as a sparse weighted sum of $\mathcal{L}_1$ subwords. We experiment with either monolingual or parallel data to learn the similarity matrix S in this case.

RAMEN. RAMEN (Tran, 2020) leverages alignments learned from either monolingual or parallel data to initialize the $\mathcal{L}_2$ subword embeddings. Unlike our approach, for monolingual data, common words are used to initialize a supervised word alignment method (Joulin et al., 2018), and then the word alignment is transferred to subwords using several approximations. In contrast to our method, RAMEN does not employ any parameter sharing but trains a full embedding layer for $\mathcal{L}_2$.

mBERT. For comparison, we use multilingual BERT (Devlin et al., 2019) in the same zero-shot cross-lingual transfer setting. However, results are not strictly comparable to the above models, since mBERT has a larger shared vocabulary, hence more parameters (178M compared to 133M for RAMEN) and is trained for more steps. We include mBERT in our experiments as a reference for high-performing multilingual models.

5.2 Data and Settings

For XNLI experiments, we select five target languages that vary in terms of language family, typology and script: Spanish (Es), German (De), Greek (El), Russian (Ru) and Arabic (Ar). We obtain monolingual corpora from the Wikipedia of each language using WikiExtractor[5]. We use these corpora for MLM training, similar to Devlin et al. (2019), and to extract subword alignments using SMALA. When parallel data is used, we either use Europarl (Koehn et al., 2007) or the United Nations Parallel Corpus (Ziemski et al., 2016). We use the same amount of parallel data for each pair and we subsample the data, if needed. Both monolingual and parallel data are lowercased and tokenized with the Moses tokenizer (Koehn et al., 2007).

For our implementation we use Hugging Face's Transformers library (Wolf et al., 2019) and for RAMEN we use the public implementation from the author. We choose BERT-BASE (110M parameters) as our pretrained LM. We further train all bilingual models on MLM for 120k steps with a batch size of 76 and a maximum sequence length of 256. Each batch contains equal numbers of samples from both languages, similar to Tran (2020). We optimize bilingual LMs using Adam (Kingma and Ba, 2015) with bias correction, a learning rate of 5e−5 and linear decay.

We fine-tune the adapted bilingual LMs on the MultiNLI dataset (Williams et al., 2018) in *English*, using a batch size of 32 and a maximum sequence length of 256. We also use Adam with a learning rate of 2e−5, a linear warm up schedule over the 10% initial steps, bias correction and linear decay. We fine-tune each model for five epochs and evaluate five times per epoch, as suggested by Dodge et al. (2020). We select the best model based on validation loss.

We evaluate on the test data for $\mathcal{L}_2$ from the XNLI dataset (Conneau et al., 2018), with no specific training for $\mathcal{L}_2$ (zero-shot). As in the robust evaluation scheme for zero-shot cross-lingual transfer used by Wu and Dredze (2020), we report mean and variance over the systems resulting from five different runs of the fine-tuning stage, with the same hyper-parameters but different seeds. We did not perform any exhaustive hyper-parameter search for this task, and use the exact same settings for all model variants and languages.

For each target language, we learn a new subword vocabulary using the WordPiece[6] algorithm (Schuster and Nakajima, 2012). The bilingual models contain two language-specific embedding layers corresponding to these vocabularies.[7] For RAMEN, which does not share parameters, the size of the $\mathcal{L}_2$ embedding layer is the same as the original one. For methods that employ sharing (OURS and JOINT), the parameters of the shared subwords are tied, reducing the size of the new embedding layer. Table 2 presents the percentage of the $\mathcal{L}_2$ embeddings that are shared with $\mathcal{L}_1$ for all methods.

[4] In our experiments, even a random alignment produced better results than random initialization.

[5] https://github.com/attardi/wikiextractor

[6] As implemented at: https://huggingface.co/docs/tokenizers/python/latest/.

[7] Following Tran (2020), we initialize special tokens ([CLS], [SEP], [MASK], [PAD] and [UNK]) with their pretrained representations, in all methods except mBERT.

Method	Data	Es	De	El	Ru	Ar
JOINT	mono	70.0 ± 0.2	64.4 ± 0.8	61.2 ± 0.9	56.2 ± 1.1	45.8 ± 0.4
OURS	mono	**74.2** ± 0.4	**70.6** ± 0.1	**70.0** ± 0.7	**65.4** ± 0.9	**62.3** ± 0.4
OURS+ALIGN	mono	76.5 ± 0.4	72.8 ± 0.5	72.9 ± 0.5	70.2 ± 0.6	67.0 ± 0.4
OURS+ALIGN	para	**77.1** ± 0.8	**74.1** ± 0.5	**75.1** ± 0.7	**71.9** ± 0.4	**67.8** ± 0.8
RAMEN	mono	76.5 ± 0.6	72.5 ± 0.8	72.5 ± 0.8	68.6 ± 0.7	66.1 ± 0.8
RAMEN	para	**77.3** ± 0.6	**74.1** ± 0.9	**74.5** ± 0.6	**71.6** ± 0.8	**68.6** ± 0.6
mBERT	mono	74.9 ± 0.4	71.3 ± 0.6	66.6 ± 1.2	68.7 ± 1.1	64.7 ± 0.6

Table 1: Zero-shot classification scores (accuracy) on XNLI: mean and standard deviation over 5 runs, when either monolingual or parallel corpora are used for alignment (or token matching for JOINT). Systems in the first 4 rows use parameter sharing, while those in rows 5-6 train a full embedding layer. Moreover, rows 1-2 only share subwords, while rows 3-4 also use alignment for initialization. The best model in each subgroup is in **bold**.

Method	Data	Es	De	El	Ru	Ar
JOINT	mono	26%	25%	11%	9%	10%
OURS	mono	44%	37%	33%	31%	30%
OURS	para	32%	26%	21%	21%	15%

Table 2: Percentage of $\mathcal{L}_2$ embeddings that are shared with $\mathcal{L}_1$ (English) for each system and language.

5.3 Results on XNLI

We present the results of our experiments on XNLI in Table 1. Our approach is significantly better than sharing based on surface form (OURS vs. JOINT), and the improvement increases with the distance of $\mathcal{L}_2$ from English (for Greek, Russian and Arabic). This can be attributed to the erroneous sharing of non-informative subwords (e.g. letters and English words) in the JOINT model.

Our approach is more parameter-efficient than JOINT, as shown in Table 2, as it enables the sharing of a larger number of embeddings, especially for distant languages. Therefore, despite the smaller number of parameters, results are significantly improved. Moreover, the results also demonstrate the applicability of our approach to languages with different scripts.

Among methods that do not make use of parallel data (rows 1-3 and 5 in Table 1), we notice a significant gap between the performance of anchoring based on surface form (JOINT) and training a full embedding layer, without sharing, initialized by alignment (RAMEN with mono). Our approach can sufficiently bridge this gap, with a smaller number of parameters, demonstrating the importance of the choice of anchors in cross-lingual vocabularies.

Among methods that use alignment (rows 3-6), our approach with additional alignment of the non-shared subwords (OURS+ALIGN) performs on par or better than RAMEN. This trend is consistent across the use of monolingual and parallel data for the alignment. In the latter case, the alignment is learned with the same method and data in both systems. Our higher score supports our claim that better anchoring can lead to more parameter-efficient vocabularies without sacrificing performance.

Finally, in Table 1, we observe that all methods that employ alignment outperform mBERT. In some cases, even our approach without alignment performs comparably (Es, De) or even better (El) than mBERT. These results show that our method – which transfers a monolingual LM to an unseen language with minimal computation demands – is a competitive alternative to using an off-the-shelf multilingual model. This is particularly useful when the considered language is not modeled well (e.g. Greek) or not covered at all by the multilingual model.

6 Experiments with Machine Translation

In the second set of experiments, we apply SMALA to MT by leveraging subword alignments to create shared bilingual vocabularies from scratch, instead of joint subword vocabularies learned on concatenated source and target corpora.

6.1 Applying SMALA to MT

The majority of current Transformer-based MT systems (Vaswani et al., 2017) share the vocabulary and the corresponding embedding layer between the encoder and the decoder of a sequence-to-sequence architecture. To apply SMALA to MT, instead of jointly learning the subwords on the concatenated corpora, we learn separate subword vocabularies for each language, and then merge them into a joint one. We use SMALA to extract

Languages	En-Ru		En-De		En-Ro		En-Ar	
Data	25M		5.85M		612k		239k	
	←	→	←	→	←	→	←	→
JOINT	30.0	26.1	32.1	27.1	30.9	23.2	29.0	11.8
OURS	30.2	**26.6**	32.1	27.0	30.8	23.3	28.8	12.2

Table 3: BLEU scores of baseline and our system for machine translation. Language pairs are ordered by decreasing size of training data (numbers of sentences). **Bold** indicates statistical significance ($p < 0.05$).

alignments from the available parallel data of each language pair, and use aligned pairs as unique subwords (shared entries), serving as anchors in the shared embedding layer. These anchors play the same role as identical subwords in joint vocabularies, and thus address the problem of false negatives. Conversely, identical subwords that are not aligned with SMALA remain two distinct language-specific entries, thus addressing the problem of false positives.

To create a subword vocabulary of a given size n using SMALA, we first learn two monolingual vocabularies of size $m > n$, one for the source and one for the target language. Then, we select a number of alignments α with the highest similarity scores, as defined in Section 3.2, with $\alpha = 2m - n$. This ensures that, when the two vocabularies are joined and the α pairs of anchors are merged, the size of the resulting vocabulary is n.

6.2 Data, Tools and Settings

We choose four language pairs that represent different levels of data availability and language relatedness, and run experiments in both directions: Russian, German, Romanian and Arabic, to and from English. Training and test data comes from WMT17[8] for En-Ru and En-De, WMT16[9] for En-Ro, and IWSLT17[10] for En-Ar. We tokenize the data using the Unigram LM model (Kudo, 2018) as implemented in SentencePiece[11]. We choose the size of the shared subword vocabulary based on the size of the data, following Kudo (2018): 32k for high-resource pairs (En-Ru and En-De) and 16k for medium and low-resource pairs (En-Ro and En-Ar). We tokenize data using the Moses Tokenizer (Koehn et al., 2007). We report BLEU scores (Papineni et al., 2002) obtained with SacreBLEU (Post, 2018) on detokenized text.[12]

We train OpenNMT-py (Klein et al., 2017) for a maximum of 100k steps on high-resource pairs and 40k steps on medium or low-resource ones. Our base model is Transformer-Base (L=6, H=512) (Vaswani et al., 2017) with the same regularization and optimization procedures. We use a batch size of 4k tokens and evaluate every 5k steps. We select the best model based on validation loss. Final translations are generated with a beam width of five.

6.3 Results

We present the results for our method and the baseline in Table 3. Our method yields comparable results to the baseline across all conditions of data availability and language relatedness. This demonstrates the viability of SMALA as an alternative for the creation of shared bilingual vocabularies. We observe a slight increase in performance in distant language pairs (En-Ru and En-Ar), which could be explained by the difference in scripts. Indeed, joint tokenization (baseline system) is not able to identify anchors when the script is not shared between languages, resorting to a small number of shared subwords that are mostly uninformative, often due to the presence of English words in the other language. In this case, the anchors found by SMALA (subword pairs corresponding to false negatives in the baseline) help to improve the joint vocabulary.

Comparing the results of Tables 1 and 3 we see that our approach does not equally improve results in both settings. We attribute this difference to the amount of supervision available in MT in the form of bitext, and to the strong contextual constraints from the decoder. Although false positives and negatives are present in both scenarios, the availability of parallel data for training forces NMT models to disambiguate these subwords based on context in both languages at the same time.

[8] http://statmt.org/wmt17/translation-task.html

[9] http://statmt.org/wmt16/translation-task.html

[10] TED talks from: https://wit3.fbk.eu/

[11] https://github.com/google/sentencepiece

[12] Signature: BLEU+c.mixed+#.1+.exp+tok.13a+v.1.5.1

7 Analysis

In this section we attempt to quantify the effect of false positives and false negatives on each of the tasks.

7.1 Ablation Study on XNLI

We begin with a model that creates cross-lingual anchors based on surface form (JOINT) and we address either false positives only (−FP) or false negatives only (−FN) among shared subwords. In the latter case, if a subword is both a false positive and a false negative, then we treat it as a false negative – e.g., *also* in English should be not aligned with *also* in German but with *auch*. We follow the pipeline of Section 4 and present the results in XNLI in Table 4.

Method	Es	De	El	Ru	Ar
JOINT	70.0	64.4	61.2	56.2	45.8
−FP	68.5	61.7	62.6	53.6	44.8
−FN	74.3	70.0	70.2	65.8	63.1
OURS (−FP−FN)	74.2	70.6	70.0	65.4	62.3

Table 4: Effect of removing false positives or false negatives in XNLI (accuracy).

We observe that by only removing false positives (−FP) performance drops compared to JOINT. This can be attributed to the ability of the model to disambiguate false positives in the presence of context. But this could also be due to a limitation of our method to identify false positives with high precision especially (sub)words that have more than one sense. Conversely, the problem of false negatives seems to be the most important and by addressing it (−FN) results improve significantly over JOINT. The similar performance of −FN and OURS may be due to the removal of certain false positives along with many false negatives (see also Appendix A.2).

7.2 False Positives and Negatives in MT

In order to quantify the effect of false positives and false negatives in MT, we compare the performance of joint tokenization with SMALA for cases where the presence of such subwords is significant. Table 5 presents BLEU scores for sentences that contain a high percentage of false positives and/or negatives (more than 50% of the tokens) in the source side, along with the number of sentences in this case. BLEU scores for percentages between 0% and 60% are represented graphically in the Appendix, Figure 4.

Languages	En-Ru		En-De		En-Ro		En-Ar	
	←	→	←	→	←	→	←	→
Sentences	49	2225	1674	2216	1249	1295	141	866
JOINT	39.2	27.6	33.1	27.0	31.6	24.6	37.8	16.2
OURS	42.2	28.0	33.0	27.0	32.0	24.8	40.4	16.6
Δ	+3.0	+0.4	-0.1	0.0	+0.4	+0.2	+2.6	+0.3

Table 5: BLEU scores for sentences where 50% of tokens are false positives and/or false negatives. The number of selected sentences (out of a total of 3,000) is given for each translation direction.

The results of Table 5 show improved performance of our method over the baseline, confirming our original intuition regarding false positives and negatives. Despite the fact that MT models with joint tokenization use context to disambiguate false positives – as it can help to also disambiguate polysemous words to a certain extent (Rios Gonzales et al., 2017; Pu et al., 2018) – when their number increases performance tends to drop compared to SMALA. The gap in performance between JOINT and OURS (using SMALA) is bigger for pairs that do not have shared scripts (En-Ru and En-Ar) which is a possible indication of the impact of false negatives, despite the smaller sample sizes. Overall, the results of Tables 3 and 5 demonstrate that our approach is competitive with joint tokenization in most cases and superior in challenging cases with multiple false positives and negatives.

7.3 Cross-lingual Word Representations

In order to validate our claim that SMALA facilitates cross-lingual transfer, we perform an intrinsic evaluation of the obtained representations. We compare the quality of representations created using SMALA vs. joint tokenization for Bilingual Lexicon Induction (BLI), a standard evaluation task for cross-lingual word embedding methods. Specifically, we compare the performance of the bilingual models from the first setting (see Section 4) after the bilingual MLM training step, but before the XNLI fine-tuning. We do not include methods that use alignment to initialize the embedding layer (for these results see Appendix A.5), in order to isolate the effect of anchors.

We follow the setting of Vulić et al. (2020) to compute word-level representations. We encode each word in isolation using the model, in the form [CLS] *word* [SEP]. We extract the representations from the embedding layer excluding representations of special tokens. If a word is split into more than one subword, we average the obtained representations. We perform this operation for every word of the test set for both languages. We retrieve word translation using *Cross-Domain Similarity Local Scaling* (CSLS) with K=10 number of neighbours (Lample et al., 2018a).

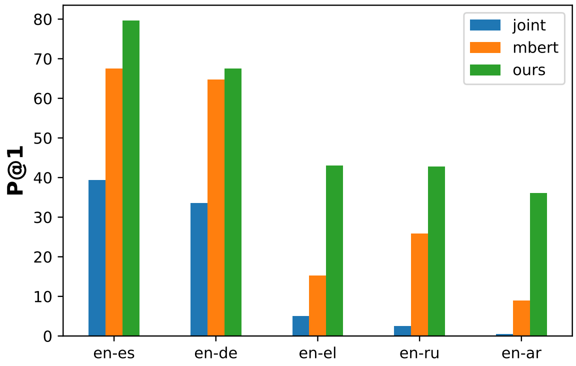

Figure 1: Precision@1 results for the BLI task.

Our results on the MUSE benchmark (Lample et al., 2018a), a bilingual dictionary induction dataset, are presented in Figure 1, using precision at 1 scores (P@1), following standard practices. We observe that by using SMALA to create cross-lingual anchors (OURS) we can greatly improve performance on BLI compared to methods that use identical subwords (JOINT and mBERT). Figure 1 also shows that the performance of JOINT and mBERT significantly decreases as the two languages are more distant and their vocabulary does not have considerable overlap, which points at the limitations of joint tokenization and especially false negatives which are the most frequent in this case.

Similar to Wang et al. (2020b), we also evaluate on words that are not shared, by removing test pairs with the same surface form (e.g. (*epic*, *epic*) as a test pair for en-es) and present the difference in performance in Figure 2. We find that the performance of JOINT and mBERT decreases significantly, unlike OURS. For languages with different scripts (en-el, en-ru and en-ar) the performance of our approach even increases in this scenario due to the fact that our system is able identify and not retrieve false positives. This confirms our intuition that the use of surface form to create cross-lingual anchors leads to poorly aligned cross-lingual representations for the non-shared subwords.

8 Conclusion

In this work we introduced SMALA, a novel approach to construct shared subword vocabularies that leverages similarity instead of identical subwords to create anchors. We demonstrate that our approach outperforms current methods for joint construction of multilingual subword vocabularies in cases where there is no cross-lingual signal, apart from the anchors. When cross-lingual supervision is available, our approach performs comparably to the baseline, while showing improved performance in cases with numerous false positive and false negatives.

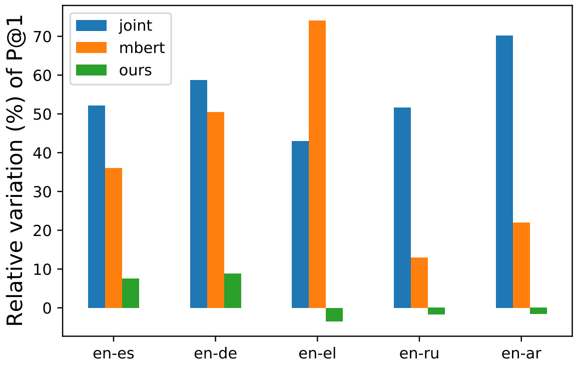

Figure 2: Precision@1 difference on BLI when test pairs of same surface form are removed.

In future work, we aim to extend our method to more than two languages. We also intend to explore the effectiveness of SMALA for closely related languages and compare SMALA to other approaches, such as those using transliteration. In addition, we aim to apply SMALA to settings of varying cross-lingual supervision levels, such as unsupervised MT.

Acknowledgments

We are grateful for their support to the Swiss National Science Foundation through grant n. 175693 for the DOMAT project: “On-demand Knowledge for Document-level Machine Translation” and to Armasuisse for the FamilyMT project.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. 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 3874–3884.

Alham Fikri Aji, Nikolay Bogoychev, Kenneth Heafield, and Rico Sennrich. 2020. In neural machine translation, what does transfer learning transfer? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7701–7710.

Chantal Amrhein and Rico Sennrich. 2020. On romanization for model transfer between scripts in neural machine translation. *Findings of the Association for Computational Linguistics: EMNLP 2020*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2017. Learning bilingual word embeddings with (almost) no bilingual data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 451–462.

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 (Volume 1: Long Papers)*, pages 789–798.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Alexandra Chronopoulou, Dario Stojanovski, and Alexander Fraser. 2020. Reusing a pretrained language model on languages with limited corpora for unsupervised NMT. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2703–2711.

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

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

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485.

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

Wietse de Vries and Malvina Nissim. 2021. As good as new. How to successfully recycle English GPT-2 to make models for other languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 836–846, Online. Association for Computational Linguistics.

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

Prajit Dhar and Arianna Bisazza. 2021. Understanding cross-lingual syntactic transfer in multilingual recurrent neural networks. In *Proceedings of the 23rd Nordic Conference on Computational Linguistics (NoDaLiDa)*, pages 74–85.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah A. Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *CoRR*, abs/2002.06305.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295.

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.

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

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

Armand Joulin, Piotr Bojanowski, Tomas Mikolov, Hervé Jégou, and Edouard Grave. 2018. Loss in translation: Learning bilingual word mapping with a retrieval criterion. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2979–2984.

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

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72, Vancouver, Canada. Association for Computational Linguistics.

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

Taku Kudo. 2018. Subword regularization: Improving neural network translation models with multiple subword candidates. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 66–75, Melbourne, Australia. Association for Computational Linguistics.

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

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

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159.

Andre Martins and Ramon Astudillo. 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1614–1623.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*.

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

Benjamin Müller, Antonis Anastasopoulos, Benoît Sagot, and Djamé Seddah. 2020. When being unseen from mBERT is just the beginning: Handling new languages with multilingual language models. *CoRR*, abs/2010.12858.

Toan Q. Nguyen and David Chiang. 2017. Transfer learning across low-resource, related languages for neural machine translation. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 296–301.

Aitor Ormazabal, Mikel Artetxe, Gorka Labaka, Aitor Soroa, and Eneko Agirre. 2019. Analyzing the limitations of cross-lingual word embedding mappings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4990–4995.

Aitor Ormazabal, Mikel Artetxe, Aitor Soroa, Gorka Labaka, and Eneko Agirre. 2021. Beyond offline mapping: Learning cross-lingual word embeddings through context anchoring. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6479–6489, Online. Association for Computational Linguistics.

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.

Barun Patra, Joel Ruben Antony Moniz, Sarthak Garg, Matthew R. Gormley, and Graham Neubig. 2019. Bilingual lexicon induction with semi-supervision in non-isometric embedding spaces. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 184–193.

Jonas Pfeiffer, Ivan Vulic, Iryna Gurevych, and Sebastian Ruder. 2020. UNKs everywhere: Adapting multilingual language models to new scripts. *CoRR*, abs/2012.15562.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191.

Ivan Provilkov, Dmitrii Emelianenko, and Elena Voita. 2020. BPE-dropout: Simple and effective subword regularization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1882–1892, Online. Association for Computational Linguistics.

Xiao Pu, Nikolaos Pappas, James Henderson, and Andrei Popescu-Belis. 2018. Integrating weakly supervised word sense disambiguation into neural machine translation. *Transactions of the Association for Computational Linguistics*, 6:635–649.

Annette Rios Gonzales, Laura Mascarell, and Rico Sennrich. 2017. Improving word sense disambiguation in neural machine translation with sense embeddings. In *Proceedings of the Second Conference on Machine Translation*, pages 11–19, Copenhagen, Denmark. Association for Computational Linguistics.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? On the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135.

Mike Schuster and Kaisuke Nakajima. 2012. Japanese and Korean voice search. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 5149–5152.

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.

Anders Søgaard, Sebastian Ruder, and Ivan Vulić. 2018. On the limitations of unsupervised bilingual dictionary induction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 778–788.

Ke M. Tran. 2020. From English to foreign languages: Transferring pre-trained language models. *CoRR*, abs/2002.07306.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Ivan Vulić, Edoardo Maria Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7222–7240.

Xinyi Wang, Sebastian Ruder, and Graham Neubig. 2021. Multi-view subword regularization. *CoRR*, abs/2103.08490.

Zihan Wang, Karthikeyan K, Stephen Mayhew, and Dan Roth. 2020a. Extending multilingual BERT to low-resource languages. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2649–2656.

Zirui Wang, Jiateng Xie, Ruochen Xu, Yiming Yang, Graham Neubig, and Jaime G. Carbonell. 2020b. Cross-lingual alignment vs joint training: A comparative study and a simple unified framework. In *International Conference on Learning Representations*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

Shijie Wu and Mark Dredze. 2019. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. 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 833–844.

Shijie Wu and Mark Dredze. 2020. Do explicit alignments robustly improve multilingual encoders? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4471–4482.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011.

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The United Nations parallel corpus v1.0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534.

A Appendix

A.1 SMALA Implementation Details

To train subword embeddings we use FastText (Bojanowski et al., 2017) with dimension 1,024. Other than that, we use the default parameters, i.e. a window size of 5 and 10 negative examples. For the mapping of the embedding we use the unsupervised version of VecMap (Artetxe et al., 2018) with default hyperparameters. Indeed, we argue against identical subwords as cross-lingual anchors, and the unsupervised version takes advantage of similarity distributions of equivalent words in a way that matches our intuition. If parallel data is available, we use FastAlign (Dyer et al., 2013) with default hyperparameters. Our approach is not bound to these specific tools and can benefit from future research in the fields of (sub)word representation and (supervised or unsupervised) alignment.

A.2 Alignments Produced by SMALA

The number of alignments of SMALA depends on the language relatedness and the amount of monolingual and multilingual data. In Table 6 we present the number of subwords that are shared between languages for the first set of experiments (XNLI). We note that the maximum number of shared subwords is $30,522$ (the number of $\mathcal{L}_1$ subwords).

Method	Data	Es	De	El	Ru	Ar
JOINT	mono	7,936	7,554	3,395	2,836	2,909
OURS	mono	13,466	11,269	10,120	9,334	9,245
OURS	para	9,708	7,945	6,491	6,265	4,590
RAMEN	*	0	0	0	0	0

Table 6: Number of $\mathcal{L}_2$ subword embeddings that are shared with $\mathcal{L}_1$ for each system and language.

In Table 7 we present the number of shared subwords (anchors) for the ablation experiments of Section 7.1. The number of false positives identified by SMALA can be computed as the difference between the first and the second row, e.g. $7,780 - 4,374 = 3,406$ for Es. The number of false negatives can computed as the difference between the fourth and the second row, e.g. $13,466 - 4,374 = 9,092$ for Es. The difference between the number of false positives and the difference between the number of anchors of $-$FN and OURS reveals how many false positives are removed while removing false negatives, e.g. $3,406 - (15,269 - 13,466) = 1,603$ for Es.

For MT, we choose the number of monolingual vocabularies so that the merged vocabulary is equal in size to the one produced by joint tokenization. This leads to monolingual vocabularies of size 20k for En-De, 18.5k for En-Ru, 10k for En-Ro and 9k for En-Ar.

Method	Es	De	El	Ru	Ar
JOINT	7,780	7,395	3,283	2,685	2,743
$-$FP	4,374	3,838	285	286	230
$-$FN	15,269	13,189	11,727	10,826	10,770
OURS	13,466	11,269	10,120	9,334	9,245

Table 7: Number of shared subwords in the case of only false positives or only false negatives. OURS amounts to $-$FP$-$FN.

A.3 Scores on Validations Sets

Tables 8 and 9 present the results on the development sets for the two sets of experiments.

A.4 Model Training Details

The amount of shared subwords of Table 6 translates to fewer parameters in the first setting. For Spanish (Es), for example, the number of added parameters (on top of the 110M parameters of pretrained BERT) for OURS with mono is $(30,522 - 13,466) \times 768$ compared to $30,522 \times 768$ for RAMEN, where 768 is the dimension of the token embeddings.

We train the bilingual LMs of Section 5.1 on two GeForce GTX 1080 Ti GPUs for approximately 55 hours. We then fine-tune our models on one GPU for 8 hours, except for mBERT where we use two due to the increased number of parameters.

For MT, the Transformer model for the high-resource pairs has 60.6M parameters and for the medium and low-resource pairs 52.4M, due to the difference in vocabulary size. For these experiments, we train the high-resource models on the same two GPUs for 50 hours and the medium/low-resource ones for 20 hours.

A.5 Additional Results on BLI

Figure 3 presents results on BLI for all methods and both directions. We also include models that use alignment for the initialization of their embeddings (i.e OURS+ALIGN and RAMEN), but only compare methods that use monolingual data. The initialization of non-shared subwords further improves results, which is expected since it provides a cross-lingual signal for all subword representations.

Furthermore, RAMEN slightly outperforms OURS+ALIGN, which could be attributed to the larger number of parameters. Another reason could be the inductive bias of SMALA, which leads to retrieval of the aligned target (sub)word for a given source (sub)word, ignoring other possible translations. Although this might hurt cross-lingual representations if context is absent (i.e. subword embeddings), our results show that it improves performance for zero-shot cross-lingual transfer.

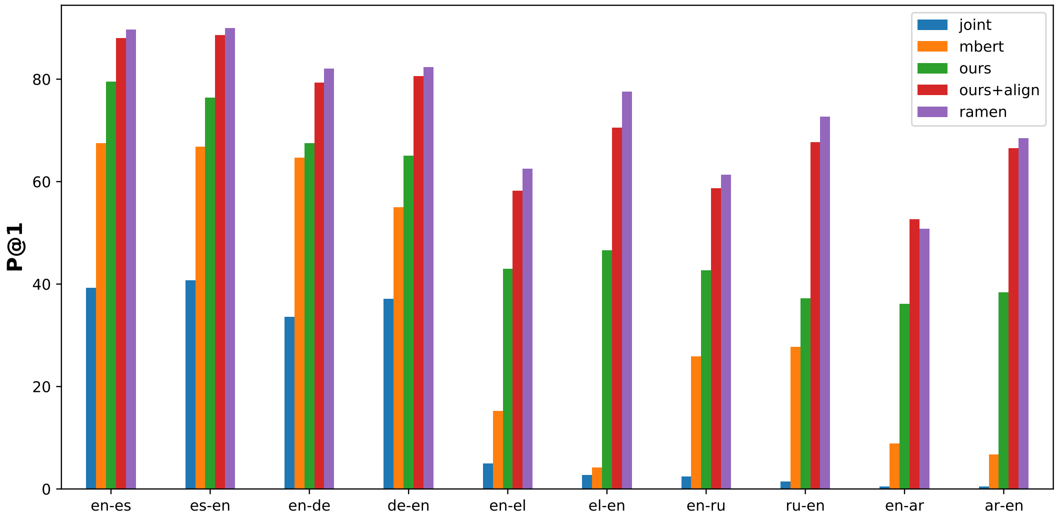

Figure 3: Precision@1 results for the BLI task.

Method	Data	Es	De	El	Ru	Ar
JOINT	mono	70.2 ± 1.2	64.5 ± 1.2	61.0 ± 0.9	56.3 ± 1.2	45.5 ± 0.5
OURS	mono	74.3 ± 0.5	69.6 ± 0.6	68.6 ± 0.9	65.5 ± 1.3	62.6 ± 0.3
OURS+ALIGN	mono	76.5 ± 0.4	71.9 ± 0.6	71.9 ± 0.5	68.9 ± 0.9	65.8 ± 0.2
OURS+ALIGN	para	76.5 ± 0.8	73.7 ± 0.6	75.3 ± 0.7	70.3 ± 0.8	66.9 ± 0.9
RAMEN	mono	75.5 ± 0.8	72.0 ± 1.3	72.2 ± 0.4	67.7 ± 0.9	64.9 ± 0.8
RAMEN	para	76.9 ± 0.8	73.9 ± 1.2	74.9 ± 0.9	69.7 ± 0.7	68.1 ± 1.3
mBERT	mono	74.6 ± 0.6	72.1 ± 0.7	66.3 ± 1.2	68.5 ± 1.0	62.9 ± 0.8

Table 8: Zero-shot classification scores on XNLI dev set (accuracy): mean and standard deviation over five runs are reported. Results follow the same format as those in Table 1.

	En-Ru		En-De		En-Ro		En-Ar	
	←	→	←	→	←	→	←	→
JOINT	30.0	27.8	34.6	31.7	33.1	26.5	33.1	15.5
OURS	30.2	28.3	34.6	31.6	33.0	26.1	31.8	15.5

Table 9: BLEU scores (detokenized) of baseline and our system for machine translation on the development set.

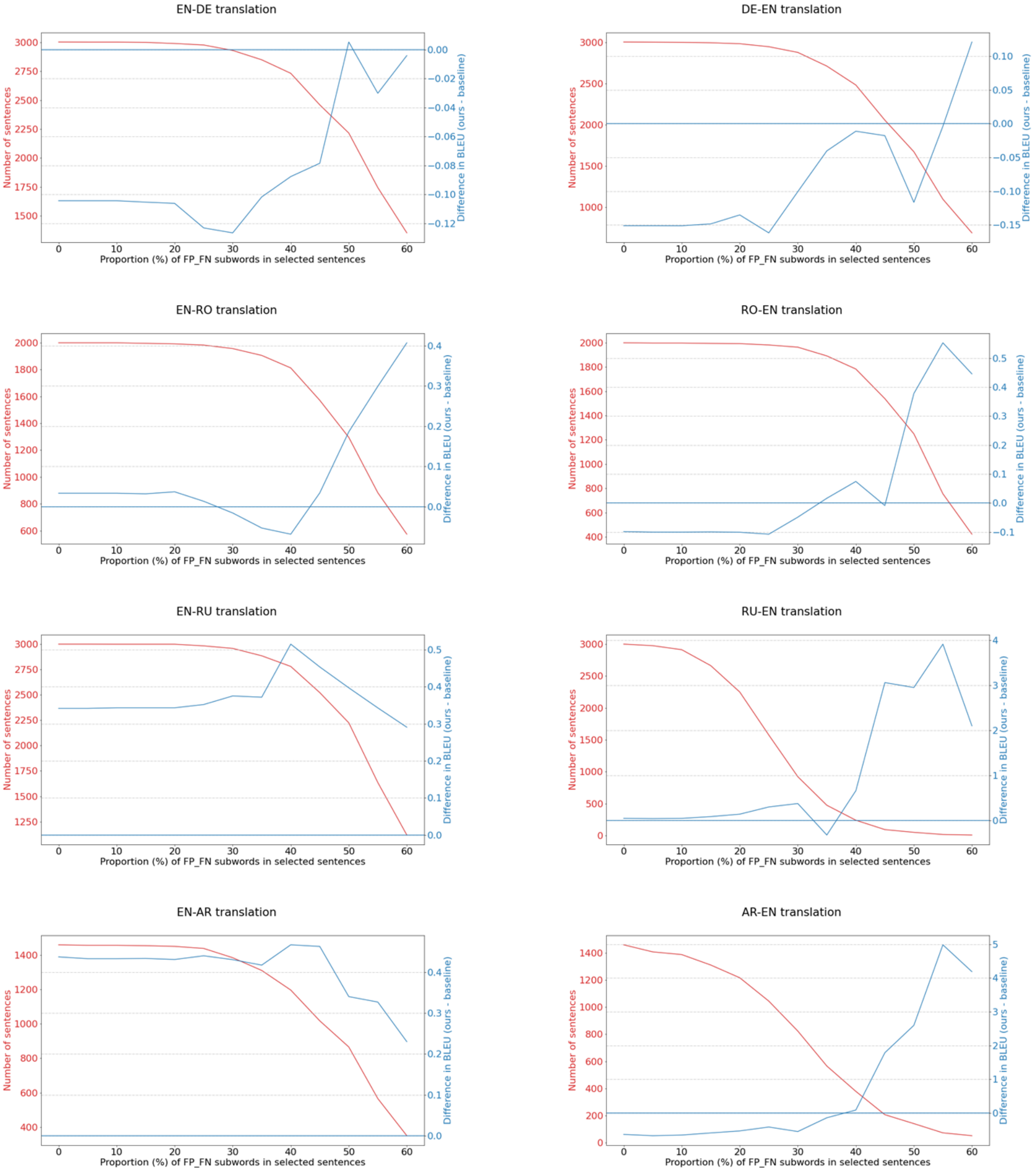

Figure 4: BLEU scores for sentences that contain a high percentage of false positives and/or false negatives.