Translate and Classify: Improving Sequence Level Classification for English-Hindi Code-Mixed Data

Devansh Gautam **Kshitij Gupta** **Manish Shrivastava**
International Institute of Information Technology Hyderabad
{devansh.gautam,kshitij.gupta}@research.iiit.ac.in,
m.shrivastava@iiit.ac.in

Abstract

Code-mixing is a common phenomenon in multilingual societies around the world and is especially common in social media texts. Traditional NLP systems, usually trained on monolingual corpora, do not perform well on code-mixed texts. Training specialized models for code-switched texts is difficult due to the lack of large-scale datasets. Translating code-mixed data into standard languages like English could improve performance on various code-mixed tasks since we can use transfer learning from state-of-the-art English models for processing the translated data. This paper focuses on two sequence-level classification tasks for English-Hindi code mixed texts, which are part of the GLUECoS benchmark - Natural Language Inference and Sentiment Analysis. We propose using various pre-trained models that have been fine-tuned for similar English-only tasks and have shown state-of-the-art performance. We further fine-tune these models on the translated code-mixed datasets and achieve state-of-the-art performance in both tasks. To translate English-Hindi code-mixed data to English, we use mBART, a pre-trained multilingual sequence-to-sequence model that has shown competitive performance on various low-resource machine translation pairs and has also shown performance gains in languages that were not in its pre-training corpus.

1 Introduction

In the last decade, social media has become a significant part of the lives of a large population in the world. Unlike previously popular communication platforms, online messaging is very informal, and in recent years, it has led to an increase in the usage of emojis, slang, and even a hybrid form of language, code-mixed language.

Code-mixed language is a mixture of multiple languages where words belonging to different languages are interleaved with each other in the same conversation. It is commonly used by multilingual speakers. It does not follow a formally defined structure and often varies from person to person, although some studies (Poplack, 1980; Belazi et al., 1994) have proposed linguistic constraints on code-switching. Code-mixing and code-switching are similar terms that slightly differ technically, but they are often used interchangeably by the research community. We will also be using them interchangeably in our paper.

In this paper, we work with English-Hindi code-mixed data. English-Hindi code-mixed language often called *Hinglish* is very common in India because of a large number of bilingual speakers who often use English in their professional lives while using Hindi in their personal lives. An example of an English-Hindi code-mixed sentence from a dataset released by Dhar et al. (2018) is shown below:

- **Original Sentence:** My brother always told me ki in retrospect, badi dikkatein chhoti lagti hain.
- **Gloss:** [My brother always told me] that [in retrospect], big problems small seem are.
- **Translation:** My brother always told me that, in retrospect, big problems seem to be small.

Although there is a large population globally that communicates using code-mixed languages, annotated datasets remain scarce even when the monolingual constituent languages have large-scale datasets. Recent work suggests that multilingual models trained on several monolingual datasets perform well with zero-shot cross-lingual transfer in code-switched settings (Patwa et al., 2020; Khanuja et al., 2020b). However, Khanuja et al. (2020b) conclude that their model had varying performance across tasks and especially struggled with NLI and sentiment analysis tasks. Another challenge with code-mixed language research is that,

Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching, pages 15–25
June 11, 2021. ©2021 Association for Computational Linguistics
https://doi.org/10.26615/978-954-452-056-4_003

unlike monolingual data, there are no formal data sources like news articles or books written in code-mixed languages. Instead, most research uses informal sources such as social media texts or messages, which are usually challenging to obtain. Also, most of the data is written in the Roman script, and Hindi words are transliterated informally without any standard rules. Instead, individuals generally provide a rough phonetic transcription of the intended word, which can vary from individual to individual due to any number of factors, including regional or dialectal differences in pronunciations, differing conventions of transcription, or simple idiosyncrasy (Roark et al., 2020). This makes it challenging to prepare reliable datasets to train robust deep learning models. Most of the existing datasets focus on a few language pairs and have been prepared by several shared task organizers.

To address these issues, we propose translating the code-mixed data to English (a high-resource language) and applying powerful models trained on English data to perform sequence-level classification tasks on the translated data. To translate the code-mixed data to English, we propose using mBART (Liu et al., 2020), a pre-trained multilingual sequence-to-sequence model. We experiment with our pipeline on two English-Hindi code-mixed sequence classification tasks of the GLUECoS (Khanuja et al., 2020b) benchmark - Natural Language Inference and Sentiment Analysis. We achieve state-of-the-art performance in both tasks. The code for our proposed system is available at `https://github.com/devanshg27/cm_translatify`.

The main contributions of our work are as follows:

- We explore the effectiveness of using mBART for low resource code-mixed Hinglish-English translation with transfer learning from Hindi-English translation.
- We propose performing sequence-level classifications on the code-mixed data by first translating it to English and then using powerful models trained on English data to classify the translated data.
- We achieve state-of-the-art performance on two classification tasks of the GLUECoS benchmark - Natural Language Inference and Sentiment Analysis with an absolute increase of 12.4% and 5.3%, respectively.

The rest of the paper is organized as follows. We discuss prior work related to code-mixed language processing and also discuss work related to machine translation, Natural language Inference, and Sentiment Analysis. We describe the translation system we use and show the effect of different training choices. We describe our pipeline for code-mixed sequence level classification tasks on the chosen tasks - Natural Language Inference and Sentiment Analysis and show its performance against past work. We conclude with a direction for future work and highlight our main findings.

2 Related Work

Code-mixing occurs when a speaker uses words belonging to different languages interleaved with each other in the same conversation. With the rise of social media and messaging platforms, there has been a significant increase in code-mixed language usage.

Several shared tasks have been conducted as a part of code-switching workshops (Diab et al., 2014, 2016; Aguilar et al., 2018b) which were held in notable conferences. These tasks include language identification (Solorio et al., 2014; Molina et al., 2016), named entity recognition (Aguilar et al., 2018a; Rao and Devi, 2016), information retrieval (Roy et al., 2013; Choudhury et al., 2014; Sequiera et al., 2015; Banerjee et al., 2018), Part-of-speech tagging (Jamatia et al., 2016), sentiment analysis (Patra et al., 2018; Patwa et al., 2020), and question answering (Chandu et al., 2018).

Although these tasks have helped progress code-switching language research, most tasks require building specialized systems for the specific task and language pair due to the limited dataset sizes. Recently, large pre-trained multilingual models have been used for various code-mixed tasks (Patwa et al., 2020; Khanuja et al., 2020b).

Machine Translation refers to translating a text from a source language to its counterpart in a target language using machines. It has widespread applications in the real world and has been an active area of research.

Earlier works in machine translation mostly focused on statistical or rule-based approaches. In contrast, neural machine translation gained popularity in the last decade after Kalchbrenner and Blunsom (2013) successfully proposed the first DNN model for translation. Recent works use transformer-based approaches (Vaswani et al., 2017). Some approaches utilize multilingual pre-training (Song et al., 2019; Conneau and Lample, 2019; Edunov et al., 2019; Liu et al., 2020); however, these works focus only on monolingual language pairs.

Despite the significant usage of English-Hindi code-mixing, there has been little work regarding English-Hindi code-mixed translation (Srivastava and Singh, 2020; Singh and Solorio, 2018; Dhar et al., 2018), which leads to a massive gap in communication as these texts can only be understood by people who are proficient in both these languages.

Natural Language Inference is the task of determining if the given "premise" supports a given "hypothesis" and classifying the hypothesis as true (entailment), false (contradiction), or undetermined (neutral). It is arguably one of the most fundamental tasks in natural language understanding. Wang et al. (2018) and Yin et al. (2019) suggest that various NLP tasks can be reduced to Natural Language Inference, which makes it an even more valuable task to solve.

Natural Language Inference for English texts has been an active area of research. It has been extensively studied under different tasks such as RTE (Recognizing Textual Entailment) (Dagan et al., 2006), NLI (Natural Language Inference) (Bowman et al., 2015), FEVER (Fact Extraction and VERification) (Thorne et al., 2018). In recent years, large-scale pre-trained models (Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019) have dominated these tasks and have achieved close-to-human performance.

Although NLI on English data has seen many advances, there has been little work on NLI for code-mixed data. Khanuja et al. (2020a) release the first NLI dataset for code-mixed languages. It consists of conversations from Hindi movies (Bollywood) as premises. Chakravarthy et al. (2020) compare the effectiveness of various approaches on the dataset.

Sentiment Analysis is the task of understanding the sentiment expressed in the text and classifying the text into positive, negative, or neutral classes. It has several applications such as customer feedback, marketing, and social media monitoring. There has been extensive research on sentiment analysis of English texts with various shared tasks and datasets. Sentiment analysis for code-mixed texts is an essential task due to the widespread usage of code-mixed texts on social media in multilingual societies. There has been some work related to code-mixed sentiment analysis with a few shared tasks (Patra et al., 2018; Patwa et al., 2020). The participants of the task organized by Patwa et al. (2020) explored various approaches such as pre-trained language models, RNN, CNN, and word embeddings.

	Dhar et al. (2018)	**Srivastava and Singh (2020)**
# of sentences	6,096	13,738
# of tokens	63,913	200,326
# of Hindi tokens	37,673	103,887
# of English tokens	16,182	38,511
# of 'Other' tokens	10,094	57,928

Table 1: The statistics of the English-Hindi code-mixed sentences in the two datasets we use. We use the language tokens predicted by the CSNLI library for both the datasets.

3 Translating Code-Mixed Text

In this section, we describe our proposed model, which uses mBART (Liu et al., 2020) to translate code-mixed texts to English.

3.1 mBART

We fine-tune mBART, which is a multilingual sequence-to-sequence denoising auto-encoder. It has been pre-trained using the BART (Lewis et al., 2020) objective on large-scale monolingual corpora of 25 languages extracted from Common Crawl[1] (Wenzek et al., 2020; Conneau et al., 2020). Both English and Hindi are part of the pre-training corpus with 55,608 million tokens (300.8 GB) and 1,715 million tokens (20.2 GB), respectively. It uses a standard sequence-to-sequence Transformer architecture (Vaswani et al., 2017), with 12 encoder and decoder layers each and a model dimension of 1024 on 16 heads resulting in ~680 million parameters.

3.2 Data Preparation

We use the datasets released by Dhar et al. (2018) and Srivastava and Singh (2020), the statistics of the datasets are provided in the Table 1. Since both the datasets contain Hindi words in Roman script, we use the CSNLI library[2] (Bhat et al., 2017, 2018) as a preprocessing step. It transliterates the Hindi words to Devanagari and also performs text normalization. We split the datasets into an 8:1:1

[1] https://commoncrawl.org/

[2] https://github.com/irshadbhat/csnli

train:validation:test split. We merge the training and validation sets of the two datasets and use the merged datasets for all our experiments.

We also use the dataset released by Kunchukuttan et al. (2018) which contains parallel sentences for English and Hindi. We use the training set, which contains 1,609,682 sentences, for training our systems.

3.3 Optimization

We use the implementation of mBART available in the fairseq library[3] (Ott et al., 2019). We fine-tune on 4 Nvidia GeForce RTX 2080 Ti GPUs with an effective batch size of 1024 tokens per GPU. We use the Adam optimizer ($\epsilon = 10^{-6}, \beta_1 = 0.9, \beta_2 = 0.98$) (Kingma and Ba, 2015) with 0.2 label smoothing, 0.3 dropout, 0.1 attention dropout and polynomial decay learning rate scheduling. We validate the models every 8000 steps and select the best checkpoint based on the lowest validation loss. To train our systems efficiently, we prune mBART's vocabulary by removing the tokens which are not present in any of the datasets mentioned in the previous section.

We compare the following 3 strategies for fine-tuning mBART:

- **mBART-cm:** We fine-tune mBART on the merged dataset with parallel English-Hindi code-mixed sentences. We fine-tune for 20,000 steps with 2,500 warm-up steps and a learning rate of $3 * 10^{-5}$.

- **mBART-hien:** We fine-tune mBART on the dataset with parallel English-Hindi sentences. We fine-tune for 80,000 steps with 2,500 warm-up steps and a learning rate of $3 * 10^{-5}$.

- **mBART-hien-cm:** We fine-tune mBART on the dataset with parallel English-Hindi sentences for 80,000 steps with 2,500 warm-up steps and a learning rate of $3 * 10^{-5}$, followed by further fine-tuning on on the merged dataset with parallel English-Hindi code-mixed sentences for 10,000 steps with 2,500 warm-up steps and a learning rate of 10^{-5}.

3.4 Results

We use BLEU scores as the metric for comparing our systems, the scores are computed using the SacreBLEU library[4] (Post, 2018) after tokenization using the `TweetTokenizer` available with the NLTK library[5] (Bird et al., 2009). The scores of our systems are shown in Table 2. We find that **mBART-hien** which was only fine-tuned for Hindi-English translation, performs considerably worse than the other models, showing that fine-tuning on English-Hindi code-mixed data improves the performance substantially. We also find that **mBART-hien-cm** has the best performance among the systems we consider. It uses transfer learning from Hindi to English translation to improve Hinglish-English translation.

Model	Datasets	
	Dhar et al. (2018)	Srivastava and Singh (2020)
mBART-hien	17.2	16.7
mBART-cm	30.5	31.6
mBART-hien-cm	**31.7**	**33.0**

Table 2: BLEU scores of our systems on the test sets of the two datasets.

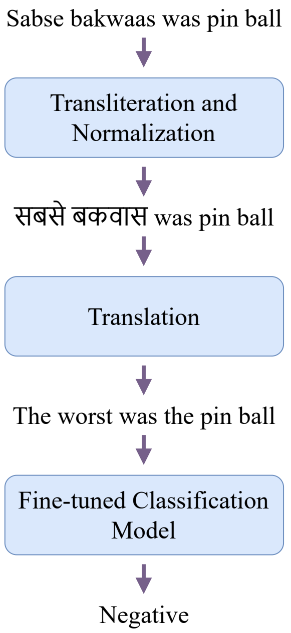

Sabse bakwaas was pin ball
Transliteration and Normalization
सबसे बकवास was pin ball
Translation
The worst was the pin ball
Fine-tuned Classification Model
Negative

Figure 1: The working of our pipeline for the task of code-mixed Natural Language Inference is demonstrated on an example (with minor edits) from the dataset (the details of the dataset are discussed later).

4 Code-Mixed Sequence-level Classification

In this section, we describe our approach for code-mixed sequence-level classification tasks using our

[3] https://github.com/pytorch/fairseq

[4] https://github.com/mjpost/sacrebleu

[5] https://www.nltk.org/

Model	Architecture	Dataset(Number of samples)				#Parameters
		SNLI (570k)	MultiNLI (433k)	FEVER-NLI (250k)	ANLI(R1,R2,R3) (170k)	
(1) (Liu et al., 2019)	RoBERTa $_{\text{large}}$		✓			~355M
(2) (Nie et al., 2020)	RoBERTa $_{\text{large}}$	✓	✓	✓	✓	~355M
(3) (Nie et al., 2020)	XLNet $_{\text{large}}$	✓	✓	✓	✓	~340M
(4) (Nie et al., 2020)	ALBERT $_{\text{xxlarge}}$	✓	✓	✓	✓	~223M
(5) (He et al., 2021)	DeBERTa $_{\text{large}}$		✓			~390M

Table 3: The pre-trained checkpoints we use along with their architecture, number of parameters and finetuning datasets.

	Train Set	Dev Set	Test Set
# of sentences	1,392	400	447
# of entailed sentences	696	200	224
# of contradictory sentences	696	200	223
# of tokens	123,366	33,932	40,072
# of Hindi tokens	75,865	20,837	24,413
# of English tokens	19,952	5,457	6,624
# of 'Other' tokens	27,549	7,638	9,035

Table 4: The statistics of the Natural Language Inference dataset. We use the language tokens predicted by the CSNLI library.

translation system. Our pipeline is shown in Figure 1. We evaluate the performance of our pipeline on two tasks - Natural Language Inference and Sentiment Analysis.

4.1 Natural Language Inference

4.1.1 Data Preparation

We use the dataset released by Khanuja et al. (2020a), which is a part of the GLUECoS benchmark. The dataset consists of code-mixed conversations from Hindi Movies (*Bollywood*) as premises that have been annotated with hypotheses that are either entailed or contradicted by the conversational premise. The statistics for the dataset are shown in Table 4. Since the dataset consists of Hindi words in Roman script, we use the CSNLI library to transliterate the Hindi words to Devanagari and perform text normalization. The data is then translated to English using our best-performing translation system - **mBART-hien-cm**. The dataset has a split between a train set and a test set with 1792 and 447 premise-hypothesis pairs in each, respectively. We split the train set into a validation set to create a 3.5:1:1.25 train:validation:test split finally.

4.1.2 System Overview

Our systems use different models which have shown competitive performance on Natural Language Inference for English texts. We use publicly available checkpoints for each model, which have been fine-tuned for Natural Language Inference on various English datasets such as SNLI (Bowman et al., 2015), MultiNLI (Williams et al., 2018), FEVER-NLI (Nie et al., 2019), ANLI (R1, R2, R3) (Nie et al., 2020). We fine-tune the checkpoints further on the code-mixed data translated to English. The details about the checkpoints we use are shown in Table 3.

4.1.3 Optimization

For the implementation of our systems, we use the HuggingFace Transformers library[6] (Wolf et al., 2020) and the AdamW optimizer ($\epsilon = 10^{-8}, \beta_1 = 0.9, \beta_2 = 0.999, \text{wd} = 0.01$) available in PyTorch[7] (Paszke et al., 2019) with a learning rate of 10^{-6}. All models were fine-tuned using 4 Nvidia GeForce RTX 2080 Ti GPU with a batch size of 8. The maximum sequence length was 512 for **(1)** and **(2)** and 256 for the other models. We fine-tune the models for 5 epochs with validation every 100 steps and choose the model with the best performance on the validation set. We use cross-entropy as the loss function.

4.1.4 Results

We compare the performance of our systems against the system with the highest test set performance discussed in Chakravarthy et al. (2020) and the baselines provided by Khanuja et al. (2020b).

The performance of our systems is shown in Table 5. All our systems perform better than the current state-of-the-art. We find that **(2)** performs better than **(1)**, which shows that transfer learning from a larger English dataset improves the performance on code-mixed texts. The confusion matrix for the predictions from our best model is shown

[6] `https://huggingface.co/transformers/`

[7] `https://pytorch.org/`

Model	Accuracy
mBERT (Khanuja et al., 2020b)	61.09
Mod. mBERT (Khanuja et al., 2020b)	63.1
mod-mBERT (Chakravarthy et al., 2020)	62.41
(1) - RoBERTa $_{\text{large}}$	73.65 $\pm$0.82
(2) - RoBERTa $_{\text{large}}$	**75.53** $\pm$1.08
(3) - XLNet $_{\text{large}}$	68.97 $\pm$1.16
(4) - ALBERT $_{\text{xxlarge}}$	70.74 $\pm$1.66
(5) - DeBERTa $_{\text{large}}$	73.92 $\pm$0.61

Table 5: NLI Performance with different checkpoints: Mean and standard deviation of the metrics from 5 independent runs.

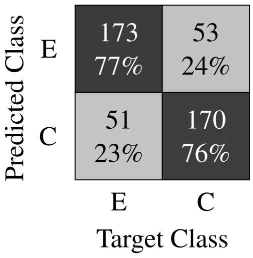


Figure 2: Confusion matrix of the test set predictions by our best model. The percentages show the ratio of the target class, which was predicted as that class. C: Contradictory, E: Entailed.

in Figure 2. We find that the performance of our system on entailed and contradictory statements is similar.

4.2 Sentiment Analysis

4.2.1 Data Preparation

We use the dataset released by Patra et al. (2018), which is part of the GLUECoS benchmark. The dataset was created by collecting code-mixed tweets using common Hindi words as search keywords. The tweets were annotated with word-level language tags and sentiment tags (positive, negative, or neutral). A transliterated version of the dataset is also provided where the Hindi words are in the Devanagari script. We use the transliterated version and translate it to English using **mBART-hien-cm** after normalizing the text with the `DevanagariNormalizer` function available in the IndicNLP Library[8] (Kunchukuttan, 2020). The statistics for the dataset are shown in Table 6. We use the provided train:validation:test split, which is in the ratio 8:1:1.

	Train Set	Dev Set	Test Set
# of sentences	10,079	1,260	1,262
# of negative sentences	2,319	283	290
# of neutral sentences	4,559	578	586
# of positive sentences	3,202	399	385
# of tokens	159,528	20,652	18,985
# of Hindi tokens	65,245	8,486	7,841
# of English tokens	62,678	8,028	7,453
# of 'Other' tokens	31,605	4,138	3,691

Table 6: The statistics of the Sentiment Analysis dataset. We use the word-level language tags provided along with the dataset.

4.2.2 System Overview

We use the following models which have shown competitive performance on sentiment analysis of English tweets:

(1) BERTweet (Nguyen et al., 2020): A large-scale pre-trained language model for English tweets which has been pre-trained on a large corpus of 850M English tweets. It has the same architecture as BERT$_{\text{base}}$ with ~110M parameters.

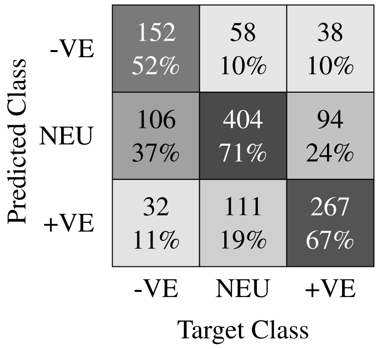
(2) RoB-RT (Barbieri et al., 2020): The pre-trained RoBERTa$_{\text{base}}$ model which has been re-trained on a corpus of 58M English tweets. It has ~125M parameters.

We use publicly available checkpoints of the above models, which have been fine-tuned on the sentiment analysis dataset released for SemEval-2017 Task 4 (Rosenthal et al., 2017) which is part of the TweetEval (Barbieri et al., 2020) benchmark. The dataset consists of ~60,000 tweets. We fine-tune the checkpoints further for sentiment analysis of code-mixed tweets that have been translated to English.

4.2.3 Optimization

For the implementation of our systems, we again use the HuggingFace Transformers library and the AdamW ($\epsilon = 10^{-8}, \beta_1 = 0.9, \beta_2 = 0.999, \text{wd} = 0.01$) optimizer available in PyTorch with a learning rate of 10^{-6}. All models were fine-tuned using 4 Nvidia GeForce RTX 2080 Ti GPU with a batch size of 16. The maximum sequence length was 128 for **(1) BERTweet** and 512 for **(2) RoB-RT**. We fine-tune the models for 5 epochs with validation every 100 steps and choose the model with

[8] http://anoopkunchukuttan.github.io/indic_nlp_library/



Predicted Class	-VE	NEU	+VE
-VE	152 52%	58 10%	38 10%
NEU	106 37%	404 71%	94 24%
+VE	32 11%	111 19%	267 67%

Figure 3: Confusion matrix of the test set predictions by our best model. The percentages show the ratio of the target class, which was predicted as that class. -VE: Negative, NEU: Neutral, +VE: Positive.

Model	**F1-weighted**
IIIT-NBP (Patra et al., 2018)	56.9
mBERT (Khanuja et al., 2020b)	58.24
Mod. mBERT (Khanuja et al., 2020b)	59.35
(1) BERTweet	**64.6** ± 0.3
(2) RoB-RT $_{\text{base}}$	**64.6** ± 0.4

Table 7: Sentiment Analysis Performance with different checkpoints: Mean and standard deviation of the metrics from 5 independent runs.

the best performance on the validation set. We use cross-entropy as the loss function.

4.2.4 Results

We compare the performance of our systems against the system achieving the highest score in the task organized by Patra et al. (2018) and the two best-performing baselines provided by Khanuja et al. (2020b).

The performance of our systems is shown in Table 7. Both the systems we consider have similar performance and perform better than the current state-of-the-art. The confusion matrix for the predictions from our best model is shown in Figure 3. We find that our model struggles with negative sentiment tweets and misclassifies them as neutral sentiment in 37% of cases.

5 Conclusion

In this paper, we demonstrate that mBART can be used to translate English-Hindi code-mixed sentences to English and show that transfer learning from Hindi-English translation improves its performance on code-mixed translation. We evaluate how our translation system can be used to improve performance in code-mixed sequence classification tasks. We develop a pipeline that uses our translation system to translate code-mixed data to English and then uses large-scale pre-trained English models for the downstream tasks. Our experiments show that our pipeline achieves state-of-the-art performance on two tasks of the GLUECoS benchmark - Natural Language Inference and Sentiment Analysis.

The performance of our pipeline shows that improving code-mixed translation can improve the performance of several code-mixed tasks. In future work, we would like to improve our translation system by creating a larger parallel corpus or synthetically generating parallel sentences for data augmentation. We would also like to extend our system to other code-mixing language pairs.

Acknowledgments

We would like to thank the anonymous reviewers for their time and insightful comments.

References

Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Mona Diab, Julia Hirschberg, and Thamar Solorio. 2018a. 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.

Gustavo Aguilar, Fahad AlGhamdi, Victor Soto, Thamar Solorio, Mona Diab, and Julia Hirschberg, editors. 2018b. *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*. Association for Computational Linguistics, Melbourne, Australia.

Somnath Banerjee, Kunal Chakma, Sudip Kumar Naskar, Amitava Das, Paolo Rosso, Sivaji Bandyopadhyay, and Monojit Choudhury. 2018. Overview of the mixed script information retrieval (msir) at fire-2016. In *Text Processing*, pages 39–49, Cham. Springer International Publishing.

Francesco Barbieri, Jose Camacho-Collados, Luis Espinosa Anke, and Leonardo Neves. 2020. TweetEval: Unified benchmark and comparative evaluation for tweet classification. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1644–1650, Online. Association for Computational Linguistics.

Hedi M. Belazi, Edward J. Rubin, and Almeida Jacqueline Toribio. 1994. Code switching and x-bar theory: The functional head constraint. *Linguistic Inquiry*, 25(2):221–237.

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.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural Language Processing with Python*. O'Reilly Media.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Sharanya Chakravarthy, Anjana Umapathy, and Alan W Black. 2020. Detecting entailment in code-mixed Hindi-English conversations. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 165–170, Online. Association for Computational Linguistics.

Khyathi Chandu, Ekaterina Loginova, Vishal Gupta, Josef van Genabith, Günter Neumann, Manoj Chinnakotla, Eric Nyberg, and Alan W. Black. 2018. Code-mixed question answering challenge: Crowd-sourcing data and techniques. In *Proceedings of the Third Workshop on Computational Approaches to Linguistic Code-Switching*, pages 29–38, 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, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The pascal recognising textual entailment challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg. Springer Berlin Heidelberg.

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.

Mrinal Dhar, Vaibhav Kumar, and Manish Shrivastava. 2018. Enabling code-mixed translation: Parallel corpus creation and MT augmentation approach. In *Proceedings of the First Workshop on Linguistic Resources for Natural Language Processing*, pages 131–140, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Mona Diab, Pascale Fung, Mahmoud Ghoneim, Julia Hirschberg, and Thamar Solorio, editors. 2016. *Proceedings of the Second Workshop on Computational Approaches to Code Switching*. Association for Computational Linguistics, Austin, Texas.

Mona Diab, Julia Hirschberg, Pascale Fung, and Thamar Solorio, editors. 2014. *Proceedings of the First Workshop on Computational Approaches to Code Switching*. Association for Computational Linguistics, Doha, Qatar.

Sergey Edunov, Alexei Baevski, and Michael Auli. 2019. Pre-trained language model representations for language generation. 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 4052–4059, Minneapolis, Minnesota. Association for Computational Linguistics.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention. In *International Conference on Learning Representations*.

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.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent continuous translation models. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1700–1709, Seattle, Washington, USA. Association for Computational Linguistics.

Simran Khanuja, Sandipan Dandapat, Sunayana Sitaram, and Monojit Choudhury. 2020a. A new dataset for natural language inference from code-mixed conversations. In *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching*, pages 9–16, Marseille, France. European Language Resources Association.

Simran Khanuja, Sandipan Dandapat, Anirudh Srinivasan, Sunayana Sitaram, and Monojit Choudhury. 2020b. GLUECoS: An evaluation benchmark for code-switched NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3575–3585, Online. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Anoop Kunchukuttan. 2020. The IndicNLP Library. `https://github.com/anoopkunchukuttan/indic_nlp_library/blob/master/docs/indicnlp.pdf`.

Anoop Kunchukuttan, Pratik Mehta, and Pushpak Bhattacharyya. 2018. The IIT Bombay English-Hindi parallel corpus. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

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.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. BERTweet: A pre-trained language model for English tweets. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 9–14, Online. Association for Computational Linguistics.

Yixin Nie, Haonan Chen, and Mohit Bansal. 2019. Combining fact extraction and verification with neural semantic matching networks. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6859–6866. AAAI Press.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 48–53, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

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.

Parth Patwa, Gustavo Aguilar, Sudipta Kar, Suraj Pandey, Srinivas PYKL, Björn Gambäck, Tanmoy Chakraborty, Thamar Solorio, and Amitava Das. 2020. SemEval-2020 task 9: Overview of sentiment analysis of code-mixed tweets. In *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, pages 774–790, Barcelona (online). International Committee for Computational Linguistics.

Shana Poplack. 1980. Sometimes i'll start a sentence in spanish y termino en espaÑol: toward a typology of code-switching 1. *Linguistics*, 18:581–618.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.

Pattabhi R. K. Rao and S. Devi. 2016. Cmee-il: Code mix entity extraction in indian languages from social media text @ fire 2016 - an overview. In *FIRE*.

Brian Roark, Lawrence Wolf-Sonkin, Christo Kirov, Sabrina J. Mielke, Cibu Johny, Isin Demirsahin, and Keith Hall. 2020. Processing South Asian languages written in the Latin script: the dakshina dataset. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2413–2423, Marseille, France. European Language Resources Association.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2017. SemEval-2017 task 4: Sentiment analysis in Twitter. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 502–518, Vancouver, Canada. Association for Computational Linguistics.

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*, FIRE '12 & '13, New York, NY, USA. Association for Computing Machinery.

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.

Thoudam Doren Singh and Thamar Solorio. 2018. Towards translating mixed-code comments from social media. In *Computational Linguistics and Intelligent Text Processing*, pages 457–468, Cham. Springer International Publishing.

Thamar Solorio, Elizabeth Blair, Suraj Maharjan, Steven Bethard, Mona Diab, Mahmoud Ghoneim, Abdelati Hawwari, Fahad AlGhamdi, Julia Hirschberg, Alison Chang, and Pascale Fung. 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, Doha, Qatar. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2019. MASS: Masked sequence to sequence pre-training for language generation. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 5926–5936. PMLR.

Vivek Srivastava and Mayank Singh. 2020. PHINC: A parallel Hinglish social media code-mixed corpus for machine translation. In *Proceedings of the Sixth Workshop on Noisy User-generated Text (W-NUT 2020)*, pages 41–49, Online. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. 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 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, Vishrav Chaudhary, Francisco Guzmán, Armand Joulin, and Edouard Grave. 2020. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 4003–4012, Marseille, France. European Language Resources Association.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. 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 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Wenpeng Yin, Jamaal Hay, and Dan Roth. 2019. Benchmarking zero-shot text classification: Datasets, evaluation and entailment approach. 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 3914–3923, Hong Kong, China. Association for Computational Linguistics.