Cross-lingual Transfer of Twitter Sentiment Models Using a Common Vector Space

Marko Robnik-Šikonja*, **Kristjan Reba***, **Igor Mozetič**†

* University of Ljubljana, Faculty of Computer and Information Science
Večna pot 113, SI-1000 Ljubljana, Slovenia
marko.robnik@fri.uni-lj.si   kr3377@student.uni-lj.si

† Jožef Stefan Institute
Jamova 39, SI-1000 Ljubljana
igor.mozetic@ijs.si

Abstract

Word embeddings represent words in a numeric space in such a way that semantic relations between words are encoded as distances and directions in the vector space. Cross-lingual word embeddings map the vector space of one language to the vector space of another language, or vector spaces of multiple languages to the joint vector space where similar words are aligned. Cross-lingual embeddings can be used to transfer machine learning models between languages, thereby compensating for insufficient data in less-resourced languages. We use cross-lingual word embeddings to transfer machine learning prediction models for Twitter sentiment between 13 languages. We focus on two transfer mechanisms using the joint numerical space for many languages as implemented in the LASER library: the transfer of trained models and the expansion of training sets with instances from other languages. Our experiments show that the transfer of models between similar languages is sensible, while dataset expansion did not increase the predictive performance.

1. Introduction

Word embeddings are representations of words in numerical form, as vectors of typically several hundred dimensions. The vectors are used as input to machine learning models; for complex language processing tasks, these generally are deep neural networks. The embedding vectors are obtained from specialized neural network-based embedding algorithms, e.g., word2vec (Mikolov et al., 2013), or fastText (Bojanowski et al., 2017). Modern word embedding spaces exhibit similar structures across languages, even when considering distant language pairs like English and Vietnamese (Mikolov et al., 2013). This means that embeddings independently produced from monolingual text resources can be aligned, resulting in a common cross-lingual representation, called cross-lingual embeddings, which allows for fast and effective integration of information in different languages.

There exist several approaches to cross-lingual embeddings. The first group of approaches uses monolingual embeddings with the optional help from a bilingual dictionary to align the pairs of embeddings (Artetxe et al., 2018a). The second group of approaches uses bilingually aligned (comparable or even parallel) corpora to construct joint embeddings in all the involved languages (Artetxe and Schwenk, 2019). The third type of approach is based on large pretrained multilingual masked language models such as BERT (Devlin et al., 2019). In this work, we focus on the second group of approaches, i.e. a joint sentence representation for many languages (Artetxe and Schwenk, 2019) as implemented in the LASER library[1] and available for 93 languages.

Sentiment annotation is a costly and lengthy operation, with relatively low inter-annotator agreement (Mozetič et al., 2016). Large annotated sentiment datasets are therefore rare, especially for low-resourced languages. The transfer of already trained models or datasets from other languages would be useful and would increase the ability to study sentiment-related phenomena for many more languages than possible today.

Using a collection of 13 large Twitter sentiment datasets, annotated in the same manner, we study two modes of cross-lingual transfer based on projections of sentences into a joint vector space. The first approach transfers trained models from source to target languages, where the model is trained on source language(s), and used for classification in the target language(s). This model transfer is possible because texts in all involved languages are embedded into the common vector space. The second approach expands the training set with instances from other languages, and then all instances are mapped into the common vector space during neural network training. Additionally, we analyse the quality of representations for the Twitter sentiment classification and compare the common vector space for several languages constructed by LASER library, multilingual BERT, and traditional bag-of-words approach. The results show a relatively low decrease in predictive performance when transferring trained sentiment prediction models between languages.

The paper is divided into four sections. In Section 2, we present background on different types of cross-lingual embeddings: alignment of monolingual embeddings, building a fixed common vector space for several languages, and multilingual contextual models. In Section 3, we present a large collection of tweets from 13 languages used in our empirical evaluation, the implementation details of our deep neural network prediction models, and the evaluation metrics used. Section 4 contains four series of experiments. We first analyse the transfer of trained models between lan-

[1] https://github.com/facebookresearch/LASER

guages from the same language group and from a different language group, followed by the expansion of datasets with instances from other languages. We end the experimental part with the evaluation of representation spaces and compare the common vector space with the multilingual BERT model. In Section 5, we summarise the results, draw the conclusions, and present ideas for further work.

2. Background

Word embeddings represent each word in a language as a vector in a high dimensional vector space so that the relations between words in a language are reflected in their corresponding embeddings. Cross-lingual embeddings attempt to map words represented as vectors from one vector space to the other so that the vectors representing words with the same meaning in both languages are as close as possible. Søgaard et al. (2019) present a detailed overview and classification of cross-lingual methods.

Cross-lingual approaches can be sorted into three groups, described in the following three subsections. The first group of methods uses monolingual embeddings with (an optional) help from bilingual dictionaries to align the embeddings. The second group of approaches uses bilingually aligned (comparable or even parallel) corpora for joint construction of embeddings in all involved languages. The third type of approach is based on large pretrained multilingual masked language models such as BERT (Devlin et al., 2019). The multilingual BERT is typically used as a starting model which is fine-tuned for a particular task, without explicitly extracting embedding vectors.

2.1. Alignment of monolingual embeddings

Cross-lingual alignment methods take precomputed word embeddings for each language and align them with optional use of bilingual dictionaries. Two types of monolingual embedding alignment methods exist. The first type of approaches map vectors representing words in one of the languages into the vector space of the other language (and vice-versa). The second type of approaches maps embeddings from both languages into a common vector space. The goal of both types of alignments is the same: the embeddings for words with the same meaning must be as close as possible in the final vector space. A comprehensive summary of existing approaches can be found in (Artetxe et al., 2018a). The open source implementation of the method described in (Artetxe et al., 2018a), named *vecmap*[2], is able to align monolingual embeddings either using supervised, semi-supervised, or unsupervised approach.

The supervised approach requires the use of a bilingual dictionary, which is used to match embeddings of equivalent words. The embeddings are aligned using the Moore-Penrose pseudo-inverse, which minimizes the sum of squared Euclidean distances. The algorithm always converges but can be caught in a local maximum when the initial solution is poor. To overcome this, several methods (stochastic dictionary introduction, frequency-based vocabulary cutoff, etc.) are used that help the algorithm to climb out of local maxima. A more detailed description of the algorithm is given in (Artetxe et al., 2018b).

The semi-supervised approach uses a small initial seeding dictionary, while the unsupervised approach is run without any bilingual information. The latter uses similarity matrices of both embeddings to build an initial dictionary. This initial dictionary is usually of low but sufficient quality for later processing. After the initial dictionary (either by seeding dictionary or using similarity matrices) is built, an iterative algorithm is applied. The algorithm first computes optimal mapping using the pseudo-inverse approach for the given initial dictionary. Then the optimal dictionary for the given embeddings is computed, and the procedure is repeated with the new dictionary.

When constructing mappings between embedding spaces, a bilingual dictionary can be helpful as its entries can be used as anchors for the alignment map for supervised and semi-supervised approaches. However, lately, researchers have proposed methods that do not require the use of a bilingual dictionary, but rely on adversarial approach (Conneau et al., 2018) or use the frequencies of the words (Artetxe et al., 2018b) to find a required transformation. These are called unsupervised approaches.

2.2. Projecting into a common vector space

To construct a common vector space for all involved languages, one requires a large aligned bilingual or multilingual parallel corpus. The constructed embeddings must map the same words in different languages as close as possible in the common vector space. The availability and quality of alignments in training set corpus may present an obstacle. While Wikipedia, subtitles, and translation memories are good sources of aligned texts for large languages, less-resourced languages are not well-presented and building embeddings for such languages is a challenge.

LASER (Language-Agnostic SEntence Representations) is a Facebook research project focusing on joint sentence representation for many languages (Artetxe and Schwenk, 2019). Similarly to machine translation architectures, it uses an encoder-decoder architecture. The encoder is trained on a large parallel corpus, translating a sentence in any language or script to a parallel sentence in either English or Spanish (whichever exists in the parallel corpus), thereby forming a joint representation of entire sentences in many languages in a shared vector space. The project focused on scaling to a large number of languages; currently, the encoder supports 93 different languages. Using LASER, one can train a classifier on data from just one language and use it on any language supported by LASER. A vector representation in the joint embedding space can be transformed back into a sentence using a decoder for the specific language.

2.3. Multilingual BERT

BERT (Bidirectional Encoder Representations from Transformers) embedding (Devlin et al., 2019) generalises the idea of language model (LM) to masked language models, inspired by the cloze test, which tests understanding of a text by removing a certain portion of words, which the participant is asked to replace. The masked language model randomly masks some of the tokens from the input, and the task of LM is to predict the missing token

[2] https://github.com/artetxem/vecmap

based on its neighbourhood. BERT uses transformer neural networks (Vaswani et al., 2017) in a bidirectional sense and further introduces the task of predicting whether two sentences appear in a sequence. The input representation of BERT are sequences of tokens representing sub-word units. The input is constructed by summing the embeddings of corresponding tokens, segments, and positions. Some widespread words are kept as single tokens, others are split into sub-words (e.g., frequent stems, prefixes, suffixes—if needed down to single letter tokens). The original BERT project offers pre-trained English, Chinese and multilingual model. The latter, called mBERT, is trained on 104 languages simultaneously.

To use BERT in classification tasks only requires adding connections between its last hidden layer and new neurons corresponding to the number of classes in the intended task. The fine-tuning process is applied to the whole network, and all of the parameters of BERT and new class-specific weights are fine-tuned jointly to maximise the log-probability of the correct labels.

3. Datasets and experimental settings

In this section, we present the evaluation metrics, experimental data and implementation details of the used neural prediction models.

3.1. Evaluation metrics

Following Mozetič et al. (2016) we report $\overline{F_1}$ score which takes positive and negative sentiment into account, and classification accuracy CA. $F_1(c)$ score for class value c is the harmonic mean of precision p and recall r for the given class c, where the precision is defined as the proportion of correctly classified instances from the instances predicted to be from the class c, and the recall is the proportion of correctly classified instances actually from the class c.

$$F_1(c) = \frac{2p_c r_c}{p_c + r_c}.$$

The F_1 score returns values from $[0, 1]$ interval, where 1 means perfect classification and 0 completely wrong predictions. We use F_1 score averaged over positive $(+)$ and negative $(-)$ sentiment class:

$$\overline{F_1} = \frac{F_1(+) + F_1(-)}{2}.$$

As the sentiment labels are ordered, the neutral sentiment label is implicitly taken into account in $\overline{F_1}$.

The classification accuracy CA is defined as the ratio of correctly predicted tweets N_c to all the tweets N:

$$CA = \frac{N_c}{N}$$

3.2. Datasets

We use a corpus of Twitter sentiment datasets (Mozetič et al., 2016), consisting of 15 languages, with over 1.6 million annotated tweets. The languages covered are Albanian, Bosnian, Bulgarian, Croatian, English, German, Hungarian, Polish, Portuguese, Russian, Serbian, Slovak, Slovene, Spanish, and Swedish. Authors studied the annotators' agreement on the labelled tweets. They discovered that for some languages (English, Russian, Slovak) the SVM classifier achieves significantly lower score than the annotators. This hints that for these languages, there might be a room for improvement using better classification model or larger training set.

We cleaned the above datasets by removing the duplicated tweets, weblinks, and hashtags. Due to the low quality of sentiment annotations indicated by low self-agreement and low inter-annotator agreement, we removed Albanian and Spanish datasets. For these two languages, the self-agreement expressed with $\overline{F_1}$ score (i.e. $F_1(c)$ is the fraction of equally labelled tweets out of all the tweets with a given label c) is 0.60 and 0.49, respectively; the inter-annotator agreement is 0.41 and 0.42. The characteristics of the remaining 13 datasets are presented in Table 1.

3.3. Implementation details

In our experiments, we use two different types of prediction models, BiLSTM neural networks using joint vector space embeddings constructed with the LASER library, and multilingual BERT. The multilingual BERT model is case sensitive (i.e. bert_multi_cased), pretrained on 104 languages, has 12 transformer layers, and 110 million parameters. We fine-tune only the last layer of the network, using the batch size of 32, and 3 epochs.

The cross-lingual embeddings from LASER library are pretrained on 93 languages, using BiLSTM networks, and are stored as 1024 dimensional embedding vectors. Our classification models contain the embedding layer, followed by multilayer perceptron hidden layer of size 8, and an output layer with three neurons (corresponding to three output classes, negative, neutral, and positive sentiment) using the softmax. We use ReLU activation function and Adam optimizer. The fine-tuning uses a batch size of 32 and 10 epochs.

Further technical details are available in the freely available source code.

4. Experiments and results

Our experimental evaluation focuses on text representation using embeddings into a common vector space with the LASER library. We conducted several experiments reported below: transfer of models between languages from the same and different language family, expansion of training sets with varying amounts of data from other languages, and comparison of the joint space embeddings with the multilingual BERT. We did not systematically test all possible combinations of languages and language groups as this would require too much computational time and results would not fit into the paper. Instead, we arbitrary selected a few language combinations in advance. We leave a more systematic approach based on informative features (Lin et al., 2019) for further work.

4.1. Transfer to the same language family

We first test the transfer of prediction models between similar languages from the same language family. The transfer between similar languages is the most likely to be successful. As the source and target languages, we tried

	Number of tweets				Agreement	
Language	Negative	Neutral	Positive	All	Self-	Inter-
Bosnian	12,868	11,526	13,711	38,105	0.81	0.51
Bulgarian	15,140	31,214	20,815	67,169	0.77	0.50
Croatian	21,068	19,039	43,894	84,001	0.81	0.51
English	26,674	46,972	29,388	103,034	0.79	0.67
German	20,617	60,061	28,452	109,130	0.73	0.42
Hungarian	10,770	22,359	35,376	68,505	0.76	-
Polish	67,083	60,486	96,005	223,574	0.84	0.67
Portuguese	58,592	53,820	44,981	157,393	0.74	-
Russian	34,252	44,044	29,477	107,773	0.82	-
Serbian	24,860	30,700	16,161	71,721	0.81	0.51
Slovak	18,716	14,917	36,792	70,425	0.77	-
Slovene	38,975	60,679	34,281	133,935	0.73	0.54
Sweedish	25,319	17,857	15,371	58,547	0.76	-

Table 1: The left-hand side reports the number of tweets from each of the category and the overall number of instances for individual languages. The right-hand side contains self-agreement of annotators, and inter-annotator agreement for languages where more than one annotator was involved.

		Transfer		Both target	
Source	Target	$\overline{F_1}$	CA	$\overline{F_1}$	CA
German	English	0.55	0.59	0.62	0.65
Polish	Russian	0.64	0.59	0.70	0.70
Polish	Slovak	0.63	0.59	0.72	0.72
German	Swedish	0.58	0.57	0.67	0.65
German Swedish	English	0.58	0.60	0.62	0.65
Slovene Serbian	Russian	0.53	0.55	0.70	0.70
Slovene Serbian	Slovak	0.59	0.52	0.72	0.72
Average performance gap		0.09	0.11		

Table 2: The transfer of trained models between languages from the same language family using common vector space (left-hand side) and comparison with both training and testing set from the target language (on the right-hand side).

several combinations of Slavic and Germanic languages. We report the results in Table 2.

In each experiment, we use the entire dataset(s) of the source language as the training set, and the whole dataset of the target language as the testing set. We compare the results with the training and testing set from the target language, where 70% of the dataset is used for training and 30% for testing. The latter results can be taken as an upper bound of what the transfer models could achieve in an ideal condition.

The results from Table 2 show that there is a gap between transfer learning models and native models from 4% to 20% (on average the decrease in performance for transfer learning is 9.3% for $\overline{F_1}$ and 11.1% for CA). For the direct transfer of models without additional target data, these results are encouraging.

4.2. Transfer to different language family

We repeat the experiments we did for languages from the same language family on languages from different language families. The transfer is less likely to be successful in this case, and we expect a lower performance in these unfavourable conditions.

The results from Table 3 show that there is a gap between transferred models and native models from 4% to 28% (on average the decrease of performance for transfer learning is 14% for $\overline{F_1}$ and 15.2% for CA). This gap is significant and makes the resulting transferred models less useful in the target languages. Another observation is that the differences between target languages are significant. It seems that the transfer to Slovak is much less successful than to Russian, while English is in between the two.

		Transfer		Both target	
Source	Target	$\overline{F_1}$	CA	$\overline{F_1}$	CA
Russian	English	0.52	0.56	0.62	0.65
English	Russian	0.57	0.58	0.70	0.70
English	Slovak	0.46	0.44	0.72	0.72
Polish, Slovene	English	0.58	0.57	0.62	0.65
German, Swedish	Russian	0.61	0.61	0.70	0.70
English, German	Slovak	0.50	0.47	0.72	0.72
Average performance gap		0.14	0.15		

Table 3: The transfer of trained models between languages from different language groups using the common vector space representation (left-hand side), and comparison with both training and testing set from the target language (on the right-hand side).

4.3. Increasing datasets with several languages

We test possible improvements in prediction performance if we increase the training sets with instances from several related and unrelated languages. The training set in each experiment consists of instances from several languages projected into the common vector space and also 70% of the target language dataset. The remaining 30% of target language instances are used as the testing set. As the text representation, we use projection into the common vector space computed with the LASER library.

The results from Table 4 show a gap between learning models using the expanded datasets and native models (the decrease for expanded models is from 2% to 7%, on average 3% for F_1 and 5.7% for CA). These results indicate that the tested expansion of datasets was unsuccessful, i.e. the provided amount of instances from the target language was

		Expanded		Only target	
Source	Target	$\overline{F_1}$	CA	$\overline{F_1}$	CA
English, Croatian, Slovene	Slovene	0.58	0.53	0.60	0.60
English, Croatian, Serbian, Hungarian, Slovak	Slovak	0.67	0.65	0.72	0.72
English, Croatian, Russian	Russian	0.67	0.65	0.70	0.70
Russian, Swedish, English	English	0.60	0.61	0.62	0.65
Average improvement		-0.03	-0.06		

Table 4: The expansion of training sets with instances from several languages projected into the common vector space using the LASER library (left-hand side) and comparison with training and testing set from the same language (on the right-hand side).

	All other & Target		Only Target	
Target	$\overline{F_1}$	CA	$\overline{F_1}$	CA
Bosnian	0.64	0.59	0.67	0.64
Bulgarian	0.54	0.56	0.50	0.59
Croatian	0.63	0.57	0.73	0.68
English	0.58	0.60	0.62	0.65
German	0.52	0.59	0.53	0.65
Hungarian	0.59	0.61	0.60	0.67
Polish	0.67	0.63	0.70	0.66
Portugal	0.44	0.39	0.52	0.51
Russian	0.66	0.64	0.70	0.70
Serbian	0.52	0.49	0.48	0.54
Slovak	0.64	0.61	0.72	0.72
Slovene	0.54	0.50	0.60	0.60
Swedish	0.63	0.59	0.67	0.65
Average improvement	-0.04	-0.07		

Table 5: The expansion of training sets with instances from all other languages mapped into the common vector space using the LASER library (left-hand side) and comparison with training and testing set from the same language (on the right-hand side).

already sufficient for successful learning. The additional instances from other languages are likely to be of lower quality than the native instances and therefore decrease the performance.

To test an even more extensive expansion of the training sets, we trained models on all other languages and 70% of the target language, while testing them on the remaining 30% of the target language. The results are presented in Table 5.

The results show that using many languages and significant enlargement of datasets can be successful. For Bulgarian and Serbian training on many languages gives higher $\overline{F_1}$ score (but not CA) than training only on the target language. For all other languages, the tried expansions of training sets are unsuccessful, and the difference to native models is on average 3.5% for $\overline{F_1}$ score and 6.8% for CA.

4.4. Comparing embeddings

In our final experiment, we compare embeddings into a common vector space obtained with LASER library with the multilingual BERT. Note that in this experiment, there is no transfer between different languages but only a test of the quality of the representation, i.e. embeddings. The training set in each experiment consists of randomly chosen 70% of the dataset for each language, while the remaining 30% of instances are used as the testing set. As a baseline, we report the results of the SVM model without neural embeddings that uses Delta TF-IDF weighted bag-of-ngrams representation with substantial preprocessing of tweets (Mozetič et al., 2016). These results are not entirely comparable with our setting as they were obtained with 10-fold stratified blocked cross-validation, while we use a single 70:30 split. Further, the datasets for Bosnian, Croatian, and Serbian language were merged in (Mozetič et al., 2016) due to the similarity of these languages; therefore, we report the performance on the merged dataset for the SVM classifier. Results are presented in Table 6.

	LASER		mBERT		SVM	
Language	$\overline{F_1}$	CA	$\overline{F_1}$	CA	$\overline{F_1}$	CA
Bosnian	**0.67**	0.64	0.65	**0.66**	0.61	0.56
Bulgarian	0.50	0.59	**0.58**	**0.60**	0.52	0.54
Croatian	**0.73**	**0.68**	0.64	**0.68**	0.61	0.56
English	0.62	0.65	**0.72**	**0.71**	0.63	0.64
German	0.53	0.65	**0.66**	**0.66**	0.54	0.61
Hungarian	0.60	0.67	**0.65**	**0.69**	0.64	0.67
Polish	**0.70**	0.66	**0.70**	**0.73**	0.68	0.63
Portugal	0.52	0.51	**0.66**	**0.67**	0.55	0.51
Russian	0.70	0.70	**0.74**	**0.75**	0.61	0.60
Serbian	0.48	0.54	0.56	0.54	**0.61**	**0.56**
Slovak	**0.72**	0.72	0.70	**0.75**	0.68	0.68
Slovene	0.60	0.60	**0.66**	**0.64**	0.55	0.54
Swedish	**0.67**	0.65	0.64	**0.66**	0.66	0.62
Average	0.62	0.64	**0.66**	**0.67**	0.61	0.59

Table 6: Comparison of different representations: supervised mapping into a common vector space with the LASER library, multilingual BERT, and bag-of-ngrams with the SVM classifier. The best score for each language and metric is in bold.

The SVM baseline using bag-of-words representation achieves lower predictive performance than the two neural embedding approaches. We speculate that the main reason is the knowledge about language structure contained in large precomputed embeddings used by the neural approaches. Together with the fact that standard feature-based machine learning approaches require much more preprocessing effort, it seems that there are no good reasons why to bother with this approach in text classification. The multilingual BERT is the best of the three tested methods, achieving the best average $\overline{F_1}$ and CA scores, as well as the best result in most languages (in bold). The downside is that the fine-tuning and execution of mBERT requires much more computational time compared to precomputed fixed embeddings. Nevertheless, with progress in optimization techniques for neural network learning and advent of computationally more efficient BERT variants, e.g., (You et al., 2020), this obstacle might disappear in the future.

5. Conclusions

We studied two approaches to the cross-lingual transfer of Twitter sentiment prediction models based on mappings of words into the common vector space: transfer of trained models, and expansion of datasets with instances from other languages. Our empirical evaluation is based on relatively large datasets of labelled tweets from 13 European languages. As word representations, we used mappings into a common vector space produced by the LASER library. The results show that there is a significant transfer potential using the models trained on similar languages; compared to training and testing on the same language, we get on average 9.3% lower $\overline{F_1}$ score and 11.1% lower CA. Using models trained on languages from different language families produces larger differences (on average 14% for $\overline{F_1}$ and 15.2% for CA). Our attempt to expand training sets with instances from different languages was unsuccessful using either additional instances from a small group of languages or instances from all other languages. Finally, we tested the quality of text representations by comparing cross-lingual joint embedding space of LASER library, multilingual BERT embeddings, and classical bag-of-ngram representation coupled with SVM classifier. The results show that the multilingual BERT is the most successful of the three, followed by the common vector space of LASER library, while bag-of-ngrams is rarely competitive. The source code of our analyses is freely available[3].

In future work, we plan to expand the experiments with other embedding techniques, in particular, the ELMo contextual embeddings (Peters et al., 2018) together with non-isomorphic cross-lingual transformations that could produce better representations in the joint vector spaces.

Acknowledgements

The research was supported by the Slovene Research Agency through research core funding no. P6-0411 and P2-103. This paper is supported by European Union's Horizon 2020 Programme project EMBEDDIA (Cross-Lingual Embeddings for Less-Represented Languages in European News Media, grant no. 825153), and Rights, Equality and Citizenship Programme project IMSyPP (Innovative Monitoring Systems and Prevention Policies of Online Hate Speech, grant no. 875263). The results of this publication reflects only the author's view and the Commission is not responsible for any use that may be made of the information it contains.

6. References

Mikel Artetxe and Holger Schwenk. 2019. Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond. *Transactions of the Association for Computational Linguistics*, 7:597–610.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018a. Generalizing and improving bilingual word embedding mappings with a multi-step framework of linear transformations. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2018b. 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.

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.

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

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.

Yu-Hsiang Lin, Chian-Yu Chen, Jean Lee, Zirui Li, Yuyan Zhang, Mengzhou Xia, Shruti Rijhwani, Junxian He, Zhisong Zhang, Xuezhe Ma, et al. 2019. Choosing transfer languages for cross-lingual learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3125–3135.

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

Igor Mozetič, Miha Grčar, and Jasmina Smailović. 2016. Multilingual Twitter sentiment classification: The role of human annotators. *PLOS ONE*, 11(5).

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

Anders Søgaard, Ivan Vulić, Sebastian Ruder, and Manaal Faruqui. 2019. *Cross-Lingual Word Embeddings*. Morgan & Claypool Publishers.

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*, pages 5998–6008.

Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. 2020. Large batch optimization for deep learning: Training BERT in 76 minutes. In *International Conference on Learning Representations, ICLR 2019*.

[3] https://github.com/kristjanreba/cross-lingual-classification-of-tweet-sentiment