El Volumen Louder Por Favor: Code-switching in Task-oriented Semantic Parsing

Arash Einolghozati Abhinav Arora Lorena Sainz-Maza Lecanda
Anuj Kumar Sonal Gupta
Facebook
{arashe,abhinavarora,lorenasml,anujk,sonalgupta}@fb.com

Abstract

Being able to parse code-switched (CS) utterances, such as Spanish+English or Hindi+English, is essential to democratize task-oriented semantic parsing systems for certain locales. In this work, we focus on Spanglish (Spanish+English) and release a dataset, CSTOP, containing 5800 CS utterances alongside their semantic parses. We examine the CS generalizability of various Cross-lingual (XL) models and exhibit the advantage of pre-trained XL language models when data for only one language is present. As such, we focus on improving the pre-trained models for the case when only English corpus alongside either zero or a few CS training instances are available. We propose two data augmentation methods for the zero-shot and the few-shot settings: fine-tune using translate-and-align and augment using a generation model followed by match-and-filter. Combining the few-shot setting with the above improvements decreases the initial 30-point accuracy gap between the zero-shot and the full-data settings by two thirds.

1 Introduction

Code-switching (CS) is the alternation of languages within an utterance or a conversation (Poplack, 2004). It occurs under certain linguistic constraints but can vary from one locale to another (Joshi, 1982). We envision two usages of CS for virtual assistants. First, CS is very common in locales where there is a heavy influence of a foreign language (usually English) in the native "substrate" language (e.g., Hindi or Latin-American Spanish). Second, for other native languages, the prevalence of English-related tech words (e.g., Internet, screen) or media vocabulary (e.g., movie names) is very common. While in the second case, a model using contextual understanding should be able to parse the utterance, the first form of CS, which is our focus in this paper, needs Cross-Lingual(XL) capabilities in order to infer the meaning.

There are various challenges for CS semantic parsing. First, collecting CS data is hard because it needs bilingual annotators. This gets even worse considering that the number of CS pairs grows quadratically. Moreover, CS is very dynamic and changes significantly by occasion and in time (Poplack, 2004). As such, we need extensible solutions that need little or no CS data while having the more commonly-accessible English data available. In this paper, we first focus on the zero-shot setup for which we only use EN data for the same task domains (we call this in-domain EN data). We show that by translating the utterances to ES and aligning the slot values, we can achieve high accuracy on the CS data. Moreover, we show that having a limited number of CS data alongside augmentation with synthetically generated data can significantly improve the performance.

Our contributions are as follows: 1) We release a code-switched task-oriented dialog data set, CSTOP[1], containing 5800 Spanglish utterances and a corresponding parsing task. To the best of our knowledge, this is the first Code-switched parsing dataset of such size that contains utterances for both training and testing. 2) We evaluate strong baselines under various resource constraints. 3) We introduce two data augmentation techniques that improve the code-switching performance using monolingual data.

2 Task

In task-oriented dialog, the language understanding task consists of classifying the intent of an utterance, i.e., sentence classification, alongside tagging the slots, i.e., sequence labeling. We use the Task-

[1]The dataset can be downloaded from https://fb.me/cstop_data

Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics, pages 1009–1021
April 19 - 23, 2021.

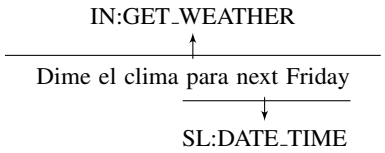

Figure 1: Example CS sentence and its annotation for the sequence [IN:GET_WEATHER Dime el clima [SL:DATE_TIME para next Friday]]

Oriented Parsing dataset released by Schuster et al. (2018) as our EN monolingual dataset. We release a similar dataset, CSTOP, of around 5800 Spanglish utterances over two domains, Weather and Device, which are collected and annotated by native Spanglish speakers. An example from the CSTOP alongside its annotation is shown in Fig. 1. Note that the intent and slot lables start with $IN:$ and $SL:$, respectively. Our task is to classify the sentence intent, here *IN:GET_WEATHER* as well as the label and value of the slots, here *SL:DATE_TIME* corresponding to the span *para next Friday*. Moreover, other words are classified as having no label, i.e., *O* class. We discuss the details of this dataset in the next section.

One of the unique challenges of this task, compared with common NER and language identification CS tasks, is the constant evolution of CS data. Since the task is concerned with spoken language, the nature of CS is very dynamic and keeps evolving from domain to domain and from one community to another. Furthermore, cross-lingual data for this task is also very rare. Most of the existing techniques, either combine monolingual representations (Winata et al., 2019a) or combine the datasets to synthesize code-switched data (Liu et al., 2019). Lack of monolingual data for the substrate language (very realistic if you replace ES with a less common language) would make those techniques inapplicable.

In order to evaluate the model in a task-oriented dialog setting, we use the exact-match accuracy (from now on, accuracy) as the primary metric. This is simply defined as the percentage of utterances for which the full parse, i.e., the intent and all the slots, have been correctly predicted.

3 CSTOP Dataset

In this section, we provide details of the CSTOP dataset. We originally collected around 5800 CS utterances over two domains; *Weather* and *Device*. We picked these two domains as they represent complementary behavior. While Weather contains slot-heavy utterances (average 1.6 slots per utterance), Device is an intent-heavy domain with only average 0.8 slots per utterance. We split the data into 4077, 1167, and 559 utterances for training, testing, and validation, respectively.

CS data collection proceeded in the following steps:

1. One of the authors, who is a native speaker of Spanish and uses Spanglish on a daily basis, generated a small set of CS utterances for Weather and Device domains. Additionally, we also recruited bilingual EN/ES speakers who met our Spanglish speaker criteria guidelines, established following Escobar and Potowski (2015).

2. We wrote Spanglish data creation instructions and asked participants to produce Spanish-English CS utterances for each intent (i.e. ask for the weather, set device brightness, etc).

3. Next, we filter out utterances from this pool to only retain those that exhibited true intra-sentential CS.

4. The collected utterances were labeled by two annotators, who identified the intent and slot spans. If the two annotators disagreed on the annotation for an utterance, a third annotator would resolve the disagreement to provide a final annotation for it.

Table. 1 shows the number of distinct intents and slots for each domain and the number of utterances in CSTOP for each domain. We have also shown the most 15 common intents in the training set and a representative Spanglish example alongside its slot values for those intents in Table. 2. The first value in a slot tuple is the slot label and the second is the slot value. We can see that while most of the verbs and stop words are in Spanish, Nouns and slot values are mostly in English. We further calculate the prevalence Spanish and English words by using a vocabulary file of 20k for each language. Each token in the CSTOP training set is assigned to the language for which that token has a lower rank. The ratio of the Spanish to English tokens is around 1.34 which matches our previous anecdotal observation. This ratio was consistent when increasing the vocabular size to even 40k. .

Domain	intents	slots	utterances
Weather	2	4	3692
Device	17	6	2112

Table 1: CSTOP Statistics

4 Model

Our base model is a bidirectional LSTM with separate projections for the intent and slot tagging (Yao et al., 2013). We use the aligned word embedding MUSE (Conneau et al., 2017) with a vocabulary size of 25k for both EN and ES. Our experiments showed that for the best XL generalization, it's best to freeze the word embeddings when the training data contains only EN or ES utterances. We refer to this model as simply MUSE.

We also use SOTA pre-trained XL models; XLM (Conneau and Lample, 2019) and XLM-R (Conneau et al., 2019). These models are pre-trained via Masked Language Modeling (MLM) (Devlin et al., 2019) on massive multilingual data. They share the word-piece token representation, BPE (Sennrich et al., 2016) and SentencePiece (Kudo and Richardson, 2018), as well as a common MLM transformer for different languages. Moreover, while XLM is pre-trained on Wikipedia, XLM-R is trained on crawled web data which contains more non-English and possibly CS data. In order to adapt these models for the joint intent classification and slot tagging task, we use the method described in Chen et al. (2019). For classification, we add a linear classifier on top of the first hidden state of the Transformer. A typical slot tagging model feeds the hidden states, corresponding to each token, to a CRF layer (Mesnil et al., 2015). To make this compatible with XLM and XLM-R, we use the hidden states corresponding to the first sub-word of every token as the input to the CRF layer.

Table 3 shows the accuracy of the above models on CSTOP. We also have listed the performance when the models were first fine-tuned on the EN data (CS+EN). We observe that in-domain fine-tuning can almost halve the gap between XLM-R and XLM, which is around 50% faster during the inference than XLM-R during inference. The training details for all our models and the validation results are listed in the Appendix.

5 Zero-shot performance

Bottom part of Table 3 shows the CS test accuracy when using only the in-domain monolingual data. Our EN dataset is the task-oriented parsing dataset (Schuster et al., 2018) described in the previous section. Since the original TOP dataset did not include any utterances belonging to the Device domain, we also release a dataset of around thousand EN Device utterances for the experiments using the EN data. In order to showcase the effect of monolingual ES data, we also experiment with using the in-domain ES dataset, i.e. ES Weather and Device queries.

We observe that having monolingual data of both languages yields very high accuracy, only a few points shy of training directly on the CS data. Moreover, in this setting, even simpler models such as MUSE can yield competitive results with XLM-R while being much faster. However, the advantage of XL pre-training becomes evident when only one of the languages is present. As such, having only the substrate language (i.e., ES) is almost the same as having both languages for XLM-R.

Note that we do not use ES data for other results in this paper. Obtaining semantic parsing dataset in another language is expensive and often only EN data is available. Our experiments show a huge performance gap when only using the EN data, and thus in this paper, we will be focusing on using the EN data alongside zero or a few CS instances.

5.1 Effect of XL Embeddings

Here, we explore how much of the zero-shot performance can be attributed to the XL embeddings as opposed to the shared XL representation. As such, we experiment with replacing MUSE embeddings with other embeddings in the LSTM model explained in the previous section. We experiment with the following strategies:: (1) Random embedding: This learns the ES and EN word embeddings from the scratch (2) Randomly-initialized SentencePiece (Kudo and Richardson, 2018) (RSP): Words are represented by wordpiece tokens that are learned from a huge unlabeled multilingual corpus. (3) Pre-trained XLM-R sentence piece (XLSP). These are the 250k embedding vectors that are learned during the pre-trainig of XLM-R.

We have shown the effects of using the aforementioned embeddings in the zero-shot setting in Table 4. We can see that by having monolingual datasets from both languages, even random

intent	utterance	slots
GET_WEATHER	¿cómo estará el clima en Miami este weekend?	(LOCATION, Miami), (DATE_TIME, este weekend)
UNSUPPORTED_WEATHER	how many centimeters va a llover hoy	(DATE_TIME, hoy)
OPEN_RESOURCE	Abreme el gallery	(RESOURCE, el gallery)
CLOSE_RESOURCE	Cierra maps	(RESOURCE, maps)
TURN_ON	Prende el privacy mode	(COMPONENT, el privacy mode)
TURN_OFF	Desactiva el speaker	(COMPONENT, el speaker)
WAKE_UP	Quita sleep mode	-
SLEEP	prende el modo sleep	-
OPEN_HOMESCREEN	Go to pagina de inicio	-
MUTE_VOLUME	Desactiva el sound	-
UNMUTE_VOLUME	Prende el sound	-
SET_BRIGHTNESS	subir el brigtness al 80	(PERCENT, 80)
INCREASE_BRIGHTNESS	Ponlo mas bright	-
DECREASE_BRIGHTNESS	baja el brightness	-
SET_VOLUME	Turn the volumen al nivel 10	(PRECISE_AMOUNT,10)
INCREASE_VOLUME	aumenta el volumen a little bit	-
DECREASE_VOLUME	Bájale a la music	-

Table 2: Examples from CSTOP intents

Lang/Model	MUSE	XLM	XLM-R
CS	87.0	86.6	94.4
CS + EN	88.1	93.0	95.4
EN	39.2	54.8	66.6
ES	69.9	78.3	88.1
EN+ES	88.2	87.8	91.2

Table 3: Full-training (top) and zero-shot (bottom) accuracy of XL models when using different monolingual corpora. ES is an internal dataset to showcase the effect of having a big Spanish corpus.

	Random	RSP	XLSP	MUSE
EN	13.5	12.2	30.3	**39.2**
ES	38.2	48.0	**70.5**	**69.9**
EN+ES	81.1	84.3	**89.0**	88.2

Table 4: Zero-shot accuracy for simple LSTM model when using different monolingual corpora and different embedding strategies.

embeddings can yield high performance. By removing one of the languages, unsurprisingly, the codeswitching generalizability drops sharply for all, but much less for XLSP and MUSE. Moreover, even though the XLSP embeddingsm, unlike MUSE, is not consttrained to only EN and ES, it yields comparable results with the word-based MUSE embeddings.

We can also see that When ES data is available, RSP provides some codeswitching generalizability, as compared with the Random strategy, but not when only EN data is available. We hypothesize that the common sub-word tokens are more helpful to generalize the slot values (which in the codeswitched data are mostly in EN) than the non-slot queries which are more commonly in ES. This is also verified by the observation that most of the gains for the RSP vs Random for the ES only scenario come from the slot tagging accuracy as compared with the intent detection.

As a final note, we observe that between $20-30\%$ of the XLM-R gains can be captured by using the pre-trained sentence-piece embeddings while the rest are coming from the shared XL representation pre-trained on massive unlabeled data. In the rest of the paper, we focus on the XLM-R model.

6 Data Augmentation Approaches

In this section, we discuss two data augmentation approaches. The first one is in a zero-shot setting and only uses EN data to improve the performance on the Spanglish test set. In the second approach, we assume having a limited number of Spanglish data and use the EN data to augment the few-shot setting.

6.1 Translate and Align

We explore creating synthetic ES data from the EN dataset using machine translation. Since our task is a joint intent and slot tagging task, creating a synthetic ES corpus consists of two parts: a) Obtaining a parallel EN-ES corpus by machine translating utterances from EN to ES, b) Projecting gold annotations from EN utterances to their ES counterparts via word alignment (Tiedemann et al., 2014; Lee et al., 2019b). Once the words in both languages are aligned, the slot annotations are simply copied

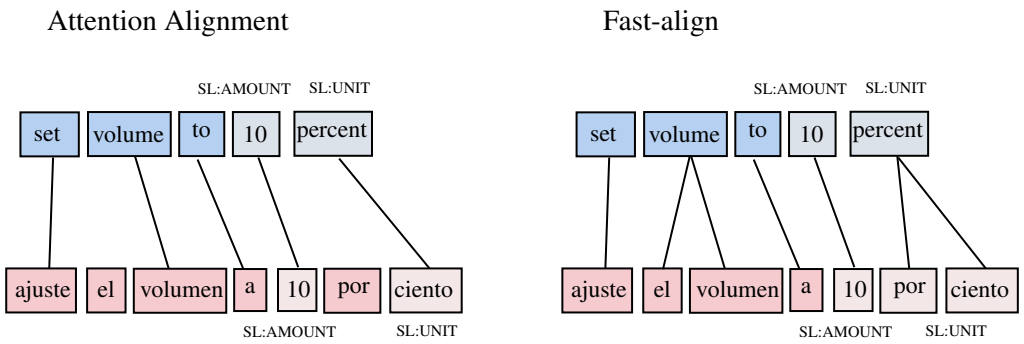

Figure 2: An example comparison between the two methods of slot label projection. The image in on the left shows Attention alignment, where every source token gets projected to a single target token. As a result, *percent*, in EN is aligned only with *ciento* in ES. The image on the right shows fast-align, which allow a many-to-many alignment. Hence *percent* is correctly aligned with *por ciento*.

over from EN to ES by word alignment. For word alignment, we explore two methods that are explained below. In some cases, word alignment may produce discontinuous slot tokens in ES, which we handle by introducing new slots of the same type, for all discontinuous slot fragments.

Our first method leverages the attention scores (Bahdanau et al., 2015) obtained from an existing EN to ES NMT model. We adopt a simplifying assumption that each source word is aligned to one target language word (Brown et al., 1993). For every slot token in the source language, we select the highest attention score to align it with a word in the target language.

Our next approach to annotation projection makes use of unsupervised word alignment from statistical machine translation. Specifically, we use the fast-align toolkit (Dyer et al., 2013) to obtain alignments between EN and ES tokens. Since fast-align generates asymmetric alignments, we generate two sets of alignments, EN to ES and ES to EN and symmetrize them using the grow-diagnol-final-and heuristic (Koehn et al., 2003) to obtain the final alignments.

In Table 5, we show the CS zero-shot accuracy when fine-tuning on the newly generated ES data (called ES*.) alongside the original EN data. We can see that unsupervised alignment results in around 2.5 absolute point accuracy improvement. On the other hand, using attention alignment ends up hurting the accuracy, which is perhaps due to the slot noise that it introduces. The assumption that a single source token aligns with a single target token leads to incorrect data annotations when the length of a translated slot is different in EN and ES. Figure 3 shows an example utterance where attention alignment produces an incorrect annotation compared to unsupervised alignment.

EN	EN+ES* Attn	EN+ES* aligned
66.6	65.8	69.2

Table 5: Zero-shot accuracy when fine-tuning XLM-R on EN monolignual data as well as the auto-translated and aligned ES data (called ES*).

6.2 Generate by Match-and-Filter in the Few-shot Setting

Here, we assume having a limited number of high-quality in-domain CS data and as such, we construct the CSTOP_{100} dataset of around 100 utterances from the original training set in the CSTOP. We make sure that every individual slot and intent (but not necessarily the combination) is presented in CSTOP_{100} and randomly sample the rest. We perform our sampling three times and report the few-shot results on the average performance. This setting is of paramount importance for bringing up a domain in a new locale when the EN data is already available. The first column in Table 6 shows the CS Few-Shot (FS) performance alongside the fine-tuning on the EN data and the aligned translated data, when average over three sampling of CSTOP_{100}.

In order to improve the FS performance, we perform data augmentation on the CSTOP_{100} dataset. Unlike methods such as Pratapa et al. (2018), we seek generic methods that do not need extra resources such as constituency parsers. Instead, we explore using pre-trained generative models while taking advantage of the EN data.

We use BART (Lewis et al., 2019), a denois-

Model/Training Data	Few Shot	Few shot+ Generate and Filter augmentation
XLM-R	61.2	70.3
XLM-R fine-tuned on EN	82.6	83.7
XLM-R fine-tuned on EN+ES*	84.1	**84.8**

Table 6: Accuracy when only a few CS instances are available during training, with and without the data augmentation. ES* is the auto-translated and aligned data.

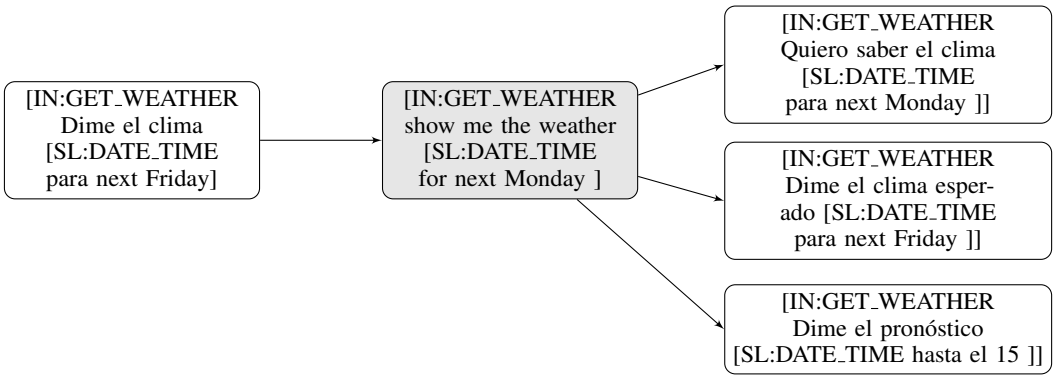

Figure 3: Match and Filter data augmentation: 1- For each CS utterance (target), find the the closest EN neighbor (source). 2- Learn a generative model from source to target 3- Perform beam search to generate more targets from the source utterances.

ing autoencoder trained on massive amount of web data, as the generative model. Our goal is to generate diverse Spanglish data from the EN data. Even though BART was trained for English, we found it very effective for this task. We hypothesize this is due to the abundance of the Spanish text among EN web data and the proximity of the word-piece tokens among them. We also experimented with multilingual BART (Liu et al., 2020a) but found it very challenging to fine-tune for this task.

First, we convert the data to a bracket format (Vinyals et al., 2015), which is called the seqlogical form in Gupta et al. (2018). Examples of this format are shown in Fig. 3. In the seqlogical form, we include the intent (i.e., sentence label) at the beginning and for each slot, we first include the label and text in brackets.

We perform our data augmentation technique in the following steps:

1. Find the top K closest EN neighbors to every CS query in the CSTOP_{100}. We enforce the neighbors to have the same parse as the CS utterance, i.e., same intent and same slot labels, and use the Levenshtein distance to rank the EN sequences.

2. Having this parallel corpus, i.e., top-K EN neighbors as the source and the original CS query as the target, Fine-tune the BART model. We use K=10 in our experiments to increase the parallel data size to around 650.

3. During the inference, Use the beam size of 5 to decode CS utterances from the same EN source data. Since both the source and target sequences are in the seqlogical form, the CS generated sequences are already annotated.

In Fig. 3, we have shown the closest EN neighbor corresponding to the original CS example in Fig. 1. The CS utterance can be seen as a rough translation of the EN sentence. We have also shown the top three generated CS utterances from the EN example.

In order to reduces the noise, we filter the generated sequences that either already exist in CSTOP_{100}, are not valid trees, or have a semantic parse different from the original utterance. We augment CSTOP_{100} with the data, and fine-tune the XLM-R baseline.

In the second column of Table 6, we have shown the average data augmentation improvement over the three CSTOP_{100} samples for the few-shot setting. We can see that even after fine-tuning on the EN monolingual data (the second row), the augmentation technique improves this strong baseline. In the last row, we first use the translation alignment of the previous section to obtain ES*. After fine-tuning on this set combined with the EN data,

we further fine-tune on the CSTOP_{100}. We can see that the best model enjoys improvements from both zero-shot (translation alignment) and the few-shot (generate and filter) augmentation techniques. We also note that the p-value corresponding to the second and third row gains are 0.018 and 0.055, respectively.

7 Related Work

7.1 XL Pre-training

Most of the initial work on pre-trained XL representations was focused on embedding alignment (Xing et al., 2015; Zhang et al., 2017; Conneau et al., 2018). Recent developments in this area have focused on the context-aware XL alignment of contextual representations (Schuster et al., 2019; Aldarmaki and Diab, 2019; Wang et al., 2019; Cao et al., 2020). Recently, pre-trained multilingual language models such as mBERT (Devlin et al., 2019), XLM (Conneau and Lample, 2019), and Conneau et al. (2019) have been introduced, and Pires et al. (2019) demonstrate the effectiveness of these on sequence labeling tasks.
Separately, Liu et al. (2020a) introduce mBART, a sequence-to-sequence denoising auto-encoder pre-trained on monolingual corpora in many languages using a denoising autoencoder objective (Lewis et al., 2019).

7.2 Code-Switching

Following the ACL shared tasks, CS is mostly discussed in the context of word-level language identification (Molina et al., 2016) and NER (Aguilar et al., 2018). Techniques such as curriculum learning (Choudhury et al., 2017) and attention over different embeddings (Wang et al., 2018; Winata et al., 2019a) have been among the successful techniques. CS parsing and use of monolingual parses are discussed in Sharma et al. (2016); Bhat et al. (2017, 2018). Sharma et al. (2016) introduces a Hinglish test set for a shallow parsing pipeline. In Bhat et al. (2017), outputs of two monolingual dependency parsers are combined to achieve a CS parse. Bhat et al. (2018) extends this test set by including training data and transfers the knowledge from monolingual treebanks. Duong et al. (2017) introduced a CS test set for semantic parsing which is curated by combining utterances from the two monolingual datasets. In contrast, CSTOP is procured independently of the monolingual data and exhibits much more linguistic diversity. In Pratapa et al. (2018), linguistic rules are used to generate CS data which has been shown to be effective in reducing the perplexity of a CS language model. In contrast, our augmentation techniques are generic and do not require rules or constituency parsers.

7.3 XL Data Augmentation

Most approaches to cross-lingual data augmentation use machine translation and slot projection for sequence labeling tasks (Jain et al., 2019). Wei and Zou (2019) uses simple operations such as synonym replacement and Lee et al. (2019a) use phrase replacement from a parallel corpus to augment the training data. Singh et al. (2019) present XLDA that augments data by replacing segments of input text with its translations in other languages. Some recent approaches (Chang et al., 2019; Winata et al., 2019b) also train generative models to artificially generate CS data. More recently, Kumar et al. (2020) study data augmentation using pre-trained transformer models by incorporating label information during fine-tuning. Concurrent to our work, Bari et al. (2020) introduce Multimix, where data augmentation from pre-trained multilingual language models and self-learning are used for semi-supervised learning. Recently, Liu et al. (2019) generate CS data by translating keywords picked based on attention scores from a monolingual model. Generating CS data has recently been studied in Liu et al. (2020b)

7.4 Task-oriented Dialog

The intent/slot framework is the most common way of performing language understanding for task oriented dialog using. Bidirectional LSTM for the sentence representation alongside separate projection layers for intent and slot tagging is the typical architecture for the joint task (Yao et al., 2013; Mesnil et al., 2015; Hakkani-Tur et al., 2016). Such representations can accommodate trees of up to length two, as is the case in CSTOP. More recently, an extension of this framework has been introduced to fit the deeper trees (Gupta et al., 2018; Rongali et al., 2020).

8 Conclusion

In this paper, we propose a new task for code-switched semantic parsing and release a dataset, CSTOP, containing 5800 Spanglish utterances over

two domains. We hope this foments further research on the code-switching phenomenon which has been set back by paucity of sizeable curated datasets. We show that cross-lingual pre-trained models can generalize better than traditional models to the code-switched setting when monolingual data from only one languages is available. In the presence of only EN data, we introduce generic augmentation techniques based on translation and generation. As such, we show that translating and aligning the EN data can significantly improve the zero-shot performance. Moreover, generating code-switched data using a generation model and a match-and-filter approach leads to improvements in a few-shot setting. We leave exploring and combining other augmentation techniques to future work.

References

Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Mona Diab, Julia Hirschberg, and Thamar Solorio. 2018. Named entity recognition on code-switched data: Overview of the CALCS 2018 shared task. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 138–147, Melbourne, Australia. Association for Computational Linguistics.

Hanan Aldarmaki and Mona Diab. 2019. Context-aware cross-lingual mapping. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3906–3911, Minneapolis, Minnesota. Association for Computational Linguistics.

Ahmed Aly, Kushal Lakhotia, Shicong Zhao, Mrinal Mohit, Barlas Oguz, Abhinav Arora, Sonal Gupta, Christopher Dewan, Stef Nelson-Lindall, and Rushin Shah. 2018. Pytext: A seamless path from NLP research to production. *CoRR*, abs/1812.08729.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

M. Saiful Bari, Muhammad Tasnim Mohiuddin, and Shafiq R. Joty. 2020. Multimix: A robust data augmentation strategy for cross-lingual NLP. *CoRR*, abs/2004.13240.

Irshad Bhat, Riyaz A. Bhat, Manish Shrivastava, and Dipti Sharma. 2017. Joining hands: Exploiting monolingual treebanks for parsing of code-mixing data. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 324–330, Valencia, Spain. Association for Computational Linguistics.

Irshad Bhat, Riyaz A. Bhat, Manish Shrivastava, and Dipti Sharma. 2018. Universal dependency parsing for Hindi-English code-switching. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 987–998, New Orleans, Louisiana. Association for Computational Linguistics.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Steven Cao, Nikita Kitaev, and Dan Klein. 2020. Multilingual alignment of contextual word representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Ching-Ting Chang, Shun-Po Chuang, and Hung-yi Lee. 2019. Code-switching sentence generation by generative adversarial networks and its application to data augmentation. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 554–558. ISCA.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *ArXiv*, abs/1902.10909.

Monojit Choudhury, Kalika Bali, Sunayana Sitaram, and Ashutosh Baheti. 2017. Curriculum design for code-switching: Experiments with language identification and language modeling with deep neural networks. In *Proceedings of the 14th International Conference on Natural Language Processing (ICON-2017)*, pages 65–74, Kolkata, India. NLP Association of India.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In H. Wallach, H. Larochelle, A. Beygelzimer, F. dAlché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 7059–7069. Curran Associates, Inc.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.

Alexis Conneau, Guillaume Lample, Marc'Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2018. Word translation without parallel data. *ArXiv*, abs/1710.04087.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Long Duong, Hadi Afshar, Dominique Estival, Glen Pink, Philip Cohen, and Mark Johnson. 2017. Multilingual semantic parsing and code-switching. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 379–389, Vancouver, Canada. Association for Computational Linguistics.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648, Atlanta, Georgia. Association for Computational Linguistics.

Anna Maria Escobar and Kim Potowski. 2015. *El Español de los Estados Unidos*.

Sonal Gupta, Rushin Shah, Mrinal Mohit, Anuj Kumar, and Mike Lewis. 2018. Semantic parsing for task oriented dialog using hierarchical representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Dilek Hakkani-Tur, Gokhan Tur, Asli Celikyilmaz, Yun-Nung Chen, Jianfeng Gao, Li Deng, and Ye-Yi Wang. 2016. Multi-domain joint semantic frame parsing using bi-directional RNN-LSTM. In *Interspeech 2016*, pages 715–719.

Alankar Jain, Bhargavi Paranjape, and Zachary C. Lipton. 2019. Entity projection via machine translation for cross-lingual NER. 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)*, pages 1083–1092, Hong Kong, China. Association for Computational Linguistics.

Aravind K. Joshi. 1982. Processing of sentences with intra-sentential code-switching. In *Coling 1982: Proceedings of the Ninth International Conference on Computational Linguistics*.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. Cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

Philipp Koehn, Franz J. Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. *CoRR*, abs/2003.02245.

Grandee Lee, Xianghu Yue, and Haizhou Li. 2019a. Linguistically motivated parallel data augmentation for code-switch language modeling. In *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*, pages 3730–3734. ISCA.

Kyungjae Lee, Sunghyun Park, Hojae Han, Jinyoung Yeo, Seung-won Hwang, and Juho Lee. 2019b. Learning with limited data for multilingual reading comprehension. 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)*, pages 2840–2850, Hong Kong, China. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020a. Multilingual denoising pre-training for neural machine translation. *CoRR*, abs/2001.08210.

Zihan Liu, Genta Indra Winata, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2019. Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems.

Zihan Liu, Genta Indra Winata, Zhaojiang Lin, Peng Xu, and Pascale Fung. 2020b. Attention-informed mixed-language training for zero-shot cross-lingual task-oriented dialogue systems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8433–8440.

Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Y Bengio, li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, and Geoffrey Zweig. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 23:530–539.

Giovanni Molina, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, Nicolas Rey-Villamizar, Mona Diab, and Thamar Solorio. 2016. Overview for the second shared task on language identification in code-switched data. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*, pages 40–49, Austin, Texas. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? In *ACL (1)*, pages 4996–5001. Association for Computational Linguistics.

Shana Poplack. 2004. *Code-Switching*, pages 589–596.

Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018. Language modeling for code-mixing: The role of linguistic theory based synthetic data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1543–1553, Melbourne, Australia. Association for Computational Linguistics.

Subendhu Rongali, Luca Soldaini, Emilio Monti, and Wael Hamza. 2020. Don't parse, generate! a sequence to sequence architecture for task-oriented semantic parsing.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2018. Cross-lingual transfer learning for multilingual task oriented dialog. In *2nd workshop on Conversational AI NeurIPS*.

Tal Schuster, Ori Ram, Regina Barzilay, and Amir Globerson. 2019. Cross-lingual alignment of contextual word embeddings, with applications to zero-shot dependency parsing. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1599–1613, Minneapolis, Minnesota. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Arnav Sharma, Sakshi Gupta, Raveesh Motlani, Piyush Bansal, Manish Shrivastava, Radhika Mamidi, and Dipti M. Sharma. 2016. Shallow parsing pipeline - Hindi-English code-mixed social media text. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1340–1345, San Diego, California. Association for Computational Linguistics.

Jasdeep Singh, Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2019. XLDA: cross-lingual data augmentation for natural language inference and question answering. *CoRR*, abs/1905.11471.

Jörg Tiedemann, Željko Agić, and Joakim Nivre. 2014. Treebank translation for cross-lingual parser induction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, pages 130–140, Ann Arbor, Michigan. Association for Computational Linguistics.

Oriol Vinyals, Ł ukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. 2015. Grammar as a foreign language. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2773–2781. Curran Associates, Inc.

Changhan Wang, Kyunghyun Cho, and Douwe Kiela. 2018. Code-switched named entity recognition with embedding attention. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 154–158, Melbourne, Australia. Association for Computational Linguistics.

Yuxuan Wang, Wanxiang Che, Jiang Guo, Yijia Liu, and Ting Liu. 2019. Cross-lingual BERT transformation for zero-shot dependency parsing. 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)*, pages 5721–5727, Hong Kong, China. Association for Computational Linguistics.

Jason Wei and Kai Zou. 2019. EDA: Easy data augmentation techniques for boosting performance on text classification tasks. 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)*, pages 6382–6388, Hong Kong, China. Association for Computational Linguistics.

Genta Indra Winata, Zhaojiang Lin, Jamin Shin, Zihan Liu, and Pascale Fung. 2019a. Hierarchical meta-embeddings for code-switching named entity recognition. 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)*, pages 3541–3547, Hong Kong, China. Association for Computational Linguistics.

Genta Indra Winata, Andrea Madotto, Chien-Sheng Wu, and Pascale Fung. 2019b. Code-switched language models using neural based synthetic data from parallel sentences. In *Proceedings of the 23rd Conference on Computational Natural Language Learning, CoNLL 2019, Hong Kong, China, November 3-4, 2019*, pages 271–280. Association for Computational Linguistics.

Chao Xing, Dong Wang, Chao Liu, and Yiye Lin. 2015. Normalized word embedding and orthogonal transform for bilingual word translation. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1006–1011, Denver, Colorado. Association for Computational Linguistics.

Kaisheng Yao, Geoffrey Zweig, Mei-Yuh Hwang, Yangyang Shi, and Dong Yu. 2013. Recurrent neural networks for language understanding. In *Interspeech*, pages 2524–2528.

Meng Zhang, Yang Liu, Huanbo Luan, and Maosong Sun. 2017. Adversarial training for unsupervised bilingual lexicon induction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1959–1970, Vancouver, Canada. Association for Computational Linguistics.

A Appendix

Here, we describe the details regarding the training as the validation results.

A.1 Model and Training Parameters

In Table 7, we have shown the training details for all our models. We use ADAM (Kingma and Ba, 2014) with Learning Rate (LR), Weight Decay (WD), and Batch Size (BSz) that is listed for each model. We have also shown the number of epochs and the average training time for the full CS data using 8 V100 Nvidia GPUs. For all our XLM-R experiments, we use the XLM-R large from the PyText[2] (Aly et al., 2018) which is pre-trained on 100 languages. For the XLM experiments, we use XLM-20 pre-trained over 20 languages and use the same fine-tuning parameters as XLM-R but run for more epochs.

For the LSTM models, we use a two-layer LSTM with hidden dimension of 256 and dropout of 0.3 for all connections. We use one layer of MLP of dimension 200 for both the slot tagging and the intent classification. We also use an ensemble of five models for all the LSTM experiments to reduce the variance. The LSTM model with SentencePiece embeddings in Table 4 were trained with embedding dimension of 1024 similar to the XLM-R model.

A.2 Validation Results

In Table. 9, we have shown the validation results when using the full CS training data. We have not shown the corresponding results for the zero-shot experiments as no validation data was not used and the monolingual models were tested off the shelf.

In Table. 8, we have shown the validation results for the few-shot setting.

[2] https://pytext.readthedocs.io/en/master/xlm_r.html

Model	BSz	LR	WD	Epoch	Avg Time
XLM-R (pronoun)	8	0.000005	0.0001	15	5 hr
XLM (pronoun)	8	0.000005	0.0001	20	1 hr
LSTM (pronoun+question)	64	0.03	0.00001	45	45 min

Table 7: Training Parameters

Model/Training Data	**Few shot**	**Few shot + Generate and Filter Augmentation**
XLM-R	61.7	70.4
XLM-R fine-tuned on EN	83.3	83.9
XLM-R fine-tuned on EN+ES*	83.5	84.9

Table 8: Validation Accuracy when only a few CS instances (FS) are available during training. FS+G refers to augmenting the few-shot instances with generated CS data. ES* is the auto-translated and aligned data.

Lang/Model	**MUSE**	**XLM**	**XLM-R**
CS	87.8	90.7	95.0
CS + EN	89.0	92.9	95.5

Table 9: Validation results for the Full-training on the CS data