Spartans@LT-EDI-EACL2021: Inclusive Speech Detection using Pretrained Language Models

Megha Sharma *
Amazon
meghshar@amazon.com

Gaurav Arora *
Jio Haptik
gaurav@haptik.ai

Abstract

We describe our system that ranked first in Hope Speech Detection (HSD) shared task and fourth in Offensive Language Identification (OLI) shared task, both in Tamil language. The goal of HSD and OLI is to identify if a code-mixed comment or post contains hope speech or offensive content respectively. Our work extends that of (Arora, 2020a) as we use their strategy to synthetically generate code-mixed data for training a transformer-based model RoBERTa and use it in an ensemble along with their pre-trained ULMFiT.

1 Introduction

Language has the ability to build relationships and forge connections but it is equally liable for creating barriers and impacting someone's sense of belonging. The language used on the internet has an impact on people across the globe. It is important to build language technology which makes everyone feel valued and included. We make contributions to this field by competing in two shared tasks:

1. Hope Speech Detection (HSD) at First Workshop on Language Technology for Equality, Diversity, Inclusion[1] (LT-EDI-2021)

2. Offensive Language Identification (OLI) in Dravidian Languages at First Workshop on Speech and Language Technologies for Dravidian Languages[2] (DravidianLangTech-2021)

Hope is considered significant for the well-being, recuperation and restoration of human life by health professionals. Hope speech reflects the belief that one can discover pathways to one's desired objectives and become motivated to utilize those pathways (Snyder et al., 1991; Chang, 1998). The goal of HSD task is to identify whether a YouTube comment contains hope speech or not. The datasets are available in English, code-mixed Tamil-English and Malayalam-English. OLI task intends to identify offensive language content in datasets comprising of comments/posts in code-mixed Tamil-English, Malayalam-English and Kannada-English which are collected from social media. Both datasets have been annotated at a comment level wherein a comment could comprise of more than one sentence but on average it has a single sentence.

Our work is an extension of the work done in (Arora, 2020a) as we use their synthetic code-mixed dataset for Tamil and ULMFiT model trained on that dataset. We pre-train a transformer based model RoBERTa (Liu et al., 2019) from scratch on code-mixed data and build an ensemble using ULMFiT and RoBERTa to achieve:

- Weighted F1 score of 0.61 for Tamil HSD and Rank 1 amongst 30 participating teams

- Weighted F1 score of 0.75 for Tamil OLI and Rank 4 amongst 30 participating teams

We review some related work from literature before explaining the details of our approach and the results. All experiments described in the paper can be reproduced using the source code available on GitHub[3].

2 Related Work

As noted on the LT-EDI 2021's website[4], this is the first shared task on HSD. Some work has been previously done for HSD (Palakodety et al., 2019),

*equal contribution
[1]https://sites.google.com/view/lt-edi-2021/home
[2]https://dravidianlangtech.github.io/2021/
[3]https://github.com/goru001/nlp-for-tanglish
[4]https://sites.google.com/view/lt-edi-2021

Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion, pages 188–192
April 19, 2021.

Metric	Offensive Language Identification			Hope Speech Detection		
Number of classes	6			3		
	Train	**Valid**	**Test**	**Train**	**Valid**	**Test**
Dataset size	35139	4388	4392	16160	2018	2021
%age of examples containing only Roman characters	80.4%	81%	81.2%	83.9%	83.8%	84.8%
min. no. of examples in a class	454	65	-	1961	263	-
max. no. of examples in a class	25425	3193	-	7872	998	-
avg. no. of examples in a class	5856.5	731.33	-	5386.67	672.67	-
min no. of tokens in an example	1	2	1	1	1	2
max no. of tokens in an example	183	93	138	204	159	187
avg no. of tokens in an example	12.04	12.03	11.89	9.57	9.97	11.07
median no. of tokens in an example	9	9	9	7	7	8

Table 1: Dataset statistics for OLI and HSD tasks

Model Architecture	Perplexity	Vocab size
RoBERTa	8.4	10000
ULMFiT	37.50	8000

Table 2: Validation set perplexity of RoBERTa and ULMFiT

but we are not aware of any work for HSD in Tamil language. OLI has been an area of active research in both academia and industry for the past two decades. Recent work has been done for OLI in Dravidian languages in HASOC task at FIRE (Mandl et al., 2020). HASOC task, which attracted over 40 research groups, consisted of building Hate Speech and Offensive Language identification systems by using datasets prepared by extracting comments/posts from YouTube and Twitter. In this paper, we build classification models for HSD and OLI using Transfer Learning, details of which have been explained in Section 3.

3 Methodology

In this section we look at classification datasets, discuss details of RoBERTa and ULMFiT models and the classifiers which are trained on top of these language models.

3.1 Dataset

Dataset for RoBERTa pre-training. We use synthetically generated code-mixed data for Tamil[5] prepared in (Arora, 2020a) to pretrain RoBERTa from scratch. The dataset is a collection of Tamil sentences written in Latin script. It was prepared by transliterating Tamil Wikipedia articles using Indic-Trans[6] library.

Classification datasets. Table 1 shows statistics of datasets of both tasks. We observe that the statistics are fairly consistent across train, valid and test sets. Classification dataset for HSD (Chakravarthi, 2020) has 3 classes whereas that in OLI has 6 classes. Both the classification datasets have significant class imbalance depicting real-world scenarios. Additionally, they contain code-mixed comments/posts in both Latin and native scripts, making the tasks challenging.

3.2 Modeling Details

We take a two-step approach to the problem by pre-training ULMFiT (Howard and Ruder, 2018) and RoBERTa (Liu et al., 2019) models on synthetically generated code-mixed language followed by an ensemble of two classifiers which are trained on top of ULMFiT and RoBERTa language models respectively.

3.2.1 ULMFiT Model

We use pre-trained ULMFiT model for code-mixed Tamil similar to the one used in (Arora, 2020b). Its embedding size is 400, number of hidden activations per layer are 1152 and the number of layers are 3. Two linear blocks with batch normalization and dropout have been added as custom head for the classifier with rectified linear unit activations for the intermediate layer and a softmax activation

[5]Pre-training dataset can be downloaded from https://github.com/goru001/nlp-for-tanglish

[6]https://github.com/libindic/indic-trans

Dropout Multiplicity	Batch Size	Epochs	Learning Rate	Adam Beta1	Adam Beta2	Adam Epsilon	LR Scheduler Type
0.1	8	3	5e-05	0.9	0.999	1e-08	LINEAR

Table 3: RoBERTa Classification Model Hyperparams for OLI

Model	Precision	Recall	F1 Score	Accuracy
Baseline KNN	0.62	0.72	0.65	0.72
ULMFit	0.73	0.78	0.73	0.78
RoBERTa	0.74	0.77	0.75	0.77
Ensemble	**0.75**	**0.79**	**0.76**	**0.79**

Table 4: Validation set results for OLI

Model	Precision	Recall	F1 Score	Accuracy
Baseline KNN Model	0.53	0.53	0.53	0.53
ULMFit	0.63	0.63	0.63	0.63

Table 5: Validation set results for HSD

at the last layer.

3.2.2 RoBERTa Model

RoBERTa model builds on BERT (Devlin et al., 2019) and modifies BERT's key hyperparameters, removes the next-sentence pre-training objective and trains with much larger mini-batches and learning rates. RoBERTa has the same architecture as BERT but it uses a different pre-training scheme and tokenizes text using Byte-Pair Encoding (Sennrich et al., 2016). We use implementation of RoBERTa in Huggingface's Transformers library[7] to pre-train the model from scratch. We train it for 7 epochs using a learning rate of 5e-5 and a dropout of 0.1 for attention and hidden layers. Table 2 compares perplexity of our pre-trained RoBERTa model with that of ULMFiT model which is also trained on the same code-mixed data.

3.3 Classification data pre-processing

We pre-process the classification datasets of both tasks by transforming comments in native script into Latin script using Indic-Trans library. This step is required because both of our pre-trained language models, ULMFiT and RoBERTa, are trained on code-mixed data in Latin script. We also perform other basic pre-processing steps like lowercasing and removing @username mentions. We did not apply other pre-processing steps such as stop words removal or removal of words that are too short since both of our pre-trained language models are trained on complete sentences and we wanted the model to figure out on its own if stop/short words are important for classification or not.

4 Experiment and Results

In this section we discuss details and results of our baseline model, classification models and ensemble strategy.

4.1 Experiment Details

4.1.1 OLI Task

Baseline Model. Our baseline uses KNN classifier on embeddings generated using code-mixed ULMFiT model from iNLTK[8] (Arora, 2020b). We set k=5 for all our experiments with uniform weighting on neighbors.

Ensemble. Our final model is a weighted ensemble of two classifiers where their weights sum to 1. Training of classifiers happens in two steps. First we fine-tune our language model on the downstream task of OLI and then train a classifier on the fine-tuned language model. Table 3 contains details of hyperparameters of the first classifier which is trained on our pre-trained RoBERTa. We train the second classifier using fine-tuned ULMFiT language model which is available in iNLTK. We experiment with different weights of classifiers in the ensemble. Best results on the validation set are obtained by setting a weight of 0.5 for both classifiers. Figure 1 shows the variation of weighted F1 score

[7] https://huggingface.co/transformers/model_doc/roberta.html

[8] https://github.com/goru001/inltk

Offensive Language Identifiation				Hope Speech Detection			
Precision	**Recall**	**F1 Score**	**Rank**	**Precision**	**Recall**	**F1 Score**	**Rank**
0.74	0.78	0.75	4/30	0.62	0.62	0.61	1/30

Table 6: Test set results for OLI task and HSD task

by changing weights of RoBERTa based classifier.

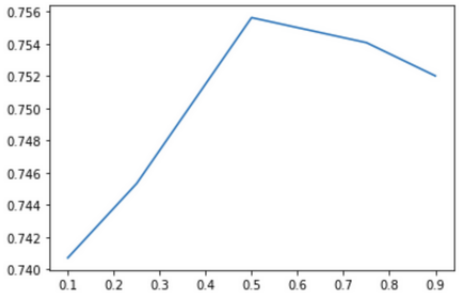

Figure 1: Change in Tamil OLI validation set F1 score on y-axis with change in RoBERTa weight shown on x-axis

4.1.2 HSD Task

We use a similar approach as that used for OLI task. Baseline model is built using the KNN algorithm and the final model is a classifier trained on a fine-tuned ULMFiT language model. Due to time and resource constraints, we weren't able to train RoBERTa based classifier.

4.2 Results

In both tasks weighted averaged Precision, weighted averaged Recall and weighted averaged F-Score are used as evaluation and ranking criteria. We participated in sub-task for Tamil and got Rank 4 in OLI task and Rank 1 in HSD task (Chakravarthi and Muralidaran, 2021). Table 4 shows the performance of different models on the validation set of former task. Best F1 score of 0.76 is obtained by using the ensemble of classifiers trained on RoBERTa and ULMFiT model which are pre-trained on code-mixed data. Table 5 contains results of models on the validation set of the latter task. We obtain an F1 score of 0.63 with ULMFiT based classifier. Results on the test-set for both the tasks have been shown in Table 6.

5 Conclusion and Future Work

In this paper we present RoBERTa language model for code-mixed Tamil which we pre-trained from scratch. Using transfer learning we fine-tune RoBERTa and ULMFiT language models on downstream tasks of OLI and HSD. We got Rank 4 in the former task using an ensemble of classifiers trained on RoBERTa and ULMFiT and Rank 1 in the latter task using classifier based on ULMFiT. In future research we will explore other transformer architectures like BERT (Devlin et al., 2018), T5 (Raffel et al., 2020), XLM (Conneau et al., 2019). We will work on improving code-mixed data generation strategies. We plan to create a dataset using a combination of native Tamil sentences, their transliterations and translations in English.

Acknowledgments

We would like to thank our teams at Amazon and Jio Haptik for motivating us to participate in these shared tasks. Please note that this work is not a by-product of any formal collaboration with Amazon and Jio Haptik. We participated in these tasks out of personal interests and in our own time.

References

Gaurav Arora. 2020a. Gauravarora@hasoc-dravidian-codemix-fire2020: Pre-training ulmfit on synthetically generated code-mixed data for hate speech detection.

Gaurav Arora. 2020b. iNLTK: Natural language toolkit for indic languages. In *Proceedings of Second Workshop for NLP Open Source Software (NLP-OSS)*, pages 66–71, Online. Association for Computational Linguistics.

Bharathi Raja Chakravarthi. 2020. HopeEDI: A multilingual hope speech detection dataset for equality, diversity, and inclusion. In *Proceedings of the Third Workshop on Computational Modeling of People's Opinions, Personality, and Emotion's in Social Media*, pages 41–53, Barcelona, Spain (Online). Association for Computational Linguistics.

Bharathi Raja Chakravarthi and Vigneshwaran Muralidaran. 2021. Findings of the shared task on Hope Speech Detection for Equality, Diversity, and Inclusion. In *Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

E. C. Chang. 1998. Hope, problem-solving ability, and coping in a college student population: some implications for theory and practice. *J Clin Psychol*, 54(7):953–962.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Jeremy Howard and Sebastian Ruder. 2018. Fine-tuned language models for text classification. *CoRR*, abs/1801.06146.

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.

Thomas Mandl, Sandip Modha, Anand Kumar M, and Bharathi Raja Chakravarthi. 2020. Overview of the hasoc track at fire 2020: Hate speech and offensive language identification in tamil, malayalam, hindi, english and german. In *Forum for Information Retrieval Evaluation*, FIRE 2020, page 29–32, New York, NY, USA. Association for Computing Machinery.

Shriphani Palakodety, Ashiqur R. KhudaBukhsh, and Jaime G. Carbonell. 2019. Kashmir: A computational analysis of the voice of peace. *CoRR*, abs/1909.12940.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

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

C. R. Snyder, C. Harris, J. R. Anderson, S. A. Holleran, L. M. Irving, S. T. Sigmon, L. Yoshinobu, J. Gibb, C. Langelle, and P. Harney. 1991. The will and the ways: development and validation of an individual-differences measure of hope. *J Pers Soc Psychol*, 60(4):570–585.