Cross-Lingual Transfer Learning for Multilingual Task Oriented Dialog

Sebastian Schuster*
Stanford Linguistics
sebschu@stanford.edu

Sonal Gupta
Facebook Conversational AI
sonalgupta@fb.com

Rushin Shah
Facebook Conversational AI
rushinshah@fb.com

Mike Lewis
Facebook AI Research
mikelewis@fb.com

Abstract

One of the first steps in the utterance interpretation pipeline of many task-oriented conversational AI systems is to identify user intents and the corresponding slots. Since data collection for machine learning models for this task is time-consuming, it is desirable to make use of existing data in a high-resource language to train models in low-resource languages. However, development of such models has largely been hindered by the lack of multilingual training data. In this paper, we present a new data set of 57k annotated utterances in English (43k), Spanish (8.6k) and Thai (5k) across the domains weather, alarm, and reminder. We use this data set to evaluate three different cross-lingual transfer methods: (1) translating the training data, (2) using cross-lingual pre-trained embeddings, and (3) a novel method of using a multilingual machine translation encoder as contextual word representations. We find that given several hundred training examples in the the target language, the latter two methods outperform translating the training data. Further, in very low-resource settings, multilingual contextual word representations give better results than using cross-lingual static embeddings. We also compare the cross-lingual methods to using monolingual resources in the form of contextual ELMo representations and find that given just small amounts of target language data, this method outperforms all cross-lingual methods, which highlights the need for more sophisticated cross-lingual methods.

1 Introduction

One of the first steps in many conversational AI systems that are used to parse utterances in personal assistants is the identification of what the user intends to do (the *intent*) as well as the arguments of the intent (the *slots*) (Mesnil et al., 2013; Liu and Lane, 2016). For example, for a request such as *Set an alarm for tomorrow at 7am*, a first step in fulfilling such a request is to identify that the user's intent is to set an alarm and that the required time argument of the request is expressed by the phrase *tomorrow at 7am*.

Given these properties of the task, the problem can be stated as a joint sentence classification (for intent classification) and sequence labeling (for slot detection) task and therefore naturally lends itself to using a biLSTM-CRF sequence labeling model (Lample et al., 2016; Vu, 2016) where the biLSTM layer is also used as the input for a projection layer for intent detection.

These models are very powerful and given enough training data, they achieve very high accuracy on the intent classification as well as the slot detection task. However, given the requirement of large amounts of labeled training data, developing a conversational AI system for many new languages is a very resource-intensive task and clearly not feasible to be done for the more than 6,500 languages that are currently spoken around the world.

For this reason, one would like to make use of methods that enable transfer learning from a high-resource language to a low-resource language. However, the development of sophisticated cross-lingual transfer methods for intent detection and slot filling has so far been hindered by the lack of multilingual data sets that have been annotated according to the same guidelines.[1] In this work, we therefore present a novel data set containing a large number of English utterances (the high-resource data) as well as a smaller set of utterances in Spanish and in Thai (the low-resource data), which were annotated according to the same annotation scheme. This data allows the systematic in-

* Work carried out during an internship at Facebook.

[1] Upadhyay et al. (2018) collected such a dataset but to the best of our knowledge, their data is not publicly available.

Proceedings of NAACL-HLT 2019, pages 3795–3805
Minneapolis, Minnesota, June 2 - June 7, 2019.

Domain	Number of utterances			Intent types	Slot types
	English	Spanish	Thai		
Alarm	9,282/1,309/2,621	1,184/691/1,011	777/439/597	6	2
Reminder	6,900/943/1,960	1,207/647/1,005	578/336/442	3	6
Weather	14,339/1,929/4,040	1,226/645/1,027	801/460/653	3	5
Total	30,521/4,181/8,621	3,617/1,983/3,043	2,156/1,235/1,692	12	11

Table 1: Summary statistics of the data set. The three values for the number of utterances correspond to the number of utterances in the training, development, and test splits. Note that the slot type *datetime* is shared across all three domains and therefore the total number of slot types is only 11.

vestigation of cross-lingual transfer learning methods from high-resource languages to low-resource languages.

We use this data set to explore different strategies to make use of training data from a high-resource language to improve intent and slot detection models for other languages. We investigate two previously proposed strategies for cross-lingual transfer, namely using cross-lingual pre-trained embeddings (XLU embeddings; see Ruder et al., 2017 for a review) as well as automatically translating the English training data to the target language. Further, we present a novel technique that uses a bidirectional neural machine translation encoder as cross-lingual contextual word representations. We compare the cross-lingual transfer methods to models that are only trained on the target language data.

Across the two languages and the various transfer methods, we find that joint training on the high-resource and the low-resource target language improves results on the target language. We further find that the optimal choice of transfer method depends on the size of the available training data in the target language: In the zero-shot case where no target language data is available, translating the training data gives the best results. However, if a small amount of training data is available, we find that jointly training on the high-resource and low-resource data works better than training on translated data.

We release the data at `https://fb.me/multilingual_task_oriented_data`.

2 Data

We originally collected a data set of around 43,000 English utterances across the domains ALARM, REMINDER, and WEATHER. Data collection proceeded in three steps. First, native English speakers were asked to produce utterances for each intent, e.g., provide examples of how they would ask for the weather. In a second step, two annotators would label the intent and the spans corresponding to slots for each utterance. As a third step, if annotators disagreed on the annotation of an utterance, a third annotator who corresponded with the authors of the guidelines adjudicated between the two annotations.

For the Spanish and Thai data, native speakers of the target language translated a sample of the English utterances. These translated utterances were then also annotated by two annotators. For Spanish, if annotators disagreed, a third annotator who was bilingual in Spanish and English adjudicated these disagreements in communication with the guideline authors. Unfortunately, for Thai, we did not have a bilingual speaker available and hence we decided to discard all utterances for which the annotators disagreed. We hope to rectify this for future data collection efforts.

We believe this data presents a great opportunity to investigate cross-lingual semantic models and to the best of our knowledge, this is the first parallel data set for a word tagging task that has been annotated according to the same guidelines across multiple languages.

Table 1 contains several summary statistics of the data set. Note that the percentage of training examples as compared to development and test examples is much higher for the English data than for the Thai and Spanish data. We decided for a more even split for the latter two languages so that we had a sufficiently large data set for model selection and evaluation.

3 NLU models

The intent detection and slot-filling model consists of two parts: It first uses a sentence classification model to identify the domain of the user utterance (in our case, ALARM, REMINDER, or WEATHER), and then uses a domain-specific model to jointly predict the intent and slots. Figure 1 shows the ba-

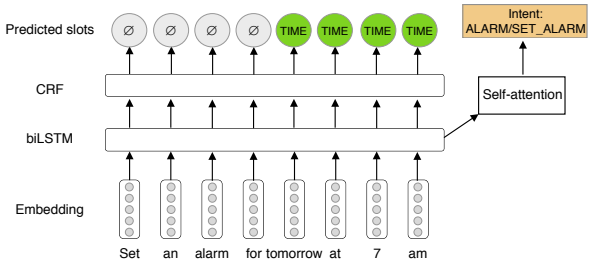

Figure 1: Slot and intent model architecture. Word embeddings are passed through a biLSTM layer which is shared across the slot detection and intent prediction tasks.

sic architecture of the joint intent-slot prediction model. It first embeds the utterance using an embedding matrix and then passes the word vectors to a biLSTM layer. For intent classification, we use a self-attention layer (Lin et al., 2017) over the hidden states of the biLSTM as input to a softmax projection layer; for slot detection, we pass for each word the concatenation of the forward and backward hidden states through a softmax layer, and then pass the resulting label probability vectors through a CRF layer for final predictions.

In our experiments, we vary how the tokens are embedded:

- **Zero embeddings**: We train the parameters of a 0-initialized embedding matrix that contains each word that appears in the training data.

- **XLU embeddings**: We embed the tokens through lookup in a pre-trained cross-lingual embedding matrix and concatenate these embeddings with tuned zero embeddings. Here, we follow Dozat et al. (2017) in having a fixed pre-trained embedding matrix combined with tuneable zero-embeddings.

- **Encoder embeddings**: We embed tokens by passing the entire utterance through a pre-trained biLSTM sentence encoder and using the hidden states of the top layer as input. We keep the parameters of the pre-trained encoder fixed and concatenate them with tuneable zero-embeddings. (See Section 4 for a detailed description of the different encoders.)

4 Encoder models

As mentioned in the previous section, some of our models use a pre-trained biLSTM encoder to generate contextual word embeddings. In all our experiments, we use a bidirectional LSTM encoder with two layers. Overall, we compare three strategies for training these encoders. The motivation for choosing these strategies is to investigate whether there is a benefit of using multilingual embeddings.

- **CoVe**: Following McCann et al. (2017), we train a neural machine translation model to translate from the low-resource language (Spanish or Thai) to English.

- **Multilingual CoVe**: We train a neural machine translation model to translate from the low-resource language to English and from English to the low-resource language. We encode the translation direction using target language-specific start tokens in the decoder (Yu et al., 2018a). In this model, the encoder does not have access to the target language and therefore we expect it to learn to encode phrases with similar meanings into similar vector spaces across languages.

- **Multilingual CoVe w/ autoencoder:** We train a bidirectional neural machine translation model and combine it with an autoencoder objective. For the language pair Spanish-English, that means given a Spanish input sentence we train the model to generate either an English translation or to reproduce the Spanish sentence depending on the start token in the decoder. Likewise, given an English sentence, we train the model to output either a Spanish translation or to reproduce the English sentence depending on the start token in the decoder. The motivation for this approach is that using the joint translation and autoencoder objective might lead to more general representations since the decoder has to be capable to output sentences in either language independent of what the source language was, and unlike in the previous model the source language does not determine the target language. We train an analogous model for the Thai-English language pair.

Spanish	Epoch	es→en	en→es	es→es	en→en
CoVe (unidirectional)	81	8.50	-	-	-
Mult. CoVe	98	8.27	6.90	-	-
Mult. CoVe + autoencoder	282	9.15	7.29	1.15	1.14
Thai	Epoch	th→en	en→th	th→th	en→en
CoVe (unidirectional)	12	13.06	-	-	-
Mult. CoVe	35	12.73	17.00	-	-
Mult. CoVe + autoencoder	92	11.76	16.31	1.12	1.13

Table 2: Perplexities on validation set for different encoder models for the Spanish-English and Thai-English language pairs. A hyphen means that an encoder was not trained for the corresponding language pair.

For Spanish, for which pre-trained ELMo (Peters et al., 2018) embeddings are available, we also evaluate the use of the ELMo embeddings by Che et al. (2018); Fares et al. (2017). Note that the ELMo encoder and the CoVe encoder are trained to encode only the low-resource language and therefore neither of them are multilingual encoders.

Implementation details We train all models using a wrapper around the fairseq (Gehring et al., 2016, 2017) sequence-to-sequence models. We use 300d randomly initialized word vectors as input to the first embedding layer. Each direction in each hidden layer has 512 dimensions which results in a total encoder output dimension of 1024. For the machine translation models, we further use dot-product attention (Luong et al., 2015) and to improve efficiency, we limit the output space of the softmax to 30 translation candidates as determined by word alignments as well as the 2,000 most frequent words (L'Hostis et al., 2016).

Data For the Spanish models, we use two copies[2] of Europarl v7 (Koehn, 2005), every eighth sentences of the Paracrawl data,[3] and the newstest2008-2011 data. For model selection, we use the newstest2012-2013 data. For the Thai models, we use 10 copies of the IWSTL training data (Cettolo et al., 2012) as well as the OpenSubtitles data (Lison et al., 2018) for training, and the IWSTL development and test data for model selection. We use the 20,000 most common words in the training data as the vocabulary. For the multilingual models, we take the union of the vocabulary from both languages. We tokenize the data using an in-house rule-based (for English and Spanish) and dictionary-based (for Thai) tokenizer. We further lowercase all data and remove all duplicates within a data set. We discard all sentences whose length exceeds 100 tokens.

Training details We train the models using stochastic gradient descent with an initial learning rate of 0.5. We decrease the learning rate by 1% after an epoch whenever perplexity on the validation data is higher than for the epoch with the lowest perplexity. We train all models for up to 100 epochs, except for the Spanish bidirectional MT model with an autoencoder which we trained for 300 epochs since it took considerably longer to converge. For multilingual models, we choose the model that has the lowest average perplexity on both translation tasks.

Table 2 shows the perplexities for the different models. In general, the translation perplexities are very similar independent of whether we train a unidirectional MT system or a bidirectional MT system, except for the Spanish bidirectional MT model with an autoencoder which even after 300 epochs still yields higher perplexities on the validation data than the other translation models.[4]

5 Cross-lingual learning

In our first set of experiments, we explore the following baselines and strategies for training models in Spanish and Thai given a large amount of English training data and a small amount of Spanish and Thai training data.

- **Target only**: Using only the low-resource target language data.

[2]We upsample the Europarl (for Spanish) and IWSLT (for Thai) data since these data sets are presumably of higher quality than the largerly automatically mined Paracrawl and OpenSubtitles data.

[3]https://paracrawl.eu, the version that was used in the WMT 2018 task

[4]We hypothesize that the slow convergence as well as the lower performance might be caused by the fact that the sentences in the Spanish-English parallel data are much longer than in the Thai-English data which might make it harder to learn good universal sentence representations.

English	Embedding type	Exact match	Domain acc.	Intent acc.	Slot F1
Target only	-	90.91	-	99.11	94.81
Spanish	Embedding type	Exact match	Domain acc.	Intent acc.	Slot F1
Target only	-	72.94	99.43	97.26	80.95
Target only	XLU embeddings	72.90	99.47	96.90	80.99
Target only	CoVe	73.93	99.52	97.43	81.51
Target only	Mult. CoVe	74.13	99.55	97.61	81.64
Target only	Mult. CoVe + auto	73.05	99.51	97.13	81.22
Target only	ELMo	74.81	99.53	96.64	82.96
Translate train	-	72.49	**99.65**	**98.47**	80.60
Cross-lingual	XLU embeddings	75.39	99.52	97.68	83.00
Cross-lingual	CoVe	75.17	99.55	97.81	82.55
Cross-lingual	Mult. CoVe	75.20	99.56	97.82	82.49
Cross-lingual	Mult. CoVe + auto	74.68	99.59	97.90	82.13
Cross-lingual	ELMo	**75.96**	99.47	97.51	**83.38**
Thai	Embedding type	Exact match	Domain acc.	Intent acc.	Slot F1
Target only	-	79.80	99.31	95.13	87.26
Target only	CoVe	84.84	99.36	96.60	90.63
Target only	Mult. CoVe	84.66	99.37	96.75	90.20
Target only	Mult. CoVe + auto	84.79	**99.41**	96.59	90.51
Translate train	-	73.37	99.37	**97.41**	80.38
Cross-lingual	CoVe	84.49	99.29	96.87	90.60
Cross-lingual	Mult. CoVe	85.76	99.39	96.98	91.22
Cross-lingual	Mult. CoVe + auto	**86.12**	99.33	96.87	**91.51**

Table 3: Results using the full training data averaged over 5 training runs. The *translate train* models are trained on the union of translated English and target language data; the *cross-lingual* models are trained on English and target language data.

- **Target only with encoder embeddings**: Using only the low-resource language training data and using pre-trained encoder embeddings.
- **Translate train**: Combining the target training data with the English data that has been automatically translated to the target language. The slot annotations are projected via the attention weights (Yarowsky et al., 2001). We translate the data using the Facebook neural machine translation system.
- **Cross-lingual with XLU embeddings**: Joint training on the English and target language data with pre-trained MUSE (Conneau et al., 2017) cross-lingual embeddings. Since MUSE embeddings are not available for Thai, we only evaluate this method for Spanish.
- **Cross-lingual with encoder embeddings**: Joint training on the English and target language data using pre-trained encoder embeddings.

Implementation details We implement all classification and sequence labeling models within the PyText framework (Aly et al., 2018). We train models for 20 epochs and select the model that performs best on the development set. We use the Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.01. We use dropout of 0.3 in the BiLSTM and we set the size of the self-attention layer to 128 dimensions.

Evaluation We evaluate our models according to four metrics: Domain accuracy, which measures the accuracy of the domain classification task; intent accuracy, which measures the accuracy of identifying the correct intent; slot F1, which is the geometric mean of the slot precision and slot recall; and the exact match metric, which indicates the number of utterances for which the domain, intent, and all slots were correctly identified. Exact match is thus the strictest metric of all. We micro-average all metrics across domains.

Results and discussion Table 3 shows the results for all evaluated models. While we get

Spanish	Embedding type	Exact match	Domain acc.	Intent acc.	Slot F1
Translate train	-	**54.95**	**88.70**	**85.39**	**72.87**
Cross-lingual	-	0.63	37.74	36.17	5.50
Cross-lingual	XLU embeddings	4.01	38.24	36.94	17.50
Cross-lingual	CoVe	1.37	39.42	37.13	5.35
Cross-lingual	mult. CoVe	10.56	59.29	53.34	22.50
Cross-lingual	mult. CoVe + auto	9.28	59.25	53.89	19.25
Cross-lingual	ELMo	0.18	35.98	35.36	2.53
Thai	Embedding type	Exact match	Domain acc.	Intent acc.	Slot F1
Translate train	-	**45.59**	**98.11**	**95.85**	**55.43**
Cross-lingual	-	0.20	39.36	39.11	3.44
Cross-lingual	CoVe	5.82	66.75	54.24	8.84
Cross-lingual	mult. CoVe	15.37	73.84	66.35	32.52
Cross-lingual	mult. CoVe + auto	20.84	81.95	70.70	35.62

Table 4: Zero-shot results averaged over 5 training runs. All models were trained only on the English data. In the case of the *translate train* models, the English data was automatically translated into the target language.

slightly different results for the two languages, there are several consistent patterns. For Spanish, we observe that adding contextual word representations to the *target only* model, consistently improves results. Not surprisingly since the ELMo embeddings were trained on a large monolingual corpus, the model that uses these embeddings outperforms all other *target only* models.

If we turn to the cross-lingual models for Spanish, the results indicate that the translation method works well for domain and intent classification but less so for slot detection, presumably due to noisy projection of the slot annotations. For slot detection, we get the best results using the ELMo embeddings which outperform the XLU embeddings as well as the bidirectional MT encoder in terms of exact match and slot F_1. Similarly as in the *target only* setting, the model with multilingual CoVe embeddings combined with the autoencoder performs worse than the other CoVe encoders. Overall, however, the choice of embeddings seems to have only a relatively small impact on the performance of the cross-lingual models. Importantly, however, we see improvements across all metrics as compared to training only on the target language data.

We observe similar results for Thai. The translation approach again yields the worst results for slot detection and we again see a consistent improvement from cross-lingual training as compared to training only on Thai data. And we again only observe small differences depending on the type of embeddings in the cross-lingual training scenario, but in this case, the models with the multilingual CoVe encoders outperform the model with the monolingual encoder.

Table 3 also shows the results for English. Not surprisingly, since we have an order of magnitude more data for English, the model trained and tested on English data still performs better than any of the evaluated methods for the other two languages. However, the gap between the numbers for English and the numbers for the other two languages does get considerably smaller for the cross-lingual models. Prima facie, the results also indicate that the models perform better for Thai than for Spanish. However, this is potentially an artifact of the data. As we mentioned above, we had to discard some of the Thai utterances for which the annotations disagreed with the annotations of their English translations and it is possible that we discarded a disproportionate number of more complex utterances which in return made parsing the Thai utterances easier.

In summary, the results from both languages suggest that pre-trained word representations as well as cross-lingual training improve results over training only on target language data without any pre-trained embeddings. The choice of embeddings, however, seems to matter less, and the overall performance seems to depend only very little on whether we use dynamic or static word representations or whether we use monolingual or bilingual word representations.

However, interestingly, for Spanish for which we compared more types of word representations than for Thai, the cross-lingual model with monolingual ELMo embeddings provided the best re-

sults. This potentially indicates that the benefit of cross-lingual training comes from sharing the biLSTM layer or the CRF layer and that embedding the tokens of the high-resource and the low-resource language in a similar space is not as important. At the same time, considering that we are getting relatively good results for both languages if we only train on the target language data, the potential of cross-lingual training might be limited in this case. To investigate whether the embedding type matters in more extreme low-resource scenarios, we also performed a series of zero-shot and low-resource experiments, which we describe in the next section.

6 Zero-shot learning and learning curves

As mentioned in the previous section, from the results on the full data, it is not entirely clear whether there is an advantage of using cross-lingual embeddings. We therefore conducted additional experiments with even smaller training sets in the target language: the case where we no annotated data in the target language exists (zero-shot) and the case where a very limited amount of training data in the target language exists. If there is no advantage of using cross-lingual embeddings over using monolingual embeddings, we expect to see similar results for all models independent of the training data size. On the other hand, if the multilingual CoVe encoder actually embeds phrases with similar meanings in the two languages in a similar vector space, we would expect the model with multilingual CoVe embeddings to perform much better in the zero-shot and very low-resource scenarios than any of the models with monolingual embeddings. Further, we can also investigate whether there is an advantage of translating the training data over other methods in extremely low-resource scenarios.

Experiments We used the same models with the same parameters as in the previous section. In the zero-shot case, we only use English data for training and model selection. For the learning curve experiments, we sample 10, 50, 100, or 200 utterances from each domain for the target language for training and model selection and upsample the target language data so that it roughly matches the size of the English data. For the zero-shot results, we present the average numbers across 5 runs. For the learning curve experiments, since we introduced another random factor by randomly sampling the training and model selection data, we repeat this process 10 times and report the average as well as the minimum and maximum values of the exact match metric for these experiments.

Results and discussion Table 4 shows the zero-shot results. These results indicate that using the multilingual CoVe embeddings works better than not using any encoder embeddings or using monolingual CoVe embeddings. This is true for the sentence-level domain and intent classification tasks as well as for slot detection. The Spanish results also suggest that in the zero-shot case, the multilingual encoder embeddings lead to better results than the XLU embeddings. However, also the models that use cross-lingual embeddings perform very poorly and contrary to the case where we have some training data in the target language, in the zero-shot scenario, the translation method works considerably better than any other of the transfer methods.

The results for different training set sizes are shown in Figure 2. These results generally confirm the patterns that we observed in the experiments with all available training data: cross-lingual training improves the results over training only on the target language (to a much bigger extent when there is much less target language training data available) and using pre-trained word representations leads to further improvements. We further observe that cross-lingual learning seems to lead to much more stable training which can be seen in the much smaller ranges of results as compared to the models trained only on the target language. Also in the extremely low-resource scenarios, the choice of embeddings seems to have very little effect. Lastly, as these plots show, the translation approach works better when there is very little training data available but the performance quickly plateaus and once there are are several hundred target language training examples available, joint training on multiple languages leads to better results.

Considering all results together, we find a consistent advantage of using cross-lingual training across all languages, training set sizes and embedding types. We further observe that the choice of embedding type has little effect as long as some form of pre-trained embeddings is being used. These facts together suggest that the main advantage of cross-lingual training comes from sharing the biLSTM and CRF layer across languages.

Figure 2: Results for different training set sizes. The top and the bottom of the error bars correspond to the highest and lowest value of the exact match metric among the 10 runs.

This is in line with the results by de Lhoneux et al. (2018) who found for cross-lingual training of dependency parsers, sharing the MLP layer for parser decisions improved results for all language pairs that they considered, whereas sharing of lower-level parameters only led to improvements for a limited set of language pairs.

7 Related work

Cross-lingual sequence labeling The task of cross-lingual and multilingual sequence labeling has gained a lot of attention recently. Yang et al. (2017) used shared character embeddings for cross-lingual transfer, and Lin et al. (2018) used shared character and sentence embeddings that were trained in a multitask setting for part-of-speech tagging and named entity recognition. Upadhyay et al. (2018) used cross-lingual embeddings for training multilingual slot-filling systems. Xie et al. (2018) used a similar model for NER but they first "translated" the high-resource training data by replacing each token with the token in the target language that was closest in vector space, and they further used character embeddings and a self-attention mechanism. Yu et al. (2018b) investigated using character-based language models for NER in several languages but did not do any cross-lingual learning. Plank and Agić (2018) used cross-lingual embeddings, projected annotations, and dictionaries for zero-shot cross-lingual part-of-speech tagging.

Cross-lingual sentence representations Recently, there was also a lot of work of using cross-lingual sequence encoders for sentence classifications using either multilingual MT encoders similar to ours (e.g., Eriguchi et al., 2018; Yu et al., 2018a; Singla et al., 2018) or training encoders and then aligning their vector spaces after pre-training (Conneau et al., 2018; Schuster et al., 2019). Even more recently, Lample and Conneau (2019) and Mulcaire et al. (2019) showed that it is also possible to directly train contextual word representations jointly on multiple languages.

Cross-lingual transfer for other tasks Apart from tasks such as slot filling and NER, cross-lingual transfer learning has also been investigated a lot for syntactic tasks, and in particular for part-of-speech tagging and dependency parsing. Early work trained part-of-speech taggers for individual languages and then trained delexicalized dependency parsers (e.g., Zeman and Resnik, 2008; McDonald et al., 2011). Further, a lot of syntactic and semantic parsing models recently successfully incorporated parameter sharing for training parsers in closely related languges (Duong et al., 2015; Ammar et al., 2016; Susanto and Lu, 2017; Smith et al., 2018; de Lhoneux et al., 2018). In the domain of dialog managers, Mrkšić et al. (2017) and Chen et al. (2018) presented methods for cross-lingual transfer for dialog state tracking.

8 Conclusion and future work

In this paper, we presented a new multilingual intent and slot filling data set for task oriented dialog of around 57,000 utterances and evaluated the performance of different methods for cross-lingual transfer learning, including a novel method using cross-lingual contextual word representations. For both investigated languages, we consistently found that cross-lingual learning improves results as compared to only training on limited amounts of target language data, and our results suggest that the choice of multilingual or monolingual embeddings has only a small impact on the overall performance.

Despite the range of models that we considered in this paper, we only scratched at the surface of possible cross-lingual (embedding) models, and hence there are many future directions of this work. First, except for the Spanish ELMo embeddings, we did not use any character embeddings in any of our experiments or models. This presumably makes sense for the English-Thai transfer learning case since these two languages use different writing systems but given the results by Lin et al. (2018) and Yang et al. (2017), we would expect additional improvements by sharing character embeddings for languages with similar writing systems.

Second, one could try to include a specific learning objective to embed translations into a similar vector space as used by Yu et al. (2018a) and Conneau et al. (2018) for multilingual sentence representations.

As yet another extension, one could combine the approaches of multilingual CoVe embeddings and monolingual ELMo (or BERT, Devlin et al., 2018) embeddings and jointly train an encoder with a language model and an MT objective, which would potentially combine the benefit of training a model on large monolingual corpora while at the same time aligning the vector spaces of the two languages. A similar approach worked well for cross-lingual NLI inference on the XNLI data set (Conneau et al., 2018) as well as for unsupervised machine translation (Lample and Conneau, 2019).

We hope that our data set will facilitate research in these directions and ultimately lead to improved natural language understanding models for low-resource languages.

Acknowledgements

We thank the three anonymous reviewers for their thoughtful comments. We also thank Luke Zettlemoyer and Christopher Manning for valuable feedback throughout this project, and Maria Sumner and Neghesti Tinsew for their help with collecting the annotations and improving their consistency.

References

Ahmed Aly, Kushal Lakhotia, Shicong Zhao, Mrinal Mohit, Barlas Oguz, Abhinav Arora, Sonal Gupta, Christopher Dewan, Stef Nelson-Lindall, and Rushin Shah. 2018. PyText: A seamless path from NLP research to production. ArXiv preprint.

Waleed Ammar, George Mulcaire, Miguel Ballesteros, Chris Dyer, and Noah A. Smith. 2016. Many languages, one parser. *Transactions of the Association for Computational Linguistics*, 4:431–444.

Mauro Cettolo, Christian Girardi, and Marcello Federico. 2012. Wit^3: Web inventory of transcribed and translated talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, Trento, Italy.

Wanxiang Che, Yijia Liu, Yuxuan Wang, Bo Zheng, and Ting Liu. 2018. Towards better UD parsing: Deep contextualized word embeddings, ensemble, and treebank concatenation. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*.

Wenhu Chen, Jianshu Chen, Yu Su, Xin Wang, Dong Yu, Xifeng Yan, and William Yang Wang. 2018. XL-NBT: A cross-lingual neural belief tracking framework. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*, pages 414–424.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. In *Proceedings of ICLR 2018*.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. ArXiv preprint.

Timothy Dozat, Peng Qi, and Christopher D. Manning. 2017. Stanford's graph-based neural dependency

parser at the CoNLL 2017 shared task. In *Proceedings of the CoNLL 2017 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015. Low resource dependency parsing: Cross-lingual parameter sharing in a neural network parser. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (ACL 2015)*, pages 845–850.

Akiko Eriguchi, Melvin Johnson, Orhan Firat, Hideto Kazawa, and Wolfgang Macherey. 2018. Zero-shot cross-lingual classification using multilingual neural machine translation. ArXiv preprint.

Murhaf Fares, Andrey Kutuzov, Stephan Oepen, and Erik Velldal. 2017. Word vectors, reuse, and replicability: Towards a community repository of large-text resources. In *Proceedings of the 21st Nordic Conference on Computational Linguistics*, pages 271–276.

Jonas Gehring, Michael Auli, David Grangier, and Yann N Dauphin. 2016. A Convolutional Encoder Model for Neural Machine Translation. ArXiv preprint.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional Sequence to Sequence Learning. ArXiv preprint.

Diederik P. Kingma and Jimmy Lei Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of ICLR 2015*.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT summit*.

Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2016)*, pages 260–270.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. ArXiv preprint.

Miryam de Lhoneux, Johannes Bjerva, Isabelle Augenstein, and Anders Søgaard. 2018. Parameter sharing between dependency parsers for related languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*, pages 4992–4997.

Gurvan L'Hostis, David Grangier, and Michael Auli. 2016. Vocabulary selection strategies for neural machine translation. ArXiv preprint.

Ying Lin, Shengqi Yang, Veselin Stoyanov, and Heng Ji. 2018. A multi-lingual multi-task architecture for low-resource sequence labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pages 799–809.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *Proceedings of ICLR 2017*.

Pierre Lison, Jörg Tiedemann, and Milen Kouylekov. 2018. OpenSubtitles2018: Statistical rescoring of sentence alignments in large, noisy parallel corpora. In *Proceedings of the 11th edition of the Language Resources and Evaluation Conference (LREC 2018)*, pages 1364–1369.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Proceedings of Interspeech 2016*.

Minh-Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 1412–1421.

Bryan McCann, James Bradbury, and Richard Socher. 2017. Learned in translation: Contextualized word vectors. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing (EMNLP 2011)*, pages 62–72.

Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *Proceedings of Interspeech 2013*.

Nikola Mrkšić, Ivan Vulić, Diarmuid Ó Séaghdha, Ira Leviant, Roi Reichart, Milica Gašić, Anna Korhonen, and Steve Young. 2017. Semantic specialization of distributional word vector spaces using monolingual and cross-lingual constraints. *Transactions of the Association for Computational Linguistics*, 5:309–324.

Phoebe Mulcaire, Jungo Kasai, and Noah A. Smith. 2019. Polyglot contextual representations improve crosslingual transfer. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*.

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 (NAACL 2018)*.

Barbara Plank and Željko Agić. 2018. Distant supervision from disparate sources for low-resource part-of-speech tagging. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 614–620.

Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2017. A survey of cross-lingual word embedding models. ArXiv preprint.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL 2019)*.

Karan Singla, Dogan Can, and Shrikanth Narayanan. 2018. A multi-task approach to learning multilingual representations. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pages 214–220.

Aaron Smith, Bernd Bohnet, and Miryam de Lhoneux. 2018. 82 treebanks, 34 models: Universal Dependency parsing with multi-treebank models. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*.

Raymond Hendy Susanto and Wei Lu. 2017. Neural architectures for multilingual semantic parsing. In *Proceedings of the 55th Annual Meeting ofthe Association for Computational Linguistics (ACL 2017)*, pages 38–44.

Shyam Upadhyay, Manaal Faruqui, Gokhan Tür, Dilek Hakkani-Tür, and Larry Heck. 2018. (Almost) zero-shot cross-lingual spoken language understanding. In *Proceedings of the IEEE ICASSP 2018*.

Ngoc Thang Vu. 2016. Sequential convolutional neural networks for slot filling in spoken language understanding. In *Proceedings of Interspeech 2016*, pages 3250–3254.

Jiateng Xie, Zhilin Yang, Graham Neubig, Noah A. Smith, and Jaime Carbonell. 2018. Neural cross-lingual named entity recognition with minimal resources. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*, pages 369–379.

Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. In *Proceedings of ICLR 2017*.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research*.

Katherin Yu, Haoran Li, and Barlas Oguz. 2018a. Multilingual seq2seq training with similarity loss for cross-lingual document classification. In *Proceedings of The Third Workshop on Representation Learning for NLP*, pages 175–179.

Xiaodong Yu, Stephen Mayhew, Mark Sammons, and Dan Roth. 2018b. On the strength of character language models for multilingual named entity recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*, pages 3073–3077.

Daniel Zeman and Philip Resnik. 2008. Cross-language parser adaptation between related languages. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*.