Multilingual Semantic Parsing and Code-switching

Long Duong, Hadi Afshar, Dominique Estival, Glen Pink, Philip Cohen, Mark Johnson
Voicebox Technologies
{longd,hadia,dominiquee,glenp,philipc,markj}@voicebox.com

Abstract

Extending semantic parsing systems to new domains and languages is a highly expensive, time-consuming process, so making effective use of existing resources is critical. In this paper, we describe a transfer learning method using crosslingual word embeddings in a sequence-to-sequence model. On the NLmaps corpus, our approach achieves state-of-the-art accuracy of 85.7% for English. Most importantly, we observed a consistent improvement for German compared with several baseline domain adaptation techniques. As a by-product of this approach, our models that are trained on a combination of English and German utterances perform reasonably well on code-switching utterances which contain a mixture of English and German, even though the training data does not contain any code-switching. As far as we know, this is the first study of code-switching in semantic parsing. We manually constructed the set of *code-switching test utterances* for the NLmaps corpus and achieve 78.3% accuracy on this dataset.

1 Introduction

Semantic parsing is the task of mapping a natural language query to a logical form (LF) such as Prolog or lambda calculus, which can be executed directly through database query (Zettlemoyer and Collins, 2005, 2007; Haas and Riezler, 2016; Kwiatkowksi et al., 2010).

Semantic parsing needs application or domain specific training data, so the most straightforward approach is to manufacture training data for each combination of language and application domain. However, acquiring such data is an expensive, lengthy process.

This paper investigates ways of leveraging application domain specific training data in one language to improve performance and reduce the training data needs for the same application domain in another language. This is an increasingly common commercially important scenario, where a single application must be developed for multiple languages simultaneously. In this paper, we investigate the question of transferring a semantic parser from a source language (e.g. English) to a target language (e.g. German). In particular, we examine the situation where there is a large amount of training data for the source language but much less training data for the target language. It is important to note that, despite surface language differences, it has long been suggested that logical forms are the same across languages (Fodor, 1975), motivating transfer learning for this paper.

We conceptualize our work as a form of domain adaptation, where we transfer knowledge about a specific application domain (e.g. navigation queries) from one language to another. Much work has investigated feature-based domain adaptation (Daume III, 2007; Ben-David et al., 2007). However, it is a non-trivial research question to apply these methods to deep learning.

We experiment with several deep learning methods for supervised crosslingual domain adaptation and make two key findings. The first is that we can use a bilingual dictionary to build crosslingual word embeddings, serving as a bridge between source and target language. The second is that machine-translated training data can also be used to effectively improve performance when there is little application domain specific training data in the target language. Interestingly, even when training on the full dataset of the target language, we show that it is still useful to lever-

Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), pages 379–389,
Vancouver, Canada, August 3 - August 4, 2017.

age information from the source language through crosslingual word embeddings. We set new state-of-the-art results on the NLmaps corpus.

Another benefit of joint training of the model is that a single model has the capacity to understand both languages. We show this gives the model the ability to parse code-switching utterances, where the natural language query contains a mixture of two languages. Being able to handle code-switching is valuable in real-world applications that expect spoken natural language input in a variety of settings and from a variety of speakers. Many people around the world are bilingual or multilingual, and even monolingual speakers are liable to use foreign expressions or phrases. Real systems must be able to handle that kind of input, and the method we propose is a simple and efficient way to extend the capabilities of an existing system.

As far as we know, this is the first study of code-switching in semantic parsing. We constructed a new set of *code-switching test utterances* for the NLmaps corpus. Our jointly trained model obtains a logical form exact match accuracy of 78.3% on this test set.

Our contributions are:

- We achieve new state-of-the-art results on the English and German versions of the NLmaps corpus (85.7% and 83.0% respectively).
- We propose a method to incorporate bilingual word embeddings into a sequence-to-sequence model, and apply it to semantic parsing. To the best of our knowledge, we are the first to apply crosslingual word embedding in a sequence-to-sequence model.
- Our joint model allows us to also process input with code-switching. We develop a new dataset for evaluating semantic parsing on code-switching input which we make publicly available.[1]

2 Related work

Deep learning and the sequence-to-sequence approach in particular have achieved wide success in many applications, reaching state-of-the-art performance for semantic parsing (Jia and Liang, 2016; Dong and Lapata, 2016), machine translation (Luong et al., 2015b), image caption generation (Xu et al., 2015), and speech recognition (Chorowski et al., 2014, 2015). Nevertheless, transferring information in a deep learning model about a source language to a target language is still an open research question, and is the focus of this paper.

Our work falls under crosslingual transfer learning category: we want to transfer a semantic parser from one language to another language. The assumption is that there is sufficient application domain specific training data in a source language to train a semantic parser, but only a small amount of application domain specific training data in the target language. We would like to leverage the source language training data to improve semantic parsing in the target language. It is common to exploit the shared structures between languages for POS tagging and Noun Phrase bracketing (Yarowsky and Ngai, 2001), dependency parsing (Täckström et al., 2012; McDonald et al., 2013), named entity recognition (Tsai et al., 2016; Nothman et al., 2013) and machine translation (Zoph et al., 2016). However, as far as we know, there is no prior work on crosslingual transfer learning for semantic parsing, which is the topic of this paper.

There are several common techniques for transfer learning across domains. The simplest approach is *Fine Tune*, where the model is first trained on the source domain and then fine-tuned on the target domain (Watanabe et al., 2016). Using some form of regularization (e.g. L_2) to encourage the target model to remain similar to the source model is another common approach (Duong et al., 2015a). In this approach, the model is trained in the *cascade* style, where the source model is trained first and then used as in a prior when training the target model. It is often beneficial to *jointly* train the source and target models under a single objective function (Collobert et al., 2011; Firat et al., 2016; Zoph and Knight, 2016). Combining source and target data together into a single dataset is a simple way to jointly train for both domains. However, this approach might not work well in the crosslingual case, i.e. transfer from one language to another, because there may not be many shared features between the two languages. We show how to use crosslingual word embeddings (§3.3.1) as the bridge to better share information between languages.

Instead of combining data, a more sophisticated

[1] github.com/vbtagitlab/code-switching

	GeoQuery	ATIS
Number of utterances	880	5410
Jia and Liang (2016)	89.3	83.3
Zettlemoyer and Collins (2007)	86.1	84.6
Kočiský et al. (2016)	87.3	-
Dong and Lapata (2016)	87.1	84.6
Liang et al. (2011)	**91.1**	-
Kwiatkowksi et al. (2010)	88.6	82.8
Zhao and Huang (2015)	88.9	84.2
`TGT Only`	86.1	**86.1**

Table 1: Performance of the baseline attentional model (`TGT Only`) on GeoQuery (Zettlemoyer and Collins, 2005) and ATIS (Zettlemoyer and Collins, 2007) dataset compared with prior work. The best performance is shown in bold.

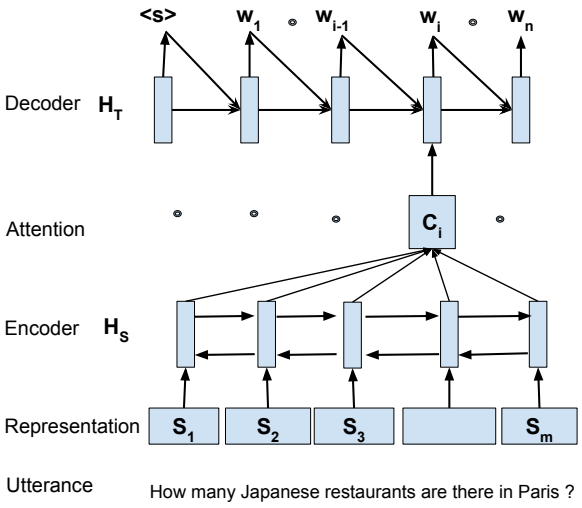


Figure 1: The baseline attentional model as applied to our tasks. The input is the natural language utterance and the output is the logical form.

approach for joint training is to modify the model to adapt for both domains (or languages). Watanabe et al. (2016) propose a dual output model where each output is used for one domain. Kim et al. (2016) extend the feature augmentation approach of Daume III (2007) for deep learning by augmenting different models for each domain. In this paper we experiment with multiple encoders for the sequence-to-sequence attentional model, as described in §3.2. While some of the methods we investigate in this paper have been explored in the domain of syntactic parsing - Tiedemann (2014) used machine translation for cross-lingual transfer, and Ammar et al. (2016) show that a single parser can produce syntactic analyses in multiple languages - our work applies them to semantic parsing.

3 Model

We base our approach on the bidirectional sequence-to-sequence (seq2seq) model with attention of Bahdanau et al. (2014). This attentional model encodes a source as a sequence of vectors, and generates output by decoding these sequences. At each decoding time step, it "attends" to different parts of the encoded sequence.

On a large dataset, it is difficult to improve on a properly tuned seq2seq model with attention. As Table 1 shows, our baseline attentional seq2seq model (described below), which we call `TGT Only` in the figures and tables, achieves competitive results on standard semantic parsing datasets. We begin by describing the basic attentional model and then present our methods for transfer learning to different languages.

3.1 Baseline attentional model

The baseline attentional seq2seq model (`TGT Only`) is shown in Figure 1. The source utterance is represented as a sequence of vectors $S_1, S_2, \ldots, S_m$. Each S_i is the output of an embeddings lookup. The model has two main components: an encoder and a decoder. For the encoder, we use a bidirectional recurrent neural network (RNN) with Gated Recurrent Units (GRU) (Pezeshki, 2015). The source utterance is encoded as a sequence of vectors $H_S = (H_S^1, H_S^2, \ldots, H_S^m)$ where each vector H_S^j ($1 \leq j \leq m$) is the concatenation of the hidden states of the forward and backward GRU at time j.

The attention mechanism is added to the model through an alignment matrix $\alpha \in \mathbb{R}^{n \times m}$, where n is the number of target tokens in the logical form. We add `<s>` and `</s>` to mark the start and end of the target sentence. The "glimpse" vector c_i of the source when generating w_i is $c_i = \sum_j \alpha_{ij} H_S^j$. The decoder is another RNN with GRU unit. At each time step, the decoder RNN receives c_i in addition to the previously-output word. Thus, the hidden state[2] at time i of the decoder is defined as $H_T^i = \text{GRU}(H_T^{i-1}, c_i, w_{i-1})$, which is used to

[2]The GRU also carries a memory cell, along with the hidden state; we exclude this from the presentation for clarity of notation.

Train	
Utt-original:	What is the homepage of the cinema Cinéma Chaplin in Paris?
LF-original:	query(area(keyval('name','Paris'),keyval('is_in:country','France')),nwr(keyval('name', 'Cinéma_Chaplin')), qtype(findkey('website')))
Utt-converted:	What is the homepage of the cinema UNK UNK in Paris?
LF-converted:	query(area(keyval('name','Paris'),keyval('is_in:country','France')),nwr(keyval('name','UNK_UNK')), qtype(findkey('website')))
Test	
Utt-original:	Would you tell me the phone number of Guru Balti in Edinburgh?
Utt-converted:	Would you tell me the phone number of UNK UNK in Edinburgh?
LF-predicted:	query(area(keyval(name,City_of_Edinburgh)),nwr(keyval(name,UNK_UNK)),qtype(findkey(phone)))
LF-lexicalised:	query(area(keyval(name,City_of_Edinburgh)),nwr(keyval(name,'Guru_Balti')), qtype(findkey(phone)))

Figure 2: Handling of unknown word at train and test times. Training examples containing capitalised low-frequency words are duplicated: in one copy, the capitalised low-frequency words are kept in both the utterance (Utt-original) and the LF (LF-original), while in the other copy they are replaced with the symbol UNK in both the utterance (Utt-converted) and the LF (LF-converted). At test time, unknown words in the input utterance are replaced with UNK symbols (in Utt-converted); the UNK symbols in the predicted LF (LF-predicted) are then replaced with the unknown words (LF-lexicalised).

predict word w_i:

$$p(w_i \mid w_1 \cdots w_{i-1}, H_S) = \text{softmax}(g(H_T^i)) \quad (1)$$

where g is a linear transformation.

We use 70 dimensions for both the hidden states and memory cells in the source GRUs and 60 dimensions for target GRUs. We train this model using RMSprop (Tieleman and Hinton, 2012) to minimize the negative log-likelihood using a mini-batch of 256 and early stopping on development data. The initial learning rate is 0.002 and is decayed with decay rate 0.1 if we did not observe any improvement after 1000 iterations. The gradients are rescaled if their L_2 norm is greater than 10. We implemented dropout for both source and target GRU units (Srivastava et al., 2014) with input and output dropout rates of 40% and 25% respectively. The initial state of the source GRU is trainable, and the initial state of target GRU is initialized with the final states of the source GRUs. The non-embeddings weights are initialized using Xavier initialization (Glorot and Bengio, 2010). We also tried stacking several layers of GRUs but did not observe any significant improvement. Choice of hyper-parameters will be discussed in more detail in §4.2.

We initialize the word embeddings in the model with pre-trained monolingual word embeddings trained on a Wikipedia dump using word2vec (Mikolov et al., 2013). We use monolingual word embeddings for all models except for the jointly trained model, where we instead use crosslingual word embeddings (§3.3.1).

In order to handle unknown words, during training, all words that are low frequency and capitalized are replaced with the special symbol UNK in both utterance and logical form. Effectively, we target low-frequency named entities in the dataset. This is a simple but effective version of delexicalization, which does not require a named entity recognizer.[3] However, unlike previous work (Jia and Liang, 2016; Gulcehre et al., 2016; Gu et al., 2016), we also retain the original sentence in the training data, which results in a substantial performance improvement. The intuition is that the model is capable of learning a useful signal even for very rare words. During test time, we replace (from left to right) the UNK in the logical form with the corresponding word in the source utterance. Figure 2 shows examples of handling unknown words during training and testing. At train time, the two words *Cinéma* and *Chaplin* are replaced with UNK in both utterance and logical form. At test time, the first and second UNK in the logical form are replaced with the unknown words *Guru* and *Balti* from the test utterance. We implement this attentional model as our baseline. We now detail our methods for transferring learning to other languages.

[3]Using named entity recognition would be another solution but we did not want to rely on additional resources.

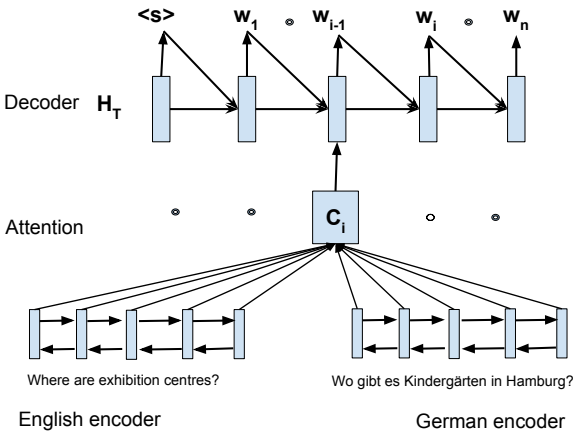


Figure 3: Dual encoder model where each language has a separate encoder but both share the same decoder. Each training mini-batch only has monolingual input, so only one encoder is used for each mini-batch.

3.2 `Dual` encoder model

Multi-task learning is a common approach for neural domain adaptation (Watanabe et al., 2016; Duong et al., 2015b; Collobert et al., 2011; Luong et al., 2015a). In this approach, the source and target domains are jointly trained under a single objective function. The idea is that many parameters can be shared between the source and target domains, and the errors in the source domain can inform the target domain and vice versa. Following this idea, we extend the baseline attentional model (§3.1) to dual encoders, one for the source language and another for the target language. In this work, we perform the evaluation with English and German as both source and target languages, i.e. in both directions (depending on the model). The decoder is shared across languages as shown in Figure 3. We refer to this as our `Dual` model. The glimpse vector c_i will be calculated using either the source or target RNN encoder, motivated by the fact that both source and target languages use the same target logical form. The model is trained on the combined data of both the source and target languages. For each mini-batch, we only use the source or target language data, and make use of the corresponding RNN encoder.

3.3 `All` model

Another straightforward domain adaptation technique is to combine the source and target language data. We create a new training data set $D_{all} = D_s \cup D_t$ where D_s and D_t are the training data for source and target language. We refer to this as our `All` model. The `All` model is a `Dual` model, but both source and target RNNs are shared and only the embedding matrices are different between source and target languages.

3.3.1 Crosslingual word embeddings

Overcoming lexical differences is a key challenge in crosslingual domain adaptation. Prior work on domain adaptation found features that are common across languages, such as high-level linguistic features extracted from the World Atlas of Language Structures (Dryer and Haspelmath, 2013), crosslingual word clusters (Täckström et al., 2012) and crosslingual word embeddings (Ammar et al., 2016). Here, we extend crosslingual word embeddings as the crosslingual features for semantic parsing.

We train crosslingual word embeddings across source and target languages following the approach of Duong et al. (2016), who achieve high performance on several monolingual and crosslingual evaluation metrics. Their work is essentially a multilingual extension of word2vec, where they use a context in one language to predict a target word in another language. The target words in the other language are obtained by looking up that word in a bilingual dictionary. Thus, the input to their model is monolingual data in both languages and a bilingual dictionary. We use monolingual data from pre-processed Wikipedia dump (Al-Rfou et al., 2013) with bilingual dictionary from Panlex (Kamholz et al., 2014).

We initialize the seq2seq source embeddings of both languages with the crosslingual word embeddings. However, we *do not update* these embeddings. We apply crosslingual word embeddings (`+XlingEmb`) to the `All` model (§3.3) and the `Dual` encoder model (§3.2) and jointly train for the source and target language. For other models described in this paper, we initialize with monolingual word embeddings.

3.4 `Trans` model

The above crosslingual word embeddings need a bilingual dictionary to connect between the source and target language. In addition, we can also leverage a machine translation system as the connection between languages. For this case, we define a `Trans` model, which applies the baseline attentional model with training data D_{trans} =

English utterance (from NLmaps)	How many universities are there in Paris?
German utterance (from NLmaps)	Wie viele Universitäten hat Paris?
Code-switching (constructed)	Wie viele Universitäten are there in Paris?
Logical form	query(area(keyval('name','Paris'),keyval('is_in:country','France')), nwr(keyval('amenity','university')),qtype(count))

Table 2: Example of data from the NLmaps corpus. The English and German utterances are translations of each other and they share the same logical form. We constructed code-switching utterances for all the logical forms in the NLmaps test corpus.

$\texttt{translate}(D_s) \cup D_t$, where `translate` is the function to translate the data from the source language to the target language. For the experiments reported in this paper, we use Google Translate (Wu et al., 2016).

4 Experiments

In this section, we evaluate the methods proposed in §3 for transfer learning for semantic parsing. The aim is to build a parser for a target language with minimum supervision given application domain specific training data for a source language. The question we want to answer is whether it is possible to share information across languages to improve the performance of semantic parsing.

4.1 Dataset

We use the NLmaps corpus (Haas and Riezler, 2016), a semantic parsing corpus for English and German. We evaluated our approach on this corpus because it is the only dataset which provides data in both English and German. Table 2 presents typical examples from this dataset, together with a constructed code-switching utterance. Utterances from different languages are assigned the same logical forms, thus motivating the approach taken in this paper. We tokenize in way similar to Kočiský et al. (2016).[4] For each language, the corpus contains 1500 pairs of natural language queries and corresponding logical forms for training and 880 pairs for testing. We use 10% of the training set as development data for early stopping and hyper-parameter tuning. For evaluation, we use exact match accuracy for the logical form (Kočiský et al., 2016).

4.2 Hyper-parameter tuning

Hyper-parameter tuning is important for good performance. We tune the baseline attentional model (§3.1) on the development data by generating 100 configurations which are permutations of different optimizers, source and target RNN sizes, RNN cell type[5], dropout rates and mini-batch sizes. We then use the same configuration for all other models.

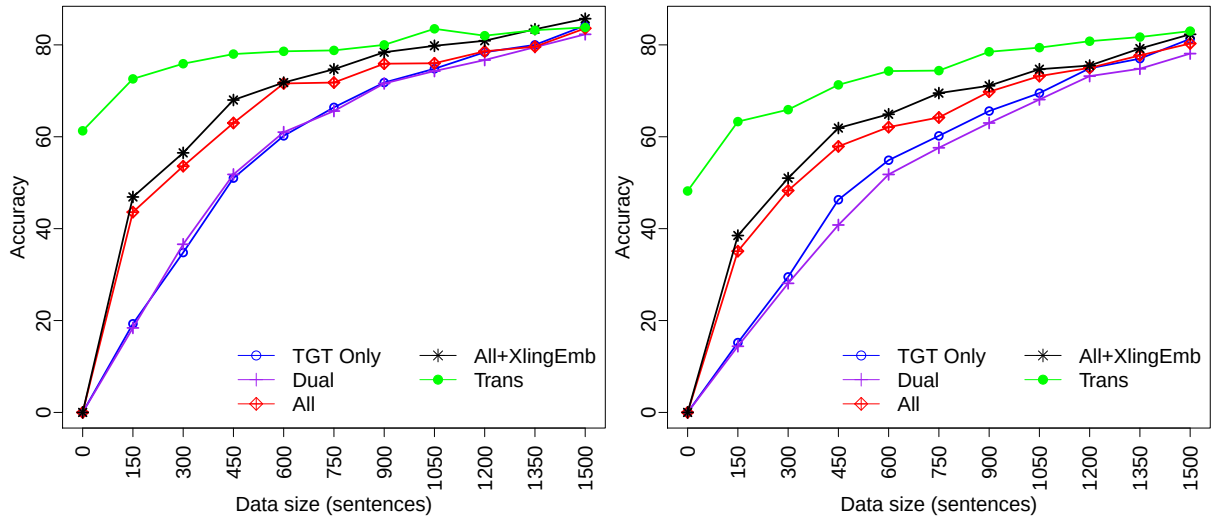
4.3 Learning curves

We experimented with transfer learning from $English \rightarrow German$ and $German \rightarrow English$. We use all the data in the NLmaps corpus for the source language and vary the amount of data for the target language. Figure 4 shows the learning curve for transfer learning in both directions.

The first observation is that the baseline attentional model trained on the target only (`TGT Only`) is very robust when trained on large datasets but performs poorly on small datasets. The `Dual` model performs similarly to the baseline attentional model for English and slightly worse for German. The simple method of combining the data (`All` model) performs surprisingly well, especially on small datasets where this model is $\approx 20\%$ better than the baseline attentional model for both languages. Incorporating crosslingual word embeddings (`+XlingEmb`) consistently improves the performance for all data sizes. The improvement is more marked for the $English \rightarrow German$ direction. Finally, if we have a machine translation system, we can further improve the performance on a target language by augmenting the data with translations from the source language. This simple method substantially improves performance on a target language, especially in the small dataset scenario. More surprisingly, if we don't use any target language data and train on $D_{trans} = \texttt{translate}(D_s)$ we achieve 61.3% and 48.2% accuracy for English and German respectively (Figure 4). This corresponds to a distant supervision baseline where the

[4] We remove quotes, add spaces around parenthesis and separate the question mark at the end of the utterance.

[5] We tried with LSTM, GRU and Highway networks.



(a) *German → English* (b) *English → German*

Figure 4: Learning curves for English and German with various models. `TGT Only` applies the baseline attentional mode (§3.1) to the target language data alone. `Dual` uses the dual encoders from §3.2. `All` is similar with `TGT Only` but trained on the combined data of both languages. `All+XlingEmb`, instead of monolingual word embeddings, uses crosslingual word embeddings (§3.3.1). `Trans` model uses a machine translation system (§3.4). At 1500 sentences, since we do not have development data for early stopping, we train the model for exactly 10k iterations.

training data is "silver standard" given by a machine translation system. This baseline is equivalent to supervised learning on 600 and 450 gold sentences on English and German respectively.

We also tried several other models such as *Fine Tune*, where the model is trained on the source language first and then fine tuned on the target language but the performance is similar to the `TGT Only` model. The other baseline we implemented is L_2, where we use the source language model as the prior to the target language objective function through an L_2 regularization. However, we did not observe any performance gain, as was also noticed by Watanabe et al. (2016).

	English	German
Haas and Riezler (2016)	68.3	-
Kočiský et al. (2016)	78.0	-
`Dual`	82.3	78.1
`TGT Only`	84.2	81.3
`All`	83.6	80.3
`All+XlingEmb`	**85.7**	82.3
`Trans`	83.8	**83.0**

Table 3: Results on the full datasets. The best result is shown in bold.

4.4 Discussion

Figure 4 shows the learning curves at various data points. Table 3 presents the results for models trained on all target language data (1500 sentences). The `Dual` encoder performs the worst. The baseline supervised learning on target data only (`TGT Only`) performs surprisingly well, probably because it is highly tuned. When training on combined English and German data (`All` model), we observe a slight decrease in performance for both English and German. Even when training on the full target language dataset, using crosslingual word embeddings improves the performance by about 2% in both English and German which highlights the effectiveness of crosslingual word embeddings. As shown in Figure 4, adding a machine translation system helps immensely for small datasets. On a full dataset, however, we only observe a small improvement for German but degradation in performance for English using `Trans` model. This might be because machine translations are hardly perfect. With a high level of confidence when training on full dataset, added translations do not contribute much to the model. Importantly, however, these results are substantially better than the previous state-of-the-art result reported in Kočiský et al. (2016).

Model	Accuracy
German `TGT Only`	14.5
English `TGT Only`	16.3
`All`	76.8
`All+XlingEmb`	**78.3**

Table 4: Accuracy of seq2seq models on the code-switching test utterances. The monolingual English and German seq2seq models (`TGT Only`) are trained only on English and German utterances respectively, while the `All` and `All+XlingEmb` models are trained on both sets of utterances. The best result is shown in bold.

5 Code-switching

An interesting result is that by jointly training the model on both English and German, we can now also handle *code-switching* data, where a natural language utterance is a mixture of English and German. We evaluate our jointly trained model's ability to parse utterances consisting of both English and German on our manually constructed code-switching testset.[6] An example of constructed code-switching utterance is shown in Table 2. Note that our models are only trained on "pure" English and German utterances; there are no code-switching training examples in the input.

Code-switching is a complex linguistic phenomenon and there are different accounts of the socio-linguistic conventions governing its use (Poplack, 2004; Isurin et al., 2009; MacSwan, 2017), as well as of the structural properties of utterances with code-switching (Joshi, 1982). Here we focus on the simple kind of code-switching where a single phrase is produced in a different language than the rest of the utterance. Our dataset was created by a fluent bilingual speaker who generated code-switching utterances for each of the 880 examples in the NLmaps test set. Approximately half of the utterances are "Denglish" (i.e., a German phrase embedded in an English matrix sentence) and half are "Gamerican" (an English phrase embedded in a German matrix sentence). NLmaps includes English and German utterances for each test example, and where possible our code-switching utterance was a combination of these (some of our code-switching examples diverge from the corresponding English and German utterances if this improves fluency).

Table 4 presents the results of our models on this new test set. This makes clear that the `All+XlingEmb` model performs noticeably better than the baseline monolingual models on the code-switching test examples, even though there were no such examples in the training set.

6 Conclusion

In this paper, we investigate ways to transfer information from one (source) language to another (target) language in a single semantic parsing application domain. This paper compared various transfer learning models with a strong sequence-to-sequence baseline. We found that a simple method of combining source and target language data works surprisingly well, much better than more complicated methods such as a `Dual` model or L_2 regularization. If bilingual dictionaries are available, crosslingual word embeddings can be constructed and used to further improve the performance. We observed $\approx 20\%$ improvement for small datasets compared to the strong baseline attentional model. Moreover, this improvement can almost be doubled if we leverage some machine translation system. Even on the full dataset, our jointly trained model with crosslingual word embeddings gives state-of-the-art results for semantic parsing of the English and German versions of NLmaps corpus.

This paper also investigated the performance of semantic parsers on *code-switching utterances* that combine English and German. We created a new code-switching test set, and showed that our simple jointly trained model with crosslingual word embeddings achieves 78.3% exact match accuracy on this set, which is more than 60% better than a corresponding monolingual sequence-to-sequence model.

For future work, we would like to try delexicalization as part of training and experiment with better ways of handling unknown word such as a copy mechanism (Jia and Liang, 2016; Gu et al., 2016; Gulcehre et al., 2016). Investigating a more sophisticated network architecture that can perform multilingual semantic parsing more accurately, or with less training data is another fruitful research direction. This work has only scratched the surface in terms of code switching. We would like to exploit the pragmatic and socio-linguistic context to better handle code-switching.

[6] github.com/vbtagitlab/code-switching

Acknowledgments

We thank the anonymous reviewers for their insightful comments.

References

Rami Al-Rfou, Bryan Perozzi, and Steven Skiena. 2013. Polyglot: Distributed word representations for multilingual NLP. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*. pages 183–192.

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

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR* abs/1409.0473.

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. 2007. Analysis of representations for domain adaptation. In P. B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, MIT Press, pages 137–144.

Jan Chorowski, Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. End-to-end continuous speech recognition using attention-based recurrent NN: first results. *CoRR* abs/1412.1602.

Jan Chorowski, Dzmitry Bahdanau, Dmitriy Serdyuk, KyungHyun Cho, and Yoshua Bengio. 2015. Attention-based models for speech recognition. *CoRR* abs/1506.07503.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *J. Mach. Learn. Res.* 12:2493–2537.

Hal Daume III. 2007. Frustratingly easy domain adaptation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Association for Computational Linguistics, pages 256–263.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 33–43.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online – walls.info*. Max Planck Institute for Evolutionary Anthropology.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015a. 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*. pages 845–850.

Long Duong, Trevor Cohn, Steven Bird, and Paul Cook. 2015b. A neural network model for low-resource universal dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 339–348.

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

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pages 866–875.

Jerry A. Fodor. 1975. *The Language of Thought*. Harvard University Press.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In Yee Whye Teh and Mike Titterington, editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. PMLR, volume 9 of *Proceedings of Machine Learning Research*, pages 249–256.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. pages 1631–1640.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 140–149.

Carolin Haas and Stefan Riezler. 2016. A corpus and semantic parser for multilingual natural language querying of openstreetmap. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 740–750.

Ludmila Isurin, Donald Winford, and Kees De Bot. 2009. *Multidisciplinary Approaches to Code Switching*. John Benjamins Publishing.

Robin Jia and Percy Liang. 2016. Data recombination for neural semantic parsing. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, pages 12–22.

Aravind K Joshi. 1982. Processing of sentences with intra-sentential code-switching. In *Proceedings of the 9th conference on Computational linguistics-Volume 1*. Academia Praha, pages 145–150.

David Kamholz, Jonathan Pool, and Susan Colowick. 2014. PanLex: Building a resource for panlingual lexical translation. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. European Language Resources Association, pages 3145–50.

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2016. Frustratingly easy neural domain adaptation. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*. pages 387–396.

Tomáš Kočiský, Gábor Melis, Edward Grefenstette, Chris Dyer, Wang Ling, Phil Blunsom, and Karl Moritz Hermann. 2016. Semantic parsing with semi-supervised sequential autoencoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1078–1087.

Tom Kwiatkowksi, Luke Zettlemoyer, Sharon Goldwater, and Mark Steedman. 2010. Inducing probabilistic CCG grammars from logical form with higher-order unification. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pages 1223–1233.

Percy Liang, Michael Jordan, and Dan Klein. 2011. Learning dependency-based compositional semantics. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 590–599.

Minh-Thang Luong, Quoc V. Le, Ilya Sutskever, Oriol Vinyals, and Lukasz Kaiser. 2015a. Multi-task sequence to sequence learning. *CoRR* abs/1511.06114.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015b. Effective approaches to attention-based neural machine translation. In *Proc. Empirical Method in Natural Language Processing (EMNLP)*. pages 1412–1421.

Jeff MacSwan. 2017. A multilingual perspective on translanguaging. *American Educational Research Journal* 54(1):167–201.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal dependency annotation for multilingual parsing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*. pages 92–97.

Tomas Mikolov, Wen-tau Yih, and Geoffrey Zweig. 2013. Linguistic regularities in continuous space word representations. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pages 746–751.

Joel Nothman, Nicky Ringland, Will Radford, Tara Murphy, and James R. Curran. 2013. Learning multilingual named entity recognition from Wikipedia. *Artificial Intelligent* 194:151–175.

Mohammad Pezeshki. 2015. Sequence modeling using gated recurrent neural networks. *CoRR* abs/1501.00299.

Shana Poplack. 2004. Code-switching. In *Soziolinguistik: an international handbook of the science of language (2nd edition)*, Walter de Gruyter, pages 589–596.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15:1929–1958.

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. 2012. Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 477–487.

Jörg Tiedemann. 2014. Rediscovering annotation projection for cross-lingual parser induction. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*. Dublin City University and Association for Computational Linguistics, Dublin, Ireland, pages 1854–1864. http://www.aclweb.org/anthology/C14-1175.

T. Tieleman and G. Hinton. 2012. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning.

Chen-Tse Tsai, Stephen Mayhew, and Dan Roth. 2016. Cross-lingual named entity recognition via Wikification. In *The SIGNLL Conference on Computational Natural Language Learning*.

Yusuke Watanabe, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2016. Domain adaptation for neural networks by parameter augmentation. In *Proceedings of the 1st Workshop on Representation Learning for NLP*. Association for Computational Linguistics, pages 249–257.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR* abs/1609.08144.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. pages 2048–2057.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics on Language technologies*. pages 1–8.

Luke Zettlemoyer and Michael Collins. 2007. Online learning of relaxed CCG grammars for parsing to logical form. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*. pages 678–687.

Luke S. Zettlemoyer and Michael Collins. 2005. Learning to map sentences to logical form: Structured /classification with probabilistic categorial grammars. In *UAI '05, Proceedings of the 21st Conference in Uncertainty in Artificial Intelligence, Edinburgh, Scotland, July 26-29, 2005*. pages 658–666.

Kai Zhao and Liang Huang. 2015. Type-driven incremental semantic parsing with polymorphism. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Association for Computational Linguistics, pages 1416–1421.

Barret Zoph and Kevin Knight. 2016. Multi-source neural translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. pages 30–34.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. pages 1568–1575.