

GLUECoS: An Evaluation Benchmark for Code-Switched NLP

Simran Khanuja¹ Sandipan Dandapat² Anirudh Srinivasan¹
Sunayana Sitaram¹ Monojit Choudhury¹

¹ Microsoft Research, Bangalore, India

² Microsoft R&D, Hyderabad, India

{t-sikha, sadandap, t-ansrin, sunayana.sitaram, monojitc}@microsoft.com

Abstract

Code-switching is the use of more than one language in the same conversation or utterance. Recently, multilingual contextual embedding models trained on multiple monolingual corpora have shown promising results on various cross-lingual and multilingual tasks. We present an evaluation benchmark GLUECoS for code-switched languages that spans several NLP tasks in English-Hindi and English-Spanish. Specifically, our evaluation benchmark includes Language Identification from text, POS tagging, Named Entity Recognition, Sentiment Analysis, Question Answering and a new task for code-switching, Natural Language Inference. We present results on all these tasks using cross-lingual word embedding models and multilingual models. In addition, we fine-tune multilingual models on artificially generated code-switched data. Although multilingual models perform significantly better than cross-lingual models, our results show that in most tasks, across both language pairs, multilingual models fine-tuned on code-switched data perform best, showing that multilingual models can be further optimized for code-switching tasks.

1 Introduction

Code-switching, or code-mixing, is the use of more than one language in the same utterance or conversation and is prevalent in multilingual societies all over the world. It is a spoken phenomenon and is found most often in informal chat and social media on the Internet. Processing, understanding, and generating code-mixed text and speech has become an important area of research.

Recently, contextual word embedding models trained on a large amount of text data have shown state-of-the-art results in a variety of NLP tasks. Models such as BERT (Devlin et al., 2018) and its multilingual version, mBERT, rely on large

amounts of unlabeled monolingual text data to build monolingual and multilingual models that can be used for downstream tasks. (Wang et al., 2018) propose a Generalized Language Evaluation Benchmark (GLUE) to evaluate embedding models on a wide variety of tasks, with the motivation being that a model that does well across different tasks can be assumed to have truly learned something about the language.

Data and annotated resources are scarce for code-switched languages, even if one or both languages being mixed are high resource. Due to this, there is a lack of standardized datasets in code-switched languages other than those used in shared tasks in a few language pairs. Although models using synthetic code-switched data and cross-lingual embedding techniques have been proposed for code-switching (Pratapa et al., 2018a), there has not been a comprehensive evaluation of embedding models across different types of tasks. Furthermore, there have been claims that multilingual models such as mBERT are competent in zero-shot cross lingual transfer and in code-switched settings as well. Though comprehensively validated by (Pires et al., 2019) in the case of zero-shot transfer, the probing in code-switched settings was limited to one dataset of one task, namely POS Tagging.

To address all these issues and inspired by the GLUE (Wang et al., 2018) benchmark, we propose GLUECoS, a language understanding evaluation framework for Code-Switched NLP. We include five tasks from previously conducted evaluations and shared tasks, and propose a sixth, *Natural Language Inference* task for code-switching, using a new dataset¹. We include tasks varying in complexity ranging from word-level tasks [*Language Identification (LID)*; *Named Entity Recognition (NER)*], syntactic tasks [*POS tagging*], semantic tasks [*Sen-*

¹complete details about this dataset are in a companion paper, under review

iment Analysis; Question Answering] and finally a *Natural Language Inference* task. Where available, we include multiple datasets for each task in English-Spanish and English-Hindi. We choose these language pairs, not only due to the relative abundance of publicly available datasets, but also because they represent variations in types of code-switching, language families, and scripts between the languages being mixed. We test various cross-lingual and multilingual models on all these tasks. In addition, we also test models trained with synthetic code-switched data. Lastly, we fine-tune the best performing multilingual model with synthetic code-switched data and show that in most cases, its performance exceeds the multilingual model, showing that multilingual models can be further optimized for code-switched settings.

The main contributions of our work are as follows:

- We point out the lack of standardized datasets for code-switching and propose an evaluation benchmark GLUECoS, which can be used to test models on various NLP tasks in English-Hindi and English-Spanish.
- In creating the benchmark, we highlight the tasks that are missing from code-switched NLP and propose a new task, *Natural Language Inference*, for code-switched data.
- We evaluate cross-lingual and pre-trained multilingual embeddings on all these tasks, and observe that pre-trained multilingual embeddings significantly outperform cross-lingual embeddings. This highlights the competence of generalized language models over cross lingual word embeddings.
- We fine-tune pre-trained multilingual models on linguistically motivated synthetic code-switched data, and observe that they perform marginally better, highlighting that these models can be further optimized for code-switched settings.

The rest of the paper is organized as follows. Section 2 relates our work to prior work. Section 3 introduces the tasks and datasets we use for GLUECoS. Section 4 describes the experimental setup with details of the models used for baseline evaluations. Section 5 presents the results and analysis. Section 6 provides a direction for future work and concludes.

2 Relation to prior work

The idea of a generalized benchmark for code-switching is inspired by GLUE (Wang et al., 2018), which has spurred research in *Natural Language Understanding* in English, to an extent that a set of harder tasks have been curated in a follow-up benchmark, SuperGLUE (Wang et al., 2019) once models beat the human baseline for GLUE. The motivation behind GLUE is to evaluate models in a multi-task learning framework across several tasks, so that tasks with less training data can benefit from others. Although our current work does not include models evaluated in a multi-task setting, we plan to implement this in subsequent versions of the benchmark.

There have been shared tasks conducted in the past as part of code-switching workshops co-located with notable NLP conferences. The first and second workshops on *Computational Approaches to Code Switching* (Diab et al., 2014, 2016) conducted a shared task on *Language Identification* for several language pairs (Solorio et al., 2014; Molina et al., 2016). The third workshop (Aguilar et al., 2018) included a shared task on *Named Entity Recognition* for the English-Spanish and Modern Standard Arabic-Egyptian Arabic language pairs (Aguilar et al., 2019).

The *Forum for Information Retrieval Evaluation* (FIRE) aims to meet new challenges in multilingual information access and has conducted several shared tasks on code-switching. These include tasks on *transliterated search*, (Roy et al., 2013; Choudhury et al., 2014) *code-mixed entity extraction* (Rao and Devi, 2016) and *mixed script information retrieval* (Sequiera et al., 2015; Banerjee et al., 2016). Other notable shared tasks include the *Tool Contest on POS Tagging for Code-Mixed Indian Social Media* at ICON 2016 (Jamatia et al., 2016), *Sentiment Analysis for Indian Languages (Code-Mixed)* at ICON 2017 (Patra et al., 2018) and the *Code-Mixed Question Answering Challenge* (Chandu et al., 2018a).

Each of the shared tasks mentioned above attracted several participants and have led to follow up research in these problems. However, all tasks have focused on a single NLP problem and so far, there has not been an evaluation of models across several code-switched NLP tasks. Our objective with proposing GLUECoS is to address this gap, and determine which models best generalize across different tasks, languages and datasets.

3 Tasks and Datasets

Some NLP tasks are inherently more complex than others - for example, a *Question Answering* task that needs to understand both the meaning of the question and answer, is harder to solve by a machine than a word-level *Language Identification* task, in which a dictionary lookup can give reasonable results. Some datasets and domains may contain very little code-switching, while others may contain more frequent and complex code-switching. Similar languages, when code-switched, may maintain the word order of both languages, while other language pairs that are very different may take on the word order of one of the languages. With these in mind, our choice of tasks and datasets for GLUE-CoS are based on the following principles :

- We choose a variety of tasks, ranging from simpler ones, on which the research community has already achieved high accuracies, to relatively more complex, on which very few attempts have been made.
- We desire to evaluate models on language-pairs from different language families, and on a varied number of tasks, to enable detailed analysis and comparison. This led us to choose *English-Hindi* and *English-Spanish*, as we found researched upon datasets for almost all tasks in our benchmark for these language pairs.
- English and Spanish are written in the Roman script, while *English-Hindi* datasets can contain Hindi words written either in the original Devanagari script, or in the Roman script.
- We include multiple datasets from each language pair where available, so that results can be compared across datasets for the same task.

Due to the lack of standardized datasets, we choose to create our own train-test-validation splits for some tasks, details of which can be found in Appendix A. Table 1 shows all the datasets that we use, with their statistics, while Table 2 shows the code-switching statistics of the data in terms of standardized metrics for code-switching (Gambäck and Das, 2014; Guzmán et al., 2017). Briefly, the code-mixing metrics include :

- *Code-Mixing Index* (CMI) : The fraction of language dependent tokens not belonging to the matrix language in the utterance.

- *Average switch-points* (SP Avg) : The average number of intra-sentential language switch-points in the corpus.
- *Multilingual Index* (M-index) : A word-count-based measure quantifying the inequality of distribution of language tags in a corpus of at least two languages.
- *Probability of Switching* (I-index) : The proportion of the number of switchpoints in the corpus, relative to the number of language-dependent tokens.
- *Burstiness* : Quantifies whether switching occurs in bursts (randomly similar to a Poisson process), or it has a more periodic character.
- *Language Entropy* (LE) : The bits of information needed to describe the distribution of language tags.
- *Span Entropy* (SE) : The bits of information needed to describe the distribution of language spans.

3.1 Language Identification (LID)

Language Identification is the task of obtaining word-level language labels for code-switched sentences. For *English-Hindi* we choose the FIRE 2013 (FIRE LID) dataset originally created for the transliterated search subtask (Roy et al., 2013). The test and development sets provided contain word-level language tagged sentences. Romanized Hindi words are provided with manually created Devanagari transliterations. For training we use a POS tagging dataset (Jamatia et al., 2016) which also contains language labels.

For *English-Spanish* we choose the dataset in (Solorio et al., 2014), provided as part of the LID shared task at EMNLP 2014. The data consists of code-switched sentences language tagged at the word level.

3.2 Part of Speech (POS) tagging

POS tagging includes labelling at the word level, grammatical part of speech tags such as noun, verb, adjective, pronoun, prepositions etc. For *English-Hindi*, we use two datasets. The first is the code-switched Universal Dependency parsing dataset provided by (Bhat et al., 2018) (UD POS). This corpus contains a *transliterated* version, where Hindi is in the Roman script, and also a *corrected* version

English-Hindi				
Corpus	Sent (Train)	Sent (Dev)	Sent (Test)	Sent (All)
Fire LID (D)	2631	500	406	3537
UD POS (D)	1384	215	215	1814
FG POS (R)	2104	263	264	2631
IIITH NER (R)	2467	308	309	3084
SAIL Sentiment (R)	10080	1260	1261	12601
QA (R)	250	-	63	313
NLI (R)	1040	130	130	1300
English-Spanish				
Corpus	Sent (Train)	Sent (Dev)	Sent (Test)	Sent (All)
EMNLP 2014	10259	1140	3014	14413
Bangor POS	2192	274	274	2758
CALCS NER	27366	3420	3421	34208
Sentiment	1681	211	211	2103

Table 1: Corpus Statistics. (R) and (D) indicates Hindi written in Roman and Devanagari script, respectively

English-Hindi							
Corpus	CMI	SP Avg	M-index	I-index	Burstiness	LE	SE
Fire LID	78.26	4.47	0.39	0.33	-0.42	0.86	1.02
UD POS	136	4.98	0.46	0.39	-0.25	1.35	1.47
FG POS	68	5.5	0.4	0.34	-0.43	0.87	1.05
IIITH NER	133	11.39	0.64	0.53	-0.26	1.28	1.36
SAIL Sentiment	72.8	5.07	0.02	0.32	-0.32	0.87	1.17
QA	142.28	3.96	0.81	0.5	-0.4	0.89	1.09
NLI	149.95	66.74	0.44	0.63	-0.2	1.53	1.39
English-Spanish							
Corpus	CMI	SP Avg	M-index	I-index	Burstiness	LE	SE
EMNLP 2014	33.46	2.86	0.33	0.29	-0.34	0.79	1.1
Bangor POS	123.06	1.67	0.32	0.27	-0.35	0.82	1.06
CALCS NER	94.52	3.17	0.004	0.31	-0.42	0.75	1.02
Sentiment	110.56	4.13	0.15	0.27	-0.21	0.79	1.42

Table 2: Code-switching Statistics

in which Hindi has been manually converted back to Devanagari by annotators.

The second *English-Hindi* dataset we use was part of the *ICON 2016 Tool Contest on POS Tagging for Code-Mixed Indian Social Media Text* (Jamatia et al., 2016) (FG POS). Since Hindi is originally in the Roman script, we use an off-the-shelf transliterator to convert Hindi words to Devanagari.

For *English-Spanish*, of the two corpora utilised in (AlGhamdi et al., 2016), we choose the Bangor Miami corpus (Bangor POS) owing to the larger size of the corpus. This corpus consists of conversations of Spanish speakers in Florida who are also fluent in English.

3.3 Named Entity Recognition (NER)

NER involves recognizing named entities such as *person, location, organization etc.* in a segment of text. For *English-Hindi* we use the Twitter NER corpus provided by (Singh et al., 2018) (IIITH NER). This corpus is originally in Roman script and is not language tagged. We use the LID tool provided by (Rijhwani et al., 2017) to obtain language tags and an off-the-shelf transliterator to transliterate it to Devanagari.

For *English-Spanish*, we use the Twitter NER corpus provided as part of the CALCS 2018 shared task on NER for code-switched data (Aguilar et al., 2019) (CALCS NER). This corpus is not language tagged, hence to evaluate the code-switching met-

rics of this corpus, we obtain word level language tags using an off-the-shelf language detector.

3.4 Sentiment Analysis

Sentiment analysis is a sentence classification task wherein each sentence is labeled to be expressing a *positive*, *negative* or *neutral* sentiment.

For *English-Hindi* we choose the sentiment annotated social media corpus used in the *ICON 2017 shared task; Sentiment Analysis for Indian Languages (SAIL)* (Patra et al., 2018). This corpus is originally language tagged at the word level with Hindi in the Roman script. We obtain the transliterated Devanagari version using an off-the-shelf transliterator.

For *English-Spanish* we choose the sentiment annotated Twitter dataset provided by (Vilares et al., 2016) which we split into an 8:1:1 train:test:validation split ensuring sentiment distribution.

3.5 Question Answering (QA)

Question Answering is the task of answering a question based on the given context or world knowledge. We choose the dataset provided by (Chandu et al., 2018a) which was part of a shared task on code-mixed question answering for *English-Hindi*. It contains two types of questions, one with context (185 article based questions) and one containing image based questions (774 questions). For the image based questions we use the *DrQA - Document Retriever module*² to extract the most relevant context from Wikipedia. Since it is a code-switched dataset, context could not be extracted for all questions. We obtain a final dataset having 313 (question-answer-context) triples.

3.6 Natural Language Inference (NLI)

Natural Language Inference is the task of inferring a positive (entailed) or negative (contradicted) relationship between a premise and hypothesis. While most NLI datasets contain sentences or images as premises, the code-switched NLI dataset we use contains conversations as premises, making it a conversational NLI task. This dataset contains scripted movie conversations in code-switched *English-Hindi*, in the Roman script. Annotators create *True* and *False* code-mixed hypotheses for each conversational context, further validated by additional annotators, to obtain a corpus of premises

²<https://github.com/facebookresearch/DrQA>

and hypotheses. Complete details about this corpus are available in a companion paper, under review. Given a conversation context and hypothesis, the NLI task is to predict whether the hypothesis is entailed in, or contradicts the conversational premise.

4 Experimental Setup

We use standard architectures for solving each of the tasks mentioned above (Refer to Appendix B). We experiment with several existing cross lingual word embeddings that have been shown to perform well on cross lingual tasks. We also experiment with the Multilingual BERT (mBERT) model released by (Devlin et al., 2018). In a survey on cross lingual word embeddings, (Ruder et al., 2017) establish that various embedding methods optimize for similar objectives given that the supervision data involved in training them is similar. Based on this, we choose the following representative embedding methods that vary in the amount of supervision involved in training them.

4.1 MUSE Embeddings

We use the MUSE library³ to train both supervised and unsupervised word embeddings. The unsupervised word embeddings are learnt without any parallel data or anchor point. It learns a mapping from the source to the target space using adversarial training and (iterative) Procrustes refinement (Conneau et al., 2017). The supervised method leverages a bilingual dictionary (or identical character strings as anchor points), to learn a mapping from the source to the target space using (iterative) Procrustes alignment.

4.2 BiCVM Embeddings

This method, proposed by (Hermann and Blunsom, 2014), leverages parallel data, based on the assumption that parallel sentences are equivalent in meaning and subsequently have similar sentence representations. We use the BiCVM toolkit⁴ to learn these embeddings. The parallel corpus we use for *English-Spanish* consists of 4.5M parallel sentences from Twitter. For *English-Hindi*, we make use of an internal parallel corpus consisting of roughly 5M parallel sentences.

³<https://github.com/facebookresearch/MUSE>

⁴<https://github.com/karlmoritz/bicvm/>

4.3 BiSkip Embeddings

This method makes use of parallel corpora as well as word alignments to learn cross-lingual embeddings. (Luong et al., 2015) adapt the skip-gram objective originally proposed by (Mikolov et al., 2013) to a bilingual setting wherein a model learns to predict words cross-lingually along with the monolingual objectives. We make use of the *fastalign toolkit*⁵ to learn word alignments given parallel corpora and use the *BiVec toolkit*⁶ to learn the final BiSkip embeddings given the parallel corpora and the word alignments. The parallel corpora utilised to learn these are the same as those used to learn the BiCVM embeddings.

4.4 Synthetic Data (GCM) Embeddings

We also experiment with skip-gram embeddings learnt from synthetically generated code-mixed data as proposed by (Pratapa et al., 2018b). We make use of the *fasttext library*⁷ to learn the skip-gram embeddings. For *English-Spanish*, we obtain data from (Pratapa et al., 2018a) which consists of 8M synthetic code-switched sentences. For *English-Hindi*, we generate synthetic data from the IITB parallel corpus.⁸ We sample from the generated sentences obtained using Switch Point Fraction (SPF), as described in (Pratapa et al., 2018a), to obtain a GCM corpus of roughly 10M sentences.

4.5 mBERT

Multilingual BERT is pre-trained on monolingual corpora of 104 languages and has been shown to perform well on zero shot cross-lingual model transfer and code-switched POS tagging (Pires et al., 2019). Specifically, we use the *bert-base-multilingual-cased* model for our experiments.

4.6 Modified mBERT

(Sun et al., 2019) show that fine-tuning BERT with in-domain data on language modeling improves performance on downstream tasks. On similar lines, we fine-tune the mBERT model with synthetically generated code-switched data (gCM) and a small amount of real code-switched data (rCM), on the masked language modeling objective. The training curriculum we use in fine-tuning this model is similar to as proposed by (Pratapa et al., 2018a),

which has been shown to improve language modeling perplexity. Although we train on real code-mixed data, it accounts for a small fraction (less than 5%) of the total code-mixed data used. Refer to Appendix C for training details.

5 Results and Analysis

Tables 3-8 show the results of using the embedding techniques described above for each task and dataset. mBERT provides a large increase in accuracy as compared to cross-lingual techniques, and in most cases, the modified mBERT technique performs best. We do not experiment with baseline or cross-lingual embedding techniques for NLI, since we find that mBERT surpasses the other techniques by a large margin for all other tasks. For NLI, as in the other cases, we find that modified mBERT performs better than mBERT. We hypothesize that this happens because code-switched languages are not just a union of two monolingual languages. The distributions and usage of words in code-switched languages differ from their monolingual counterparts, and can only be captured with real code-switched data, or synthetically generated data that closely mimics real data.

(Glavas et al., 2019) point out how all cross-lingual word embedding methods optimize for bilingual lexicon induction. Each model is trained using different language pairs and different training and evaluation dictionaries, leading to it overfitting to the task it is optimizing for and failing in other cross-lingual scenarios. Also, the loss function in training cross-lingual word embeddings has a component where w_1 in one language predicts the context of its aligned word w_2 in the other language. However, in the case of code-switching, w_1 appearing in the context of w_2 may not be natural. This clearly highlights the need to learn cross-lingual embeddings keeping code-mixed language processing as an optimization objective.

The results using mBERT cannot be directly compared to the cross-lingual models because of the difference in the magnitude of data involved in training. Also, due to the fact that mBERT is trained on 104 languages together, with massive amounts of data for a large number of epochs, it learns several common features better providing for a well represented common embedding space. The training data used for training the cross-lingual embeddings is restricted to Twitter and query logs, while mBERT is trained on the entire wiki dump.

⁵https://github.com/clab/fast_align

⁶<https://github.com/lmthang/bivec>

⁷<https://fasttext.cc/>

⁸http://www.cfil.t.iitb.ac.in/iitb_parallel/

Data	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
FIRE En-Hi	93.21	94.53	94.92	93.98
	BiCVM	GCM	mBERT	Mod. mBERT
	95.24	93.64	95.87	96.6
EMNLP2014 En-Es	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
	92.95	92.86	93.39	92.79
	BiCVM	GCM	mBERT	Mod. mBERT
	91.47	92.42	95.97	96.24

Table 3: LID results (F1)

Data	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
UD En-Hi	77.49	78.06	77.88	77.43
	BiCVM	GCM	mBERT	Mod. mBERT
	77.49	77.84	87.16	88.06
FG En-Hi	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
	60.88	60.76	60.59	60.4
	BiCVM	GCM	mBERT	Mod. mBERT
Bangor En-Es	60.2	61.03	63.42	63.31
	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
	88.775	88.65	88.82	89.2
Bangor En-Es	BiCVM	GCM	mBERT	Mod. mBERT
	87.46	89.37	93.33	93.62

Table 4: POS results (F1)

Data	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
IIITH En-Hi	71.52	71.48	72.148	72.13
	BiCVM	GCM	mBERT	Mod. mBERT
	71.55	72.37	74.96	78.21
CALCS En-Es	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
	47.9	53.74	54.17	52.98
	BiCVM	GCM	mBERT	Mod. mBERT
	51.6	53.57	59.69	61.77

Table 5: NER results (F1)

Data	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
SAIL En-Hi	50.44	48.37	51.27	48.84
	BiCVM	GCM	mBERT	Mod. mBERT
	49.56	50.01	58.24	59.35
Sentiment En-Es	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
	50.62	58.73	58.44	60.4
	BiCVM	GCM	mBERT	Mod. mBERT
	62.62	62.89	66.03	69.31

Table 6: Sentiment Analysis results (F1)

Data	Baseline	Unsup. MUSE	Sup. MUSE	BiSkip
QA En-Hi	61.39	56.11	62.78	65.56
	BiCVM	GCM	mBERT	Mod. mBERT
	62.33	62.78	71.96	68.01

Table 7: QA results (F1)

Data	mBERT	Mod. mBERT
NLI En-Hi	61.09	63.1

Table 8: NLI results (Accuracy)

Overall, the cross-lingual and mBERT models perform better for *English-Spanish* as compared to *English-Hindi*. This could be due to several reasons.

- English and Spanish are similar languages, with both mostly retaining individual word order while code-switching, which is not the case for English and Hindi.
- Romanized Hindi does not use standardized spellings, and errors made by the transliterator could have influenced the results.
- We use Twitter and social media data to train cross lingual word embeddings for *English-Spanish* which are similar in domain to the task datasets, while we use the IITB and query-based parallel corpora for *English-Hindi* which is generic in domain, constrained by the available resources at hand.

We find that for most tasks, modified mBERT performs better than mBERT. In cases where this is not true (QA En-Hi; FG En-Hi), the difference in accuracy between the two models is small. This could be attributed to errors made by the transliterator or corpus differences, but in general we observe that the *modified En-Hi* mBERT model does not significantly outperform the base mBERT model. Given the promising results obtained by modified mBERT, it would be interesting to pre-train a language model for code-switched data which is trained on the monolingual corpora of languages involved and fine-tuned on GCM as proposed to compare against fine-tuning mBERT itself, which is trained on multiple languages.

We find that accuracies vary across tasks in the GLUECoS benchmark, and except in the case of LID, code-switched NLP is far from solved. This is particularly stark in the case of Sentiment and NLI, which are three and two way classification tasks respectively. Modified mBERT performs only a little over chance, which shows that we are still in the early days of solving NLI for code-switched languages, and also indicates that our models are far from truly being able to understand code-switched

language. We plan to release the NLI dataset to the community, which we hope will spur research in this area.

6 Conclusion

In this paper, we introduce the first evaluation benchmark for code-switching, GLUECoS. The benchmark contains datasets in *English-Hindi* and *English-Spanish* for six NLP tasks - LID, POS tagging, NER, Sentiment Analysis, Question Answering and a new code-switched *Natural Language Inference* task. We test various embedding techniques across all tasks and datasets and find that multilingual BERT outperforms cross-lingual embedding techniques on all tasks. We also find that for most datasets, a modified version of mBERT that has been fine-tuned on synthetically generated code-switched data with a small amount of real code-switched data performs best. This indicates that while multilingual models do go a long way in solving code-switched NLP, they can be improved further by using real and synthetic code-switched data, since the distributions in code-switched languages differ from the two languages being mixed.

In this work, we use standard architectures to solve each NLP task individually and vary the embeddings used. In future work, we would like to experiment with a multi-task setup wherein tasks with less training data can significantly benefit from those having abundant labelled data, since most code-switched datasets are often small and difficult to annotate. We experiment with datasets having varied amounts of code-switching and from different domains and show that some tasks, such as LID and POS tagging are relatively easier to solve, while tasks such as QA and NLI have low accuracies. We would like to add more diverse tasks and language pairs to the GLUECoS benchmark in a future version.

All the datasets used in the GLUECoS benchmark are publicly available, and we plan to make the NLI dataset available for research use. We hope that this will encourage researchers to test multilingual, cross-lingual and code-switched embedding techniques and models on this benchmark.

References

- Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Mona Diab, Julia Hirschberg, and Thamar Solorio. 2019. Named entity recognition on code-switched data: Overview of the calcs 2018 shared task. *arXiv preprint arXiv:1906.04138*.
- Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Thamar Solorio, Mona Diab, and Julia Hirschberg. 2018. Proceedings of the third workshop on computational approaches to linguistic code-switching. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*.
- Fahad AlGhamdi, Giovanni Molina, Mona Diab, Thamar Solorio, Abdelati Hawwari, Victor Soto, and Julia Hirschberg. 2016. Part of speech tagging for code switched data. *EMNLP 2016*, page 98.
- Somnath Banerjee, Kunal Chakma, Sudip Kumar Naskar, Amitava Das, Paolo Rosso, Sivaji Bandyopadhyay, and Monojit Choudhury. 2016. Overview of the mixed script information retrieval (msir) at fire-2016. In *Forum for Information Retrieval Evaluation*, pages 39–49. Springer.
- Irshad Ahmad Bhat, Riyaz Ahmad Bhat, Manish Shrivastava, and Dipti Misra Sharma. 2018. Universal dependency parsing for hindi-english code-switching. *arXiv preprint arXiv:1804.05868*.
- Khyathi Chandu, Ekaterina Logiova, Vishal Gupta, Josef van Genabith, Günter Neumann, Manoj Chinnakotla, Eric Nyberg, and Alan W Black. 2018a. Code-mixed question answering challenge: Crowdsourcing data and techniques. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 29–38.
- Khyathi Chandu, Thomas Manzini, Sumeet Singh, and Alan W. Black. 2018b. [Language informed modeling of code-switched text](#). In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 92–97, Melbourne, Australia. Association for Computational Linguistics.
- Monojit Choudhury, Gokul Chittaranjan, Parth Gupta, and Amitava Das. 2014. Overview of fire 2014 track on transliterated search. *Proceedings of FIRE*, pages 68–89.
- Alexis Conneau, Guillaume Lample, Marc’Aurelio Ranzato, Ludovic Denoyer, and Hervé Jégou. 2017. Word translation without parallel data. *arXiv preprint arXiv:1710.04087*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- Mona Diab, Pascale Fung, Mahmoud Ghoneim, Julia Hirschberg, and Thamar Solorio. 2016. Proceedings of the second workshop on computational approaches to code switching. In *Proceedings of the Second Workshop on Computational Approaches to Code Switching*.
- Mona Diab, Julia Hirschberg, Pascale Fung, and Thamar Solorio. 2014. Proceedings of the first workshop on computational approaches to code switching. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*.
- Björn Gambäck and Amitava Das. 2014. On measuring the complexity of code-mixing. In *Proceedings of the 11th International Conference on Natural Language Processing, Goa, India*, pages 1–7. Citeseer.
- Goran Glavas, Robert Litschko, Sebastian Ruder, and Ivan Vulic. 2019. How to (properly) evaluate cross-lingual word embeddings: On strong baselines, comparative analyses, and some misconceptions. *arXiv preprint arXiv:1902.00508*.
- Gualberto A Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for modeling code-switching across corpora. In *INTERSPEECH*, pages 67–71.
- Karl Moritz Hermann and Phil Blunsom. 2014. [Multilingual models for compositional distributed semantics](#). In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 58–68, Baltimore, Maryland. Association for Computational Linguistics.
- Anupam Jamatia, Björn Gambäck, and Amitava Das. 2016. Collecting and annotating indian social media code-mixed corpora. In *International Conference on Intelligent Text Processing and Computational Linguistics*, pages 406–417. Springer.
- Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhat-tacharyya. 2018. The iit bombay english-hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*.
- Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. *arXiv preprint arXiv:1603.01360*.
- Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159.
- Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.
- Giovanni Molina, Nicolas Rey-Villamizar, Thamar Solorio, Fahad AlGhamdi, Mahmoud Ghoneim, Abdelati Hawwari, and Mona Diab. 2016. Overview for the second shared task on language identification in code-switched data. *EMNLP 2016*, page 40.

- Braja Gopal Patra, Dipankar Das, and Amitava Das. 2018. Sentiment analysis of code-mixed indian languages: An overview of sail_code-mixed shared task@ icon-2017. *arXiv preprint arXiv:1803.06745*.
- Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.
- Adithya Pratapa, Gayatri Bhat, Monojit Choudhury, Sunayana Sitaram, Sandipan Dandapat, and Kalika Bali. 2018a. 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.
- Adithya Pratapa, Monojit Choudhury, and Sunayana Sitaram. 2018b. Word embeddings for code-mixed language processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3067–3072.
- Pattabhi RK Rao and Sobha Lalitha Devi. 2016. Cmeel: Code mix entity extraction in indian languages from social media text@ fire 2016-an overview. In *FIRE (Working Notes)*, pages 289–295.
- Shruti Rijhwani, Royal Sequiera, Monojit Choudhury, Kalika Bali, and Chandra Shekhar Maddila. 2017. Estimating code-switching on twitter with a novel generalized word-level language detection technique. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1971–1982.
- Rishiraj Saha Roy, Monojit Choudhury, Prasenjit Majumder, and Komal Agarwal. 2013. Overview of the fire 2013 track on transliterated search. In *Post-Proceedings of the 4th and 5th Workshops of the Forum for Information Retrieval Evaluation*, page 4. ACM.
- Sebastian Ruder, Ivan Vulić, and Anders Søgaard. 2017. A survey of cross-lingual word embedding models. *arXiv preprint arXiv:1706.04902*.
- Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.
- Royal Sequiera, Monojit Choudhury, Parth Gupta, Paolo Rosso, Shubham Kumar, Somnath Banerjee, Sudip Kumar Naskar, Sivaji Bandyopadhyay, Gokul Chittaranjan, Amitava Das, et al. 2015. Overview of fire-2015 shared task on mixed script information retrieval. In *FIRE Workshops*, volume 1587, pages 19–25.
- Vinay Singh, Deepanshu Vijay, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. Named entity recognition for hindi-english code-mixed social media text. In *Proceedings of the Seventh Named Entities Workshop*, pages 27–35.
- Thamar Solorio, Elizabeth Blair, Suraj Maharjan, Steven Bethard, Mona Diab, Mahmoud Ghoneim, Abdelati Hawwari, Fahad AlGhamdi, Julia Hirschberg, Alison Chang, et al. 2014. Overview for the first shared task on language identification in code-switched data. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 62–72.
- Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019. How to fine-tune bert for text classification? *arXiv preprint arXiv:1905.05583*.
- David Vilares, Miguel A Alonso, and Carlos Gómez-Rodríguez. 2016. En-es-cs: An english-spanish code-switching twitter corpus for multilingual sentiment analysis. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 4149–4153.
- Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2019. Super-glue: A stickier benchmark for general-purpose language understanding systems. *arXiv preprint arXiv:1905.00537*.
- Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

A Additional Dataset Details

For each dataset wherein the training, development and test splits are not provided, we create balanced custom splits in an 8:1:1 ratio.

B Additional Training Details

We conduct each experiment for 5 random seed values and report the average of the results obtained.

B.1 Word-Level Tasks

For the word level tasks including Language Identification, Named Entity Recognition and Part of Speech tagging we make use of the sequence labeler.⁹ This implements a BiLSTM with a CRF layer on the top as described in (Lample et al., 2016). We use the adadelta optimizer with a learning rate of 1.0, dropout of 0.5 and a batch size of 32. We run the model for a maximum of 20 epochs and stop if the validation accuracy on the *best_model_selector* hyperparameter shows no improvement for 5 epochs continually. The *best_model_selector* hyperparameter is the weighted F1 score. The dimension of the word embeddings is 300.

We make use of the *transformers* library¹⁰ for the mBERT experiments. We use the AdamW optimizer with a learning rate of 5e-5, epsilon of 1e-8, and a batch size of 32, as suggested by (Devlin et al., 2018). We train for 5 epochs.

B.2 Sentence-Level Tasks

For the sentence level tasks (Sentiment Classification) we implement a BiLSTM with one hidden layer of dimension 256. We apply a dropout of 0.5, and use the Adam optimizer with a 0.001 learning rate and 1e-8 epsilon value. We use a batch size of 64 and train for a maximum of 15 epochs stopping if the validation accuracy continually drops for 3 epochs. The dimension of the word embeddings is 300.

We make use of the *transformers* library for the mBERT experiments. We use the AdamW optimizer with a learning rate of 5e-5, epsilon of 1e-8, and a batch size of 32, as suggested by (Devlin et al., 2018). We train for 5 epochs.

⁹<https://github.com/marekrei/sequence-labeler/tree/484a6beb1e2a2cccaac74ce717b1ee30c79fc8d8>

¹⁰<https://github.com/huggingface/transformers>

B.3 Sentence-Pair Tasks

For the embedding evaluations on the QA task, we make use of the BiDAF architecture¹¹ as proposed in (Seo et al., 2016). We keep the default training hyperparameters which include a learning rate of 0.5, a batch size of 1, training epochs as 5, a maximum context length of 400 tokens and a maximum question length of 50 tokens.

We make use of the SQuAD training script and the XNLI training script of the Transformers library with its default hyperparameters for the mBERT experiments.

C BERT LM fine-tuning

We take the bert model released by Google (*bert-base-multilingual-cased*) and fine-tune it for masked language modeling on 2 types of code-mixed datasets.

We use a curriculum wherein the model is first trained on generated code-mixed data (gCM) for 20 epochs and then on real code-mixed data (rCM) for 20 epochs.

For English-Hindi, the details of the datasets are as follows:

- 2.9M gCM sentences generated from the parallel corpus by (Kunchukuttan et al., 2018)
- 60k rCM sentences from the corpus by (Chandu et al., 2018b)

For English-Spanish, the details of the datasets are as follows:

- 8M gCM sentences generated from the corpus by (Rijhwani et al., 2017)
- 283k rCM sentences from the corpus by (Rijhwani et al., 2017)

¹¹<https://github.com/ElizaLo/Question-Answering-based-on-SQuAD>