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. Neural sequence labeling models have achieved very high accuracy on these tasks when trained on large amounts of training data. However, collecting this data is very time-consuming and therefore it is unfeasible to collect large amounts of data for many languages. For this reason, it is desirable to make use of existing data in a high-resource language to train models in low-resource languages. In this paper, we investigate the performance of three different methods for cross-lingual transfer learning, namely (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, we find that multilingual contextual word representations give better results than using cross-lingual static embeddings. We release the new data set and plan to release our implementation of the NLU models in the near future.

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*) [23, 19]. 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 lend themselves to using a biLSTM-CRF sequence labeling model [14, 28] 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, expanding a conversational AI system to many new languages is a very resource-intensive task and clearly not feasible to be done for the more than 8,000 languages that are currently spoken around the world.

*Work carried out during an internship at Facebook.

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.

In this work, we explore different strategies to make use of existing English training data to improve intent and slot detection models for other languages. Concretely, we are considering two target languages: Spanish, an Indo-European language with the same writing system as English, and Thai, a Kra-Dai language with a different writing system than English. We investigate two existing strategies for cross-lingual transfer, namely using cross-lingual pre-trained embeddings (XLU embeddings) 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 contextual word representations.

We evaluate our models on a novel data set with English, Spanish, and Thai utterances and we find for both languages that contextual cross-lingual embeddings as well as XLU embeddings consistently outperform the translation approach. Further, for extremely low-resource cases, contextual cross-lingual embeddings give additional improvements over using the static XLU embeddings.

We release the data at `http://url.to.data` and plan to release our implementation of the NLU models in the near future.

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 utterances. 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.

Despite this potential limitation of the Thai data, 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 sequence labeling 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.

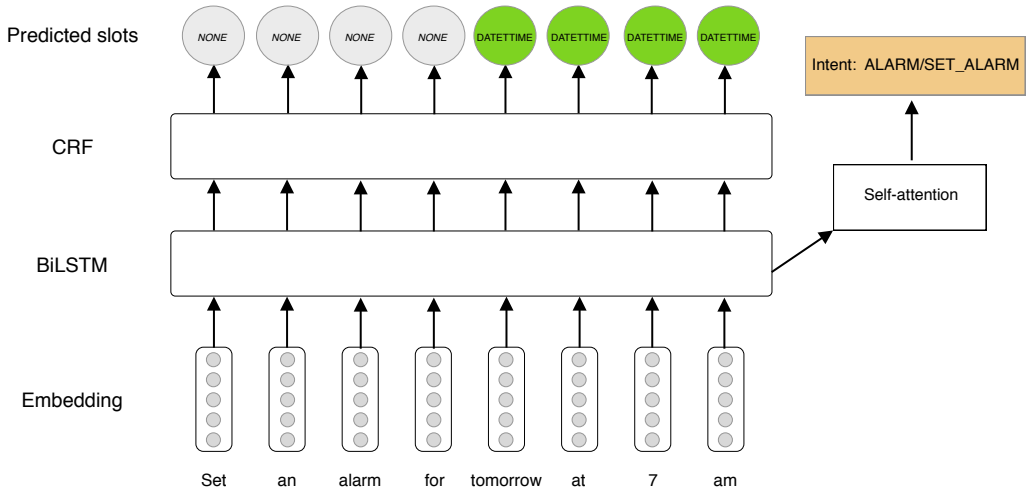

Figure 1: Slot and intent model architecture. Word embeddings are passed through a biLSTM layer which is shared across the slot detection and intent prediction tasks.

3 Approach

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 basic 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 [17] over the hidden states of the biLSTM 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[2] 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. [7] by 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 encoder.)

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 are using a bidirectional LSTM encoder with two layers. Overall, we compared three strategies for training these encoders:

- **CoVe**: Following McCann et al. [21], we train a neural machine translation model to translate from the low-resource language (Spanish or Thai) to English.

[2]In early experiments we found that this strategy gives as good results as using pre-trained embeddings and since using pre-trained embeddings would have introduced additional variables (e.g., the vocabulary and the training data), we decided to use this embedding strategy as a baseline.

- **bidirectional MT**: 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 [31]. In this model, the encoder does not have access to the target language and therefore we anticipate that it will learn to encode phrases with similar meanings into similar vector spaces across languages.
- **bidirectional MT + autoencoder** We train a bidirectional neural machine translation model and combine it with an auto-encoder 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.

Note that the CoVe encoder is trained to encode only the low-resource language and is therefore not a multilingual encoder.

Implementation details We train all models using a wrapper around the fairseq [11, 12] 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.[3] For the machine translation models, we further use dot-product attention [20] 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 [15].

Data For the Spanish models, we use two copies[4] of Europarl v7 [13], every eighth sentences of the Paracrawl data[5]. 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 [2] as well as the OpenSubtitles data [18] for training and the IWSTL development and test data for model selection. We use the 20,000 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.[6]

[3] In theory, we could have also used a weighted combination of all layers as it is common with ELMo [24]. We opted for the simplest solution of using only the hidden states of the final layer as word representations and we leave the exploration of more complex combinations of the encoder hidden states to future work.

[4] 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.

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

[6] 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.

Spanish	Epoch	es→en	en→es	es→es	en→en
CoVE (unidirectional)	81	8.50	-	-	-
bidirectional	98	8.27	6.90	-	-
bidirectional + 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	-	-	-
bidirectional	35	12.73	17.00	-	-
bidirectional + 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.

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 the 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.
- **Translate train**: Combining the target training data with the English data automatically translated to the target language. The slot annotations are projected via the attention weights. We translate the data using a commercial neural machine translation system.
- **Cross-lingual with XLU embeddings**: Joint training on the English and target language data with pre-trained MUSE [4] cross-lingual embeddings. Note that MUSE embeddings are not available for Thai and therefore we only evaluate this method for Spanish.
- **Target only with encoder embeddings**: Using only the low-resource language training data and using pre-trained encoder embeddings.
- **Cross-lingual with encoder embeddings**: Joint training on the English and target language data using pre-trained encoder embeddings.

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 frame accuracy which indicates the number of utterances for which the domain, intent, and all slots were correctly identified. Frame accuracy 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 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. The model using the bidirectional MT encoder combined with the autoencoder only marginally improves results over the baseline without any encoder embeddings.

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 MUSE embeddings which slightly outperform the bidirectional MT encoder in terms of frame accuracy and slot F_1. Also in the cross-lingual setting, the bidirectional MT encoder combined with the autoencoder performs worse than the other MT encoders. Overall, however, the choice of embeddings seems to have only a very small impact on the performance of the cross-lingual models. Nevertheless, we do 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. When we perform cross-lingual training, we also observe differences depending on the type of MT encoder: The bidirectional MT encoders outperform the CoVe encoder.

Spanish	Embedding type	Frame acc.	Domain acc.	Intent acc.	Slot F1
Target only	-	72.94	99.43	97.26	80.95
Target only	CoVe	73.93	99.52	97.43	81.51
Target only	bidir. MT	74.13	99.55	97.61	81.64
Target only	bidir. MT + auto	73.05	99.51	97.13	81.22
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	bidir. MT	75.20	99.56	97.82	82.49
Cross-lingual	bidir. MT + auto	74.68	99.59	97.90	82.13
Thai	Embedding type	Frame acc.	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	bidir. MT	84.66	99.37	96.75	90.20
Target only	bidir. MT + 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	bidir. MT	85.76	99.39	96.98	91.22
Cross-lingual	bidir. MT + auto	**86.12**	99.33	96.87	**91.51**

Table 3: Results averaged over 5 training runs.

In summary, the Thai results suggest that our models indeed facilitate cross-lingual transfer, and that the gains from adding encoder embeddings is a combination of using contextual word representations and cross-lingual transfer. For Spanish, the picture is not as clear since all three MT encoder embeddings led to very similar results, including the monolingual CoVe embeddings. This potentially indicates that the gains we observed when doing cross-lingual training mainly came from learning something about individual lexical items such as Miami being a location but not transferring any knowledge about phrases that constitute spans. At the same time, considering that we are getting good results for both languages if we only train on the low-resource language data, the potential of cross-lingual training might be limited in this case. To investigate this further, 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, is is not entirely clear what effect cross-lingual training has on the results. We therefore conducted additional experiments with even smaller training sets in the target language: the case where we no data in the target language exists (zero-shot) or the case where a very limited amount of training data in the target language exists. If cross-lingual transfer only happens at the token level, we would expect that all encoder embedding types lead to similar results. However, if the multilingual MT encoder actually embeds phrases with similar meanings in the two languages in a similar vector space, we would expect that the multilingual MT encoder performs much better in the zero-shot and very low-resource scenarios. Further, for Spanish, we can also investigate whether there is a benefit of using contextual multilingual embeddings over using static XLU embeddings.

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

Spanish	Embedding type	Frame acc.	Domain acc.	Intent acc.	Slot F1
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	bidir. MT	**10.56**	**59.29**	53.34	**22.50**
Cross-lingual	bidir. MT + auto	9.28	59.25	**53.89**	19.25
Thai	Embedding type	Frame acc.	Domain acc.	Intent acc.	Slot F1
Cross-lingual	-	0.20	39.36	39.11	3.44
Cross-lingual	CoVe	5.82	66.75	54.24	8.84
Cross-lingual	bidir. MT	15.37	73.84	66.35	32.52
Cross-lingual	bidir. MT + auto	**20.84**	**81.95**	**70.70**	**35.62**

Table 4: Zero-shot results averaged over 5 training runs.

the training and model selection data, we repeat this process 10 times and report the average as well as the minimum and maximum frame accuracy for these experiments.

Results and discussion Table 4 shows the zero-shot results. These results consistently indicate that using a multilingual sentence encoder works much 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.

We observe similar results when we consider the results for different training set sizes as shown in Figure 2. For both languages, the bidirectional MT encoder embeddings led to the best results for all investigated training data sizes with the exception of the largest Spanish training set for which the XLU embeddings yielded better results. Importantly, however, the model with the bidirectional MT encoder consistently outperformed the model with the monolingual CoVe encoder. In combination with the zero-shot results, this provides strong evidence that the bidirectional MT encoder indeed learns to embed phrases with similar meanings across languages into a similar vector space which allows for efficient cross-lingual transfer learning.

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. [30] used shared character embeddings for cross-lingual transfer, and Lin et al. [16] 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. [27] used cross-lingual embeddings for training multilingual slot-filling systems. Xie et al. [29] 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. [32] investigated using character-based language models for NER in several languages but did not do any cross-lingual learning.

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. [9], Yu et al. [31]) or training encoders and then aligning their vector spaces after pre-training [5].

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., [33, 22]). Further, a lot of syntactic and semantic parsing models recently successfully incorporated parameter sharing for training parsers in closely related languges. [8, 1, 26, 25, 6].

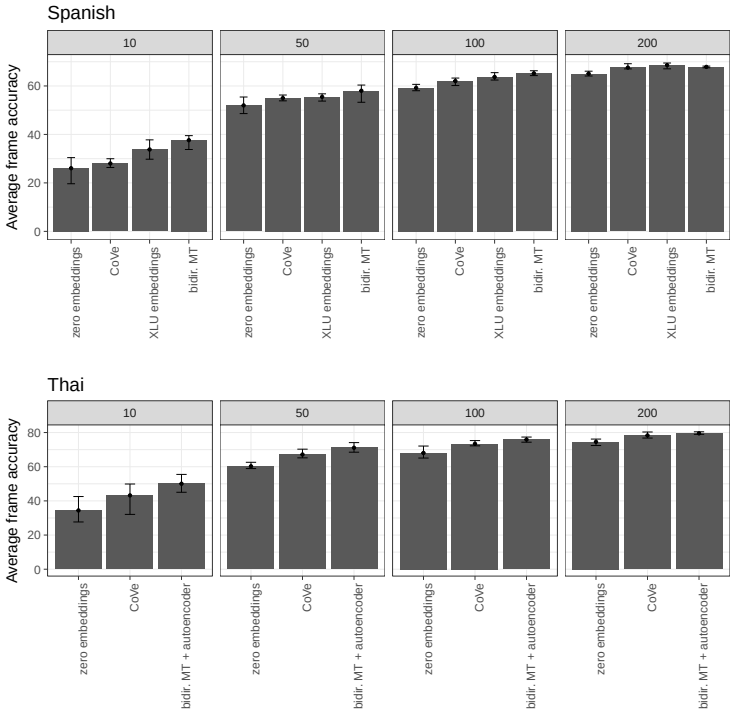

Figure 2: Results for different training set sizes. The top and the bottom of the error bars correspond to the highest and lowest frame accuracy among the 10 runs.

8 Conclusion and future work

In this paper, we collected a new multilingual intent and slot filling dataset for task oriented dialog and investigated the performance of three 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. We further found that models using cross-lingual representations – either contextual or static – outperform models trained on translated training data and that in extremely low-resource scenarios, contextual word representations seem to be beneficial over static word representations.

There are many natural extensions to this work. First, 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 alphabets; given the results by Lin et al. [16] and Yang et al. [30], we would expect additional improvements by using character embeddings.

Second, one could try to include a specific learning objective to embed translations into a similar vector space as used by Yu et al. [31] and Conneau et al. [5] for multilingual sentence representations.

Third, given the recent success of contextual word representations that were trained on monolingual data such as ElMo [24], it is likely that combining monolingual and multilingual contextual word representations would further improve the results. In fact, in preliminary experiments, we found that combining the MT encoder embeddings and the Spanish ELMo embeddings by Che et al. [3, 10] led to further improvements.

Finally, the presented multilingual MT encoder embeddings seem applicable to training multilingual models for a range of tasks and it would be interesting to investigate whether our results also hold for other sequence labeling tasks such as named entity recognition or part-of-speech tagging as well as entirely different tasks such as multilingual dependency parsing.

References

[1] Ammar, W., G. Mulcaire, M. Ballesteros, C. Dyer, and N. A. Smith (2016). Many Languages, One Parser. *Transactions of the Association for Computational Linguistics 4*, 431–444.

[2] Cettolo, M., C. Girardi, and M. 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.

[3] Che, W., Y. Liu, Y. Wang, B. Zheng, and T. 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*.

[4] Conneau, A., G. Lample, M. Ranzato, L. Denoyer, and H. Jégou (2017). Word translation without parallel data. arXiv preprint.

[5] Conneau, A., G. Lample, R. Rinott, H. Schwenk, V. Stoyanov, A. Williams, and S. R. Bowman (2018). XNLI: Evaluating Cross-lingual Sentence Representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP 2018)*.

[6] de Lhoneux, M., J. Bjerva, I. Augenstein, and A. 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)*.

[7] Dozat, T., P. Qi, and C. 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*.

[8] Duong, L., T. Cohn, S. Bird, and P. 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*.

[9] Eriguchi, A., M. Johnson, O. Firat, H. Kazawa, and W. Macherey (2018). Zero-shot cross-lingual classification using multilingual neural machine translation. arXiv preprint.

[10] Fares, M., A. Kutuzov, S. Oepen, and E. 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*, Gothenburg, Sweden.

[11] Gehring, J., M. Auli, D. Grangier, and Y. N. Dauphin (2016). A convolutional encoder model for neural machine translation. arXiv preprint.

[12] Gehring, J., M. Auli, D. Grangier, D. Yarats, and Y. N. Dauphin (2017). Convolutional sequence to sequence learning. arXiv preprint.

[13] Koehn, P. (2005). Europarl: A parallel corpus for statistical machine translation. In *MT summit*.

[14] Lample, G., M. Ballesteros, S. Subramanian, K. Kawakami, and C. 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*.

[15] L'Hostis, G., D. Grangier, and M. Auli (2016). Vocabulary selection strategies for neural machine translation. arXiv preprint.

[16] Lin, Y., S. Yang, V. Stoyanov, H. Ji, S. Clara, A. M. Learning, and M. Park (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)*.

[17] Lin, Z., M. Feng, C. N. dos Santos, M. Yu, B. Xiang, B. Zhou, and Y. Bengio (2017). A structured self-attentive sentence embedding. In *Proceedings of ICLR 2017*.

[18] Lison, P., J. Tiedemann, and M. Kouylekov (2018). Opensubtitles 2018: 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)*.

[19] Liu, B. and I. Lane (2016). Attention-based recurrent neural network models for joint intent detection and slot filling. In *INTERSPEECH*.

[20] Luong, M.-T., H. Pham, and C. 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)*.

[21] McCann, B., J. Bradbury, and R. Socher (2017). Learned in translation: Contextualized word vectors. In *Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS 2017)*.

[22] McDonald, R., S. Petrov, and K. Hall (2011). Multi-Source Transfer of Delexicalized Dependency Parsers. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*.

[23] Mesnil, G., X. He, L. Deng, and Y. Bengio (2013). Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding. In *Interspeech*.

[24] Peters, M., M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. 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 (NAACL 2018)*. Association for Computational Linguistics.

[25] Smith, A., B. Bohnet, and M. D. 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*.

[26] Susanto, R. H. and W. Lu (2017). Neural Architectures for Multilingual Semantic Parsing. In *Proceedings ofthe 55th Annual Meeting ofthe Association for Computational Linguistics (ACL 2017)*.

[27] Upadhyay, S., M. Faruqui, G. Tur, D. Hakkani-Tur, L. Heck, and D. Hakkani-t (2018). (Almost) Zero-Shot Cross-Lingual Spoken Language Understanding. In *Proceedings of the IEEE ICASSP*.

[28] Vu, N. T. (2016). Sequential convolutional neural networks for slot filling in spoken language understanding. In *Interspeech 2016*, pp. 3250–3254.

[29] Xie, J., Z. Yang, G. Neubig, N. A. Smith, and J. 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)*.

[30] Yang, Z., R. Salakhutdinov, and W. W. Cohen (2017). Transfer learning for sequence tagging with hierarchical recurrent networks. In *Proceedings of ICLR 2017*.

[31] Yu, K., H. Li, and B. Oguz (2018). Multilingual seq2seq training with similarity loss for cross-lingual document classification. In *Proceedings of the 3rd Workshop on Representation Learning for NLP*, pp. 175–179.

[32] Yu, X., S. Mayhew, M. Sammons, and D. Roth (2018). 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)*.

[33] Zeman, D. and P. Resnik (2008). Cross-Language Parser Adaptation between Related Languages. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*.