ZERO-SHOT SPOKEN LANGUAGE UNDERSTANDING FOR ENGLISH-HINDI: AN EASY VICTORY AGAINST WORD ORDER DIVERGENCE

Judith Gaspers, Quynh Do
Amazon Alexa AI
{gaspers, doquynh}@amazon.de

ABSTRACT

While the strong zero-shot performance of multilingual BERT has been shown to drop in case of word order divergence between source and target language, the problem has been studied rarely to date. In this paper, we explore light-weight techniques to improve BERT-based zero-shot spoken language understanding for English-Hindi, which are languages with divergent word orders. We show that word order divergence can be tackled by *reordering the source data to reflect target language word order*. In particular, we study two computationally inexpensive methods for re-ordering the source data to better match that of the target language: one making use of slot label information, and another one making use of syntactic parse trees. Our experiments show that the former, which is simpler and doesn't require any additional resources when compared to vanilla zero-shot transfer, can obtain surprisingly large improvements on a real-world dataset.

1 INTRODUCTION

Spoken Language Understanding (SLU) models are essential for the development of voice-controlled devices like Alexa or Google Home. The task of SLU can be typically divided into the two sub-tasks intent classification (IC) to identify the user's intent, and slot filling (SF) to extract necessary semantic constituents. For instance, if a user requests "play madonna", IC should identify *PlayMusic* as the intent, while SF should classify the two tokens in the request "play" and "madonna" as *Other* and *Artist*, respectively. Recently, as in many other language processing fields, we have been observing the success of BERT-based models which jointly learn IC and SF labels by leveraging pre-trained BERT representations (Chen et al., 2019).

Due to the growing success of natural language understanding (NLP) applications, porting NLP models from a resource-rich source language to a new target language in a cost-efficient manner, i.e. using little or no supervised target language data, has attracted an increasing interest in recent years. For zero-shot scenarios, where no supervised training data is available in the target language, a common approach is fine-tuning a pre-trained multilingual language model like multilingual BERT (M-BERT, Devlin et al. (2018)), on the supervised training data of the source language and subsequently applying the trained model on the target language. M-BERT has been shown to give impressive zero-shot results across several NLP tasks (Wu & Dredze, 2019), including SLU (Xu et al., 2020). However, while the cross-lingual performance of M-BERT in zero-shot scenarios is generally strong, it suffers when word order diverges between source and target languages (Pires et al., 2019). This is due to the fact that the linguistic patterns learned by a DNN are from the source language, and therefore may not work well on the target language if word orders diverge. Notably though, this problem gained attention only recently and has been studied rarely to date.

In this paper, we explore zero-shot transfer of SLU models from English to Hindi, which are two languages with divergent word orders, for the use case of a voice-controlled device. In particular, we fine-tune M-BERT on English SLU data as the downstream task, and subsequently apply the resulting model on Hindi as the target language. Our goal is to improve the SLU performance on the target language by tackling the issue of divergent word orders, which, to the best of our knowledge, has not yet been addressed in the literature.

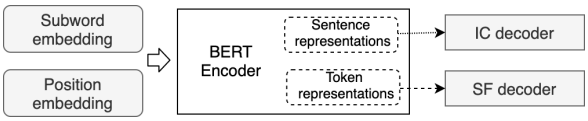

Figure 1: SLU architecture.

The typical word orders in English and Hindi are SVO and SOV, respectively. To address word order divergence, we aim to re-order the English source data to reflect the word order of the Hindi target data. Intuitively, if the model has access to word order information of the target language during training, zero-shot performance could be improved. We, therefore, investigate two *simple and computationally inexpensive* reordering approaches: one which makes use of slot label information based on the observation that user requests are often in imperative form, and another using syntactic parse trees. The empirical results on a real-world SLU dataset show that the former can achieve surprisingly large improvements on English-Hindi zero-shot SLU.

2 Related work

Various approaches have been proposed for few and zero-shot SLU, such as, using machine translated data (Gaspers et al., 2018) or pre-training a DNN on data from one or more source languages (e.g. Do & Gaspers (2019); He et al. (2020)). The effectiveness of these approaches often depends heavily on the linguistic similarity of the source and target languages. Recently, there has been a rising effort in improving cross-lingual transfer between distant languages, for example, by tackling differences at the linguistic/embedding level (Johnson et al., 2019).

As one of the greatest recent successes in NLP, M-BERT has been shown to achieve relatively strong zero-shot cross-lingual transfer learning performance for not only SLU (Xu et al., 2020) but also many other NLP tasks. There have been several works investigating the weakness of M-BERT and proposing techniques to improve BERT-based zero-shot models further. Cao et al. (2020) proposed to align contextual word embeddings using parallel text or dictionaries. Pires et al. (2019) showed that M-BERT performance may drop in case of distant source and target languages with word order divergences. Liu et al. (2020) found that reducing word order information from the source language improves zero-shot performance for downstream tasks. Motivated by these works, we aim at addressing the issue of word order divergences to improve BERT-based zero-shot SLU models. However, in contrast to Liu et al. (2020), we do not reduce only word order information of the source language, but explicitly aim to model the target language word order. Moreover, our approach does not require expensive resources like parallel text or dictionaries as in Cao et al. (2020).

One of the two reordering methods for SLU discussed in this paper is based on syntactic parse trees. This approach was previously used to address word order divergence to improve machine translation (Ramanathan et al., 2008; Murthy et al., 2019).

3 Method

In this paper, we explore M-BERT-based zero-shot transfer learning for SLU by explicitly reordering supervised source utterances to better reflect the word order of the target language. In the following, we first describe our BERT-based SLU model and subsequently two methods for reordering the source data.

3.1 SLU model

We use a common SLU architecture for joint intent classification and slot filling, which is depicted in Fig. 1. It consists of a BERT encoder, an intent decoder and a slot decoder. The BERT encoder's outputs at sentence and token level are used as inputs for the intent and slot decoders, respectively. The intent decoder is a standard feed-forward network including two standard dense layers and a softmax layer on top. Meanwhile, the slot decoder uses a CRF layer on top of two dense layers to leverage the sequential information of slot labels. During the training, the losses of IC (cross-entropy loss) and SF (CRF loss) are optimized jointly with equal weights (1.0:1.0).

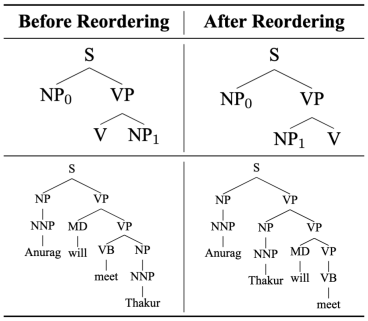

Figure 2: Example for re-ordering based on syntactic information for the use case of a transitive verb; the figure is taken from Murthy et al. (2019).

3.2 Reordering via slot labels

This simple method is based on the observation that English user requests sent to voice-controlled devices are often in imperative form, implying that English utterances often start with a verb (VO – subjects are omitted). To better reflect the Hindi word order (OV), we aim to *move the verbs from the beginning to the end of the English utterances*. For this purpose, we make use of the available slot values. Recall that for the slot filling task, semantic constituents are labelled in the supervised data, which is available in the source language English in our case. In particular, each token in an utterance is labelled either with a slot label, if it carries relevant slot information, or with *Other*, if it does not carry slot information. However, tokens labelled with *Other* may contain important information related to the intent.

Verb detection: Instead of using an extra sophisticated verb detection method, we make use of the available slot labels. In particular, we observe that when the slot label signature of an utterance starts with one or several *Other*(s), the corresponding token(s) often express the user intent, and include the main verb of the utterance. Let us consider the previous example "play madonna" which has intent label *PlayMusic* and slot label signature "*Other Artist*". In this case, the token "play" has assigned label *Other*, which is reserved for tokens which do not carry slot information. However, it is in fact the main verb which expresses the intent referring to the expected device action.

Reordering: If a training utterance starts with a sequence of *Other*(s), we move the label sequence and corresponding token(s) from the beginning to the end of the sentence. For instance, reordering the previous example would result in "madonna play" with slot label sequence "*Artist Other*". Since user requests are typically rather short and follow a similar pattern, this simple method can capture a large amount of utterances already.

3.3 Reordering via syntactic parse trees

In this approach, we adapt the work in the field of phrase-based (Ramanathan et al., 2008) and neural (Murthy et al., 2019) machine translation to SLU. Roughly speaking, for an English utterance, a syntactic parse tree is automatically created, and then hand-crafted rules are applied on top of it to reorder the utterances to Hindi-typical SOV order. Fig. 2 provides an example for a re-ordering rule and its effect on an example sentence. Further details can be found in Ramanathan et al. (2008). Noticeably, unlike machine translation data used in Ramanathan et al. (2008); Murthy et al. (2019), SLU data is labelled. We, therefore, re-order slot labels together with a sentence, i.e. each token keeps its original label in the reordered version.

4 Experimental set up

4.1 Datasets

Since we present a practical approach based on observations from real-world data, which to the best of our knowledge are not reflected well in publicly available SLU datasets, we present results on real-

world data. In particular, we extracted random samples, which are representative of user requests to voice-controlled devices, from a large-scale commercial SLU system. The samples were suitably anonymized and manually annotated with intent and slot labels. In particular, we use English and Hindi data from six domains, i.e. *HomeAutomation*, *Weather*, *Video*, *Music*, *Notifications* and *Books*. Data statistics are shown in Table 1; for each domain, we use the Hindi data for testing and split the English data into 90% training and 10% validation data.

Domain	**# Enlish utt.**	**# intents**	**# slots**	**# Hindi utt.**	**# intents**	**# slots**
HomeAutomation	100,000	20	41	10,000	11	17
Weather	69,288	2	15	6,035	1	11
Video	13,208	19	44	4,520	9	39
Music	100,000	20	84	10,000	14	51
Notifications	100,000	19	41	10,000	10	18
Books	22,902	20	38	1,582	10	15
total	405,398	100	263	52,155	55	151

Table 1: Dataset statistics.

4.2 SETTINGS

We use pre-trained M-BERT (Devlin et al., 2018) (size 768), which is pre-trained on large amounts of unlabelled texts from multiple languages, and max-pooling for sentence representation. Each of our decoders, i.e. for IC and SF, has 2 dense layers of size 768 with gelu activation. The dropout values used in IC and SF decoders are 0.5 and 0.2, respectively. For optimization, we use Adam optimizer with learning rate 0.1 and a Noam learning rate scheduler. We trained our model for 50 epochs with batch size of 64. For evaluation, we use the standard metrics for SLU, i.e. F1 for SF and accuracy for IC. In addition, following Gaspers et al. (2018), we use a semantic error rate, which measures IC and SF jointly and is defined as:

$$SemER = \frac{\#(\text{slot+intent errors})}{\#\text{slots in reference + 1}} \tag{1}$$

5 EXPERIMENTS

For each domain, we build three SLU models using the English data:

- *Baseline*: The baseline model is obtained by training on the original English data.
- *Reordering via slot labels*: The utterances in the English data are reordered using slot label information as described in Section 3.2. The model is then obtained by training on the reordered English data.
- *Reordering via syntactic parse*: The English utterances are reordered using rules and syntactic parse trees as described in Section 3.3. Syntactic parse tress are generated with the Stanford parser (Socher et al., 2013). We reorder labels accordingly, so that each token keeps its original label. The model is obtained by training on the reordered English data.

Subsequently, we measure zero-shot performance for each model by applying it directly on the corresponding domain's Hindi test data. The results are presented in Table 5.

The results indicate that the simple approach based on slot label information is effective, yielding consistent reductions in semantic error rate across domains of up to 30.16% and an average reduction of 13.33% compared to the "baseline" zero-shot approach of training on the original English data. The metrics for the individual sub-tasks reveal that it is mostly the SF task which benefitted. In particular, slot F1 shows a large relative improvement of 15.47% on average, while the gain in intent accuracy is rather small. This may be expected, as word order information seems to be much more relevant for the sequence labelling task SF than for the easier intent classification task.

The, in general, more sophisticated method based on rules and parse-tree information yields mixed results. That is, the method improves performance for certain domains and individual tasks, but decreases performance for some others. For instance, for the *Music* domain, it achieved 22.0% and 1.62% relative gains in intent accuracy and slot filling, respectively.

Domain	Reordering via slot labels			Reordering via syntactic parse		
	SemER	Slot F1	IC acc.	SemER	Slot F1	IC acc.
HomeAutomation	-13.54	+12.17	0.0	+2.72	-1.4	+1.49
Weather	-30.16	+20.1	+0.26	+9.12	-63.34	-0.07
Video	-5.83	+9.68	-6.18	+0.94	+3.83	-13.17
Music	-13.0	+22.12	+14.33	-1.98	+1.62	+22.0
Notifications	-8.72	+11.83	+5.17	+1.17	-2.14	+4.6
Books	-8.72	+16.94	-6.22	-7.47	+6.26	-0.79
Avg.	-13.33	+15.47	+1.23	+0.75	-9.2	+2.34

Table 2: Relative change in semantic error rate (SemER), intent classification accuracy and slot F1 for zero-shot English-Hindi SLU for training on English training data with re-ordered word order compared to training on the original English data as the baseline. Negative numbers indicate better performance for SemER, while positive numbers indicate better performance for slot F1 and intent classification accuracy.

6 Discussion

The limited performance of the reordering method using syntactic information may due to the domain mismatch between our data and the Stanford parser's training data. In particular, our dataset comprises user requests in spoken form, which are on average rather short and often in imperative form starting with a verb. By contrast, the Stanford parser was trained on written news texts which typically comprise comparatively longer sentences which usually do not start with a verb. We expect that an in-domain syntactic parser, which was not available for our experiments, could potentially help to improve the performance of this syntax-based reordering method. Since standard tools developed in Academia may not yield satisfying performance on specific industry datasets, we cannot conclude that this reordering method is generally not useful. In fact, we would assume that it could be quite effective when applied to datasets being closer to the parser's training data domain. Additional experiments are needed to determine in which scenarios gain can be expected and how the approach can potentially be improved. For instance, additional rules for re-ordering based on syntactic parse trees may be developed.

However, as shown with our other approach, large gains are also possible without any additional linguistic or computational resources, making it in particular well suited for low-resource languages. The large improvements obtained by reordering using slot label information also provide a first proof of concept that slot labels are a useful source to address a particular word order divergence scenario. In this paper, we have focused on a simple heuristic which already gave large improvements for zero-shot transfer between English and Hindi. This heuristic may be useful for some other languages with similar divergence, in particular for other Indian languages. For future work, another interesting scenario would be investigating few-shot transfer, in which a (small) amount of supervised data is available in the target language. In this case, it might be possible to learn automatically how to re-order the source data by leveraging the slot label information in (aligned) source and target utterances. This approach can be potentially applied to various word order divergence scenarios and may be applicable for language pairs other than English-Hindi.

7 Conclusion

In this paper, we investigated two computationally inexpensive approaches for addressing word order divergence for English-Hindi zero-shot SLU: one making use of slot label information, and another making use of syntactic parse trees. We presented empirical results on a real-world SLU dataset, showing that by using a simple, yet effective approach for reordering the English source data, large improvements can be achieved. In particular, by reordering using slot label information, consistent gains were achieved across data from six domains with up to 30.13% relative reduction in semantic error rate and up to 22.12% relative gain in F1 for slot filling. This paper also provides a first proof of concept that slot labels in supervised data can be used as a useful source to address a particular word order divergence scenario in SLU. Future work may target additional scenarios, and more sophisticated methods for leveraging slot label information may be developed.

REFERENCES

Steven Cao, Nikita Kitaev, and Dan Klein. Multilingual alignment of contextual word representations, 2020.

Q. Chen, Z. Zhuo, and W. Wang. Bert for joint intent classification and slot filling. *arXiv:1902.10909*, 2019.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. URL http://arxiv.org/abs/1810.04805.

Quynh Do and Judith Gaspers. Cross-lingual transfer learning with data selection for large-scale spoken language understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 1455–1460, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1153. URL https://www.aclweb.org/anthology/D19-1153.

Judith Gaspers, Penny Karanasou, and Rajen Chatterjee. Selecting machine-translated data for quick bootstrapping of a natural language understanding system. *Proceedings of NAACL-HLT*, 2018.

K. He, W. Xu, and Y. Yan. Multi-level cross-lingual transfer learning with language shared and specific knowledge for spoken language understanding. *IEEE Access*, 8:29407–29416, 2020.

Andrew Johnson, Penny Karanasou, Judith Gaspers, and Dietrich Klakow. Cross-lingual transfer learning for Japanese named entity recognition. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Industry Papers)*, pp. 182–189, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-2023. URL https://www.aclweb.org/anthology/N19-2023.

Zihan Liu, Genta Indra Winata, Samuel Cahyawijaya, Andrea Madotto, Zhaojiang Lin, and Pascale Fung. On the importance of word order information in cross-lingual sequence labeling, 2020.

Rudra Murthy, Anoop Kunchukuttan, and Pushpak Bhattacharyya. Addressing word-order divergence in multilingual neural machine translation for extremely low resource languages. 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)*, pp. 3868–3873, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1387. URL https://www.aclweb.org/anthology/N19-1387.

Telmo Pires, Eva Schlinger, and Dan Garrette. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4996–5001, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1493. URL https://www.aclweb.org/anthology/P19-1493.

Ananthakrishnan Ramanathan, Jayprasad Hegde, Ritesh M. Shah, Pushpak Bhattacharyya, and Sasikumar M. Simple syntactic and morphological processing can help English-Hindi statistical machine translation. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*, 2008. URL https://www.aclweb.org/anthology/I08-1067.

Richard Socher, John Bauer, Christopher D. Manning, and Andrew Y. Ng. Parsing with compositional vector grammars. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 455–465, Sofia, Bulgaria, August 2013. Association for Computational Linguistics. URL https://www.aclweb.org/anthology/P13-1045.

Shijie Wu and Mark Dredze. Beto, bentz, becas: The surprising cross-lingual effectiveness of BERT. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 833–844, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-1077. URL https://www.aclweb.org/anthology/D19-1077.

Weijia Xu, Batool Haider, and Saab Mansour. End-to-end slot alignment and recognition for cross-lingual nlu, 2020. URL https://arxiv.org/abs/2004.14353.