Gated Convolutional Sequence to Sequence Based Learning for English-Hingilsh Code-Switched Machine Translation.

Suman Dowlagar
LTRC
IIIT-Hyderabad
suman.dowlagar
@research.iiit.ac.in

Radhika Mamidi
LTRC
IIIT-Hyderabad
radhika.mamidi
@iiit.ac.in

Abstract

Code-Switching is the embedding of linguistic units or phrases from two or more languages in a single sentence. This phenomenon is practiced in all multilingual communities and is prominent in social media. Consequently, there is a growing need to understand code-switched translations by translating the code-switched text into one of the standard languages or vice versa. Neural Machine translation is a well-studied research problem in the monolingual text. In this paper, we have used the gated convolutional sequences to sequence networks for English-Hinglish translation. The convolutions in the model help to identify the compositional structure in the sequences more easily. The model relies on gating and performs multiple attention steps at encoder and decoder layers.

1 Introduction

Language is a social phenomenon. The day-to-day interactions are made possible via language. The adaptive nature of languages and the flexibility to use multiple languages in one text message might help the speakers to communicate efficiently. This form of language interaction/contact is considered to be an essential phenomenon, especially in multilingual societies. In bilingual or multilingual communities, speakers use their native tongue and their second language during interactions. This form of alternation of two or more languages is called Code-Switching (CS) (Muysken et al., 2000).

Through the advent of social media, people from around the world can connect and exchange information instantly. Users from a Multilingual community often express their thoughts or opinions on social media by mixing different languages in the same utterance (Dowlagar and Mamidi, 2021). This mixing or alteration of two or more languages is known as code-mixing or code-switching (Wardhaugh, 2011).

There are no standard grammar rules that are meant to be practiced in the code-switched text. The code-switched data often contain variations of spellings and grammar. The computational processing of code-mixed or code-switched data is challenging due to the nature of the mixing and the presence of non-standard variations in spellings and grammar, and transliteration (Bali et al., 2014). Because of such linguistic complexities, code-switching poses several unseen difficulties in fundamental fields of natural language processing (NLP) tasks such as language identification, part-of-speech tagging, shallow parsing, Named entity recognition, sentiment analysis, offensive language identification etc.

To encourage research on code-mixing text, the Computational Approaches to Linguistic Code-Switching (CALCS) community has organized several workshops on language identification, Named Entity Recognition (Aguilar et al., 2018). This task focus on machine translation in the code-switched environment in multiple language combinations and directions [1].

This paper presents a gated convolutional sequence to sequence encoder and decoder models (Gehring et al., 2017) for machine translation on the code-mixed text. We have used the convolutional model because of its sliding window concept to deal with contextual words and the convolutions to extract rich representations.

The paper is organized as follows. Section 2 provides related work on the code-switched text for machine translation. Section 3 provides information on the task and dataset. Section 4 describes the proposed work. Section 5 presents the experimental setup and the performance of the model. Section 6 concludes our work.

[1] https\protect\leavevmode@ifvmode\kern+.2222em\relax//code-switching.github.io/2021#shared-task

Proceedings of the Fifth Workshop on Computational Approaches to Linguistic Code-Switching, pages 26–30

https://doi.org/10.26615/978-954-452-056-4_004

#	English (source translation)	Hinglish (target translation)
1	Hello! do you like comedy, adventure, and animation movies?	namaskaar! kya aapako komedee, edavenchar aur eneemeshan philmen pasand hain?
2	It was a strange choice	bahut strange choice thi ye
3	are you still there?	TUM ABHEE BHI VAHAAN HO
4	Hello.	Hello.

Table 1: Example translations

2 Related Work

There is relatively less research in the field of the machine translation of the code-switched text, partially due to the relative lack of structured corpora and also potentially because it also poses significant linguistic challenges such as ambiguity in language identification, spelling variations, informal style of writing, Misplaced/skipped punctuation, etc. Nonetheless, some researchers have provided datasets to enable research in code-mixed machine translation, specifically in Hindi-English code-switched scenario (Srivastava and Singh, 2020; Dhar et al., 2018). Dhar et al. (2018) presented a parallel corpus of the 13,738 code-mixed Hindi-English sentences and their corresponding human translation in English. In addition, they also provided a translation pipeline built on top of Google Translate. The pipeline fragments the input sentence into multiple chunks and identifies the language of each word in the chunk before feeding it to google-translate. The pipeline gives a BLEU-1 metric of 0.153 on the given English dataset. Dhar et al. (2018) translated the 6,096 code-mixed English-Hindi sentences into English and presented a translation augmentation pipeline. The pipeline is presented as a pre-processing step and can be plugged into any existing MT system. The pre-processed data is then given to translation systems like Moses, Google Neural Machine Translation System (NMTS), and Bing Translator, where the pre-processed data with NMTS has outperformed all the baselines with a BLEU score of 28.4.

3 Task Description

The goal of this task is the machine translation for code-switching settings in multiple language combinations and directions, such as involving English, Hinglish, Spanish, Spanglish, Modern Standard Arabic, and Egyptian Arabic languages. The code-mixed dataset is obtained from comments/posts from social media. In this paper, we have focussed on the English-Hinglish dataset. The English-Hinglish code-mixed dataset has 8060 train, 952 dev, and 960 test with source, and target translations. The task is to translate the given English sentence into a code-mixed Hindi-English sentence. The examples of the given English-Hinglish translation are given in the table 1.

In the first translation, one can see that the Hinglish sentence has a mixture of non-standard variations of words such as komedee(comedy), edavenchar(adventure), eneemeshan(animation), and the second translation exhibits the switching of English and Hindi phrases. In the third translation, the sentence is completely translated to Hindi (with roman script). The fourth translation shows that no translation is followed. The above sentences depict the diversity of the code-mixed translations, thus making the research and translation of the code-mixed text a complex task.

4 The proposed work

This section presents the proposed gated convolutional neural networks with encoder and decoder models for machine translation from English to code-mixed Hinglish text. The encoder model encode the source sentence into a vector and the decoder model takes the encoder information and decodes the given target sentences. The encoded vector is also known context vector. The context vector can be visualized as an abstract representation of the entire input sentence. The vector is decoded by a decoder model that learns to output the target sentence. The context needs to contain all of the information about the source sentence. It can be done by using attention.

Our encoder and decoder attention-based models use convolutions to encode the source sentence and to decode it into the target sentence. The convolutional layer uses filters. These filters have a window size. For example, if a filter has a window size of 3, then it can process three consecutive tokens. This window helps in determining the context. The convolutional layer has many of these

filters, where each filter will slide across the entire sequence by looking at all three consecutive tokens at a time. These filters will help extract different features in the given text and aid the machine translation model.

The description of the encoder and decoder convolutional models is given in the subsequent subsections.

4.1 Encoder

In the encoder model, each token in the source sentence is passed through an embedding layer. As the convolutional model has no recurrent connections, the model has no idea about the order of the tokens within a sequence. So it is necessary to add the positional embedding layer. In the positional embedding, the position of the tokens, including the start of the sequence and the end of the sequence, are encoded. Next, the token and positional embeddings are combined by elementwise sum. The obtained embedding vector contains the token and also its position within the sequence.

The given embedding vector is passed through a series of convolutional blocks. We follow the (Gehring et al., 2017) paper to implement the gated convolutional block architecture. It is formulated as,

$$h_i^l = v\left(\mathbf{W}^l\left[h_{i-k/2}^{l-1}, \ldots, h_{i+k/2}^{l-1}\right] + \mathbf{b}_w^l\right) + h_i^{l-1} \quad (1)$$

Where h_i^l is the output of the i^{th} sequence in l^{th} block. v is the gated gated linear units (GLU) (Dauphin et al., 2016) activation function. $\left[h_{i-k/2}^{l-1}, \ldots, h_{i+k/2}^{l-1}\right]$ are convolutional transformations of previous layer, $\mathbf{W}^l$ and $\mathbf{b}_w^l$ are learnable parameters and h_i^{l-1} is the residual output from the previous layer.

Passing the embedding vector through the convolutional blocks gives the convolved vector for each token in the given source sentence. The embedding vector is added as a residual connection is added to the convolved vector to get a combined vector.

4.2 Decoder

The decoder is similar to the encoder, with a few additional paddings to both the main model and the convolutional blocks inside the model.

In the decoder, the encoder convolved and combined outputs are used with attention. Finally, the output of the decoder is passed through a feed-forward layer to match the output target dimension in order to get the translated sentence.

5 Experiments

Here, we demonstrate the performance of the machine translation systems on the code-mixed text. We experiment with the popular RNN based encoder-decoder machine translation and vanilla transformer models and evaluate their performance on the given English-Hinglish machine translation task. We use BLEU metrics to evaluate system performance (Papineni et al., 2002).

5.1 Baseline MT models

RNN based encoder-decoder model (Bahdanau et al., 2014) The model uses RNN blocks to encode and decode the given sequence. The model allows the decoder to look at the entire source sentence at each decoding step by using attention.

Transformer We have implemented the Transformer model from the paper Vaswani et al. (2017). The transformer model uses multi-headed attention, layer normalizations, and feed-forward networks to implement the transformer models. The positional embeddings are used to remember the sequence of the sentence.

5.2 Hyperparameters and libraries

The parameters used to train our neural machine translation model are: the number of epochs used to train the model is 10. The Adam optimizer is used with cross-entropy loss with the gradient clipping of 0.1. The embedding and hidden dimensionalities are set as 256 and 512. The number of encoder and decoder convolutional layers used is 10. The default kernel window of size three is maintained. The dropout is kept at 0.25, and the maximum length used for the positional embeddings is 400. The Pytorch library is used to implement the model and is made publicly accessible [2].

5.3 Results and Error analysis

The results are guven in the table 2. From the table, it is clear that the convolutional model has obtained a better accuracy when compared to the vanilla transformer and encoder-decoder models. The use of convolutions and using the window size helped the convolutional model understand its context. We have even observed that the small length sequences and code-switching points are detected better by a convolutional model. As there is no

[2] https://github.com/suman101112/CMMT.git

Model	BLEU score (based on validation data)
Encoder-Decoder RNN model	1.52
Transformer model	2.51
proposed model (Conv seq2seq)	**2.58**

Table 2: BLEU metrics of the proposed model when compared to baselines.

Incorrect Translation	
Source Sentence (English)	Plus how many times are you going to leave your kid behind. I see they added another scene. How the robbers got caught and Kevin reuniting with his mom and family
Target Translation (Hinglish)	aur kitne baar apne bete ko chod doge. maine dekha hai ki woh log aur ek scene add kar diya. Robbers kaise pakde gaye aur kevin apni mom aur family se mila.
Translation by our model	mujhe us kahani pasand hein, jho tho yeh sach ko bahut pasand hein jab mein kabhi kabhi ko dekhna nahi hein lekin mein kabhi kabhi ko apney kabhi ko nahi nahi hein jab mujhe yakeen hai ki yah kuchh daraatee hai ki mujhe kabhi ko apney kabhi ko nahi
Proper Translation	
Source Sentence (English)	yes, it is good
Target Translation (Hinglish)	han, ye accha hai
Translation by our model	han, ye good hai

Table 3: Output of our convolutional sequence to sequence model on English-Hinglish text

recurrence in the convolutions, the computations are performed faster than the RNN's. Compared to recurrent networks, our convolutional approach allows discovering compositional structure in the sequences more easily since representations. Our model relies on gating and performs multiple attention steps. The vanilla transformer model did not perform well on this task because of the limited dataset used by the model. The vanilla transformer model is designed to be trained on larger datasets. It might be possible that the pre-trained transformer models can achieve better results when the dataset is finetuned on such models. The encoder-decoder RNN model performed worst and were very slow when compared to the other models.

During the error analysis we have found that there were repetitions in translations for the long sentences that are incorrectly translated by our model. This often lead to decrease in BLEU metric. The example is given table 3. Where as the short sentences are correctly translated by the given model. This is due to the low dependencies exhibited because of low window size and also due to presence of more Out of vocabulary (OOV) words because of the limited dataset. The improvement in the size of datasets will improve the translation accuracy of the proposed model.

6 Conclusion

This paper presents the performance of a neural machine translation model for the shared task on code-switched English-Hinglish translation. The model uses the convolutional sequence to sequence-based neural network architecture to translate the given sequence. The contextual window and the state-of-the-art convolution model helped the model learn better representations from the text and improved the model's performance compared to RNN encoder-decoder and vanilla transformer models. In the future, we wish to use the pre-trained BERT models and their ensembles and also consider other code-mixing factors such as pre-processing of the code-switched text to improve the quality of the code-switched machine translation.

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.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Kalika Bali, Jatin Sharma, Monojit Choudhury, and Yogarshi Vyas. 2014. “i am borrowing ya mixing?" an analysis of english-hindi code mixing in facebook. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 116–126.

Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier. 2016. Language modeling with gated convolutional networks. arxiv.

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.

Suman Dowlagar and Radhika Mamidi. 2021. Graph convolutional networks with multi-headed attention for code-mixed sentiment analysis. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 65–72.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *International Conference on Machine Learning*, pages 1243–1252. PMLR.

Pieter Muysken, Pieter Cornelis Muysken, et al. 2000. *Bilingual speech: A typology of code-mixing*. Cambridge University Press.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Vivek Srivastava and Mayank Singh. 2020. Phinc: a parallel hinglish social media code-mixed corpus for machine translation. *arXiv preprint arXiv:2004.09447*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *arXiv preprint arXiv:1706.03762*.

Ronald Wardhaugh. 2011. *An introduction to sociolinguistics*, volume 28. John Wiley & Sons.