niksss at HinglishEval: Language-agnostic BERT-based Contextual Embeddings with Catboost for Quality Evaluation of the Low-Resource Synthetically Generated Code-Mixed Hinglish Text

Nikhil Singh
Manipal University Jaipur
nikhil3198@gmail.com

Abstract

This paper describes the system description for the HinglishEval challenge at INLG 2022. The goal of this task was to investigate the factors influencing the quality of the code-mixed text generation system. The task was divided into two subtasks, quality rating prediction and annotators' disagreement prediction of the synthetic Hinglish dataset. We attempted to solve these tasks using sentence-level embeddings, which are obtained from mean pooling the contextualized word embeddings for all input tokens in our text. We experimented with various classifiers on top of the embeddings produced for respective tasks. Our best-performing system ranked 1st on subtask B and 3rd on subtask A. We make our code available here: https://github.com/nikhilbyte/Hinglish-qEval

1 Introduction

With the increase in popularity of social media platforms like blogs, Facebook, and Twitter in India, the amount of spoken and written Hinglish data has been on the rise. Hinglish is a blend of English and Hindi, involving code-switching between the above-mentioned languages. Due to the increasing number of users, the analysis of this new hybrid language using computational techniques has gotten important in a number of natural language processing applications like machine translation (MT) and speech-to-speech translation. (Bali et al., 2014),(Das and Gambäck, 2013).

Classical NLP problems such as language modeling (Pratapa et al., 2018), sentiment analysis (Singh and Lefever, 2020), (Chakravarthi et al., 2021),Hate-Speech Identification (Sreelakshmi et al., 2020) and language identification (Molina et al., 2016) are covered for Code-Mixed textual data. However, the generation and evaluation aspect of CM data hasn't been explored a lot.

This shared task aims to further the research of quality evaluation of the generated code-mixed text in a new way, proposing two tasks that will help quantify the quality of the synthetically generated CM text. Moreover, the organizers put forward another task that will help estimate the disagreement between the different human annotators, which further strengthens and reduce the noisiness of the ground-truth *quality* labels of the generated CM text sequence.

2 Related Work

There has been an increased interest in Code-Mixed data for various NLG tasks.(Yang et al., 2020) proposed a new pre-training strategy to tackle the complexities in CM text sequences in a non-traditional way. (Gautam et al., 2021) talks about generating low-resource Code-Mixed language from a high resource language such as English using various Seq2Seq models such as mBART (Liu et al., 2020). Other than this, various augmentation techniques were also proposed to improve the quality of generated Hinglish text sequences (Gupta et al., 2021). Due to its high linguistic diversity and lack of standardization, the basic Natural Language generation needs to be tackled and evaluated differently as shown in (Garg et al., 2021) where they propose different metrics to evaluate the quality of generated CM data and show why traditional translation metrics such as BLUE (Papineni et al., 2002) etc. cannot capture the quality evaluation properly.

3 Task Overview and Dataset

The task (Srivastava and Singh, 2021b) was divided into two subtasks. Subtask-A comprised of predicting the quality of the generated Hinglish sentences text on a scale of 1–10. 1 is low quality and 10 is the highest quality, considering the semantics and meaningfulness of the generated text sequence. However, the code-mixed language is seldom used in a formal setting, leading the popular evaluation techniques such as BLUE and WER being inappropriate. The organizers tried to tackle this using another way of evaluation to curb the noisiness of labels occurring in subtask-A by proposing another subtask-B. This subtask tests the capacity of the proposed models for estimating the disagreement between individual annotators, which often occurs when trying to evaluate the quality of informal text sequences.

The data for this task introduced in (Srivastava and Singh, 2021a) is called the HinGE dataset. Its dataset comprises 3,952 instances. Where a particular instance i comprises a text sequence triplet in English, Hindi, and hinglish language and *Average rating* as the label for subtask-A and *Annotator disagreement* as the label for subtask-B. These instances were shuffled and divided into three parts in a ratio of 70:10:20, leading to 2766, 395, and 791 data instances in train, validation, and test respectively. An instance of the dataset can be found in Figure 1.

4 Methodology

We attempted these tasks as a text triplet classification problem, wherein we have three text sequences side-by-side and a label attached to them. We analyzed the text sequences and found them to be clean and without any redundant information, hence we didn't perform any traditional pre-processing step. The following steps were taken to build the submitend system:

- Out of the three text sequences in a particular data instance, we feed the English and Hindi input sentences or texts into a transformer network named Language-agnostic BERT sentence embedding model (LaBSE) (Feng et al., 2020). The model produces contextualized word embeddings for all input tokens in our text into a shared latent space that produces similar vector/embeddings for similar sentences in a language-agnostic way. As we want a fixed-sized output representation (vector u), we need a pooling layer. Different pooling options are available, the most basic one is mean-pooling: We simply average all contextualized word embeddings the model is giving us. This gives us a fixed 768-dimensional output vector independent of how long our input text was.

SubTask	**FS**	**CK**	**MSE**
SubTask A	0.25062	0.08153	2.00000
SubTask B	0.26115	-	3.00000
Baseline A	0.26637	0.09922	2.00000
Baseline B	0.14323	-	5.00000

Table 1: Results on for Test Set

- The hinglish sequence was embedded using a BERT (Devlin et al., 2018) based model for hinglish text sequences available here [1] after using the same strategy as done for the English and Hindi counterparts.
- By this point, we have the three sentences/texts mapped to a fixed sized dense vector.
- The obtained vectors are then concatenated and fed into a catboost (Prokhorenkova et al., 2018) based classifier.
- The model was trained in a supervised manner using the default catboost classifiers with a logloss objective.A seed value of 42 was used to keep the model deterministic.
- The model took approximately 1.75 hours to train on CPU with a memory of 12Gb.
- The complete experiment was done on Google Colab Pro.
- The model architecture can be seen in Figure 2.

All our experiments were performed using SBERT [2]

5 Results

Three evaluation metrics F1-score (FS),Cohen's Kappa (CK),Mean Squared Error (MSE) were used to measure the performance of the submitted systems. We present the results obtained on test set along with the baselines in Table 1 .

6 Conclusion

We developed a system to evaluate the quality of machine-generated text sequences using a combination of deep learning feature vectors and machine

[1] https://huggingface.co/niksss/Hinglish-HATEBERT

[2] https://www.sbert.net/index.html

English	Hindi	Hinglish	Quality/Disagreement
Program module is a file that contains instruc...	माड्यूल, एक संचिका होती है, जिसमें या तो स्रोत...	module , ek program hoti hai , jismen ya to so...	7/6

Figure 1: A Single Instance from the Dataset

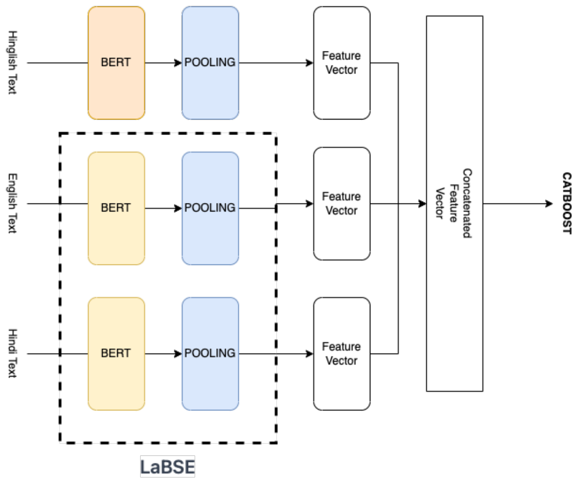

Figure 2: System Design

learning models. The results are nowhere near what would actually be used to evaluate the quality of the generated sequence. However, this is the first installment of the shared task and it sets off the baselines for future research on the same subject.

References

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, Doha, Qatar. Association for Computational Linguistics.

Bharathi Raja Chakravarthi, Ruba Priyadharshini, Navya Jose, Anand Kumar M, Thomas Mandl, Prasanna Kumar Kumaresan, Rahul Ponnusamy, Hariharan R L, John P. McCrae, and Elizabeth Sherly. 2021. Findings of the shared task on offensive language identification in Tamil, Malayalam, and Kannada. In *Proceedings of the First Workshop on Speech and Language Technologies for Dravidian Languages*, pages 133–145, Kyiv. Association for Computational Linguistics.

Amitava Das and Björn Gambäck. 2013. Code-mixing in social media text. the last language identification frontier? *Trait. Autom. des Langues*, 54:41–64.

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*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Ayush Garg, Sammed Kagi, Vivek Srivastava, and Mayank Singh. 2021. MIPE: A metric independent pipeline for effective code-mixed NLG evaluation. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*, pages 123–132, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Devansh Gautam, Prashant Kodali, Kshitij Gupta, Anmol Goel, Manish Shrivastava, and Ponnurangam Kumaraguru. 2021. Comet: Towards code-mixed translation using parallel monolingual sentences. In *CALCS*.

Abhirut Gupta, Aditya Vavre, and Sunita Sarawagi. 2021. Training data augmentation for code-mixed translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5760–5766, Online.

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. *Transactions of the Association for Computational Linguistics*, 8:726–742.

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.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

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

Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. 2018. Catboost: unbiased boosting with categorical features. *Advances in neural information processing systems*, 31.

Pranaydeep Singh and Els Lefever. 2020. Sentiment analysis for Hinglish code-mixed tweets by means of cross-lingual word embeddings. In *Proceedings of the The 4th Workshop on Computational Approaches to Code Switching*, pages 45–51, Marseille, France. European Language Resources Association.

K Sreelakshmi, B Premjith, and K.P. Soman. 2020. Detection of hate speech text in hindi-english code-mixed data. *Procedia Computer Science*, 171:737–744. Third International Conference on Computing and Network Communications (CoCoNet'19).

Vivek Srivastava and Mayank Singh. 2021a. Hinge: A dataset for generation and evaluation of code-mixed hinglish text. *arXiv preprint arXiv:2107.03760*.

Vivek Srivastava and Mayank Singh. 2021b. Quality evaluation of the low-resource synthetically generated code-mixed Hinglish text. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 314–319, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Zhen Yang, Bojie Hu, Ambyera Han, Shen Huang, and Qi Ju. 2020. Code-switching pre-training for neural machine translation. *arXiv preprint arXiv:2009.08088*.