Hope Speech Detection: Identifying Positive Actors in Toxic Discourses

Mohammad Aflah Khan
aflah20082@iiitd.ac.in

Neemesh Yadav
neemesh20529@iiitd.ac.in

Raghav Sahni
raghav20533@iiitd.ac.in

Diksha Sethi
diksha20056@iiitd.ac.in

1. Abstract

With the rise of the Internet, there has been a significant increase in the number of marginalized people seeking support online. Online social media comments and posts have been analyzed using tools like hate speech recognition, offensive language identification, and abusive language detection to find and limit the spread of negativity. However, these studies primarily focus on analyzing negativity in the English Language, but the problem is not just restricted to harmful content. Research must also focus on encouraging and supportive online content as a form of positive reinforcement. Through our project, we aim to detect Hope Speech which reinforces positivity in online discourse. This problem is relevant as, alongside penalizing bad actors, we can reward good actors if we can identify them. Some downstream applications of our project include using the classifier to curate more data which can be used to train generative models. These models can then be deployed in toxic-online settings to spread positivity.
Overall, we propose building explainable classifiers for Hope Speech Detection, which can classify text into three classes “Hope Speech”, “Non-Hope Speech”, and “Non-English” for the English language. Then we plan to train the same models on Malayalam and Tamil to analyze the results and the shortcomings of directly using models in multi-lingual settings. We also plan to utilize Machine Learning models for this task as that is a relatively unexplored area. The few works on the subject discussed in the next section primarily focus on Deep Learning based methods and show minimal use of ML Techniques

2. Literature Review

Hope Speech and Models for Hope Speech Detection The first known work on Hope Speech [1] defines Hope Speech in a very limited fashion. Their work focused on analyzing trends in Youtube Comments during peak hostile times between India and Pakistan. For them Hope Speech is “Web content which plays a positive role in diffusing hostility on social media triggered by heightened political tensions”. Our work is more closely related to [2] where the definition is much broader and inclusive. The work defines Hope Speech as “Youtube comments/posts that offer support, reassurance, suggestions, inspiration and insight” and also provides a broad list of guidelines to annotate such speech. The work also creates the first publicly available dataset HopeEDI which provides data for 3 languages English, Tamil and Malayalam. We use this dataset for our work. Subsequent works emerged in the First Workshop On Language Technology For Equality, Diversity, Inclusion (LT-EDI-2021) where Hope Speech Detection was one of the tasks and several works were submitted in the competition. The finding paper [3] released after the Workshop discusses the various models and the best performing ones. The best performing models use Deep Learning based architectures such as BERT [4], RoBERTa [5], XLM-RoBERTa [6] etc. while traditional methods appear to lag. Hope Speech Detection again emerged as a Task in the Second Workshop on Language Technology For Equality, Diversity, Inclusion (LT-EDI-2022) where datasets for 2 new languages Spanish and Kannada were also added. The conclusions from the shared task are summarized in [7]. Another notable difference between the 2 Workshops is the change in evaluation metrics. The First Workshop used Weighted F1 while the Second Workshop used Macro F1 as the ranking metric which led to numerically lower but more meaningful scores. In our work we use both of these scores.

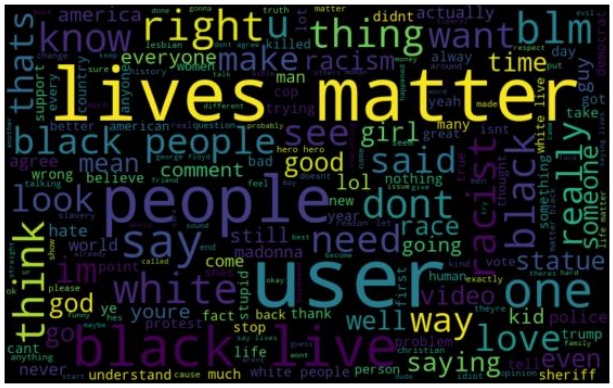
3. Dataset Description

3.1. EDA

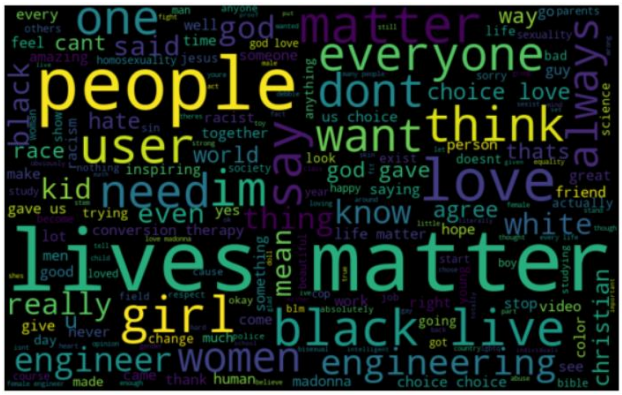
Class Distribution: There are three main classes in the dataset (Non-Hope Speech, Hope- Speech and Non-English).
Non-Hope-Speech: 25940
Hope-speech: 2484
Not-English: 27
Clearly the dataset is heavily biased. This was a major concern while selecting models to train on it. Moreover, balancing will be needed.

	English	Tamil	Malayalam
Train	22762	16160	8564
Development	2843	2018	1070
Test	2846	2020	1071
Total	28451	20198	10705

WordCloud: The word clouds for non-hope speech and hope-speech did not have any significant differences showing that the task at hand is not trivial and cannot be done simply by seeing the words in a sentence. However, the Non-English word cloud was quite different from the rest.



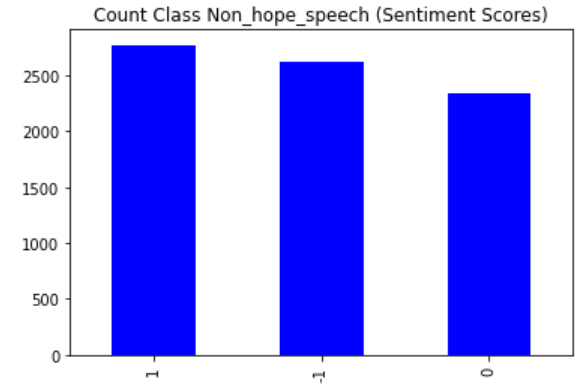
Word Cloud Non-Hope speech

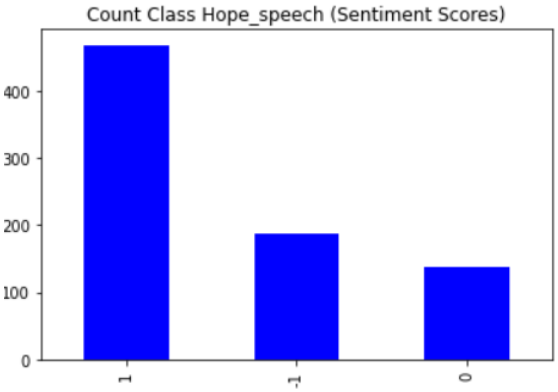


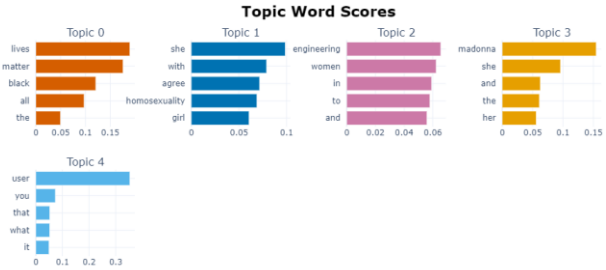
Word Cloud Hope speech

Sentiment Analysis: We also performed sentiment analysis on the data, even here for non-hope speeches there was no significant difference in the sentiment scores. For hope speech, however, a positive score was much more common than a negative or zero scores.

Topic Modeling: We perform topic modeling using BERTopic and evaluate the top-5 topics. We see clear identification markers amongst the topics. The topics captured in order are: Racism, Homosexuality, Women in Tech, References to Madonna, some General Words.







3.2. Preprocessing

To ensure that all the text inputs were uniform, we performed a few preprocessing steps. This was done to make sure that any of these textual inconsistencies would not affect the output in any way. The following steps were performed in the order in which they are listed:

1. Lower Case : To ensure that words with different case styles are not perceived in different contexts, we have lowercase each term in our dataset
2. Removal of URLs
3. Removal of usernames
4. Punctuation Removal
5. Removal of whitespaces

6. Removal of emojis : Used the emoji library for the operation
7. Tokenization: We have broken down the text sentences into smaller chunks or units using the Tweet Tokenizer. This step would essentially help to understand the meaning of the text by analysing smaller units at a time.

Several words such as usernames and URLs and special characters such as punctuation marks and emojis have been removed because we do not require them in our context.

3.3. Word Embeddings

At their core Word Embeddings are just ways to represent sentences as Vectors. There are numerous ways to convert sentences into their embeddings for our work we decide to use -

1. TF-IDF a statistical technique which tries to estimate how important each word is to a document in a collection of documents. We also use PCA to reduce the maximum number of dimensions to 1000.
2. Pretrained GloVe (2B tweets, 27B tokens, 1.2M vocab, uncased) [8], FastText (1 million word vectors, 16B Tokens) [9] and Word2Vec Embeddings (Uses subset of Google News dataset, contains 300-dimensional vectors for 3 million words and phrases) [10] using Gensim Library [11].
3. We also use 2 Transformer based pretrained embeddings, all-mpnet-base-v2 and all-MiniLM-L6-v2 from SBERT [12]. The former is the best pretrained model while the latter is the fastest. Subsequently we also perform PCA and create 2 new embeddings which retain 95% of the variance and also use them for our study.

We decide to stick to these Word Embeddings as they cover a wide range of techniques and different eras of Word Embeddings. TF-IDF derives its roots from Information Retrieval. GloVe, FastText and Word2Vec are some of the most used embeddings prior to the wide scale adoption of transformer based embeddings and SBERT is now one of the most used libraries by NLP researchers to get pre trained embeddings as empirically and theoretically Transformer Embeddings heavily outperform other embeddings in many tasks due to their Context Capturing Power.

4. Methodology

We used the already existing dataset made available by [1] to perform our experiments. We performed the experiments on the English posts from the original dataset. We kept experiments on other languages (Tamil, and Malayalam) to be our downstream task. The task for this dataset is stated to be a three-way classification task where we try to predict each of the {Hope_Speech, Non_Hope_Speech, Other_Languages} labels.

For our experiments, we tried out different word embedding techniques (GloVe [8], FastText [9], word2vec [10], TF-IDF, and Sentence-BERT [12]) and also tried various combinations with them by performing PCA or leaving them as is, to see if we can retain some amount of data while also compressing the dimensions, which we have reported in our final results. After getting the final embeddings we dumped them for future use, and each of us then took up different types of Classifier models from sklearn, and performed the stated task using all the embeddings thus generated. We have reported the Accuracy, {Weighted, Macro, Micro} {F1, Recall, and Precision} scores for each of our experiments.

5. Results and Analysis

5.1. Results

Algorithm	Embedding	Weighted F1 Test
Linear Discriminant Analysis	better-no-pca	0.927859
MLP	better-pca	0.926217
Logistic Regression	better-no-pca	0.92167
Perceptron	better-no-pca	0.920376
XGBoost	better-no-pca	0.918274
KNN	better-pca	0.917781
Random Forests	better-no-pca	0.910008
Naive Bayes	better-pca	0.906833
Decision Trees	better-pca	0.906336
ExtraTrees Classifier	better-no-pca	0.903285

Quadratic Discriminant Analysis	better-pca	0.89442
AdaBoost	better-pca	0.886988
Nearest Centroid	tf-idf	0.811931
K Means	word2vec	0.688256
K Medoids	better-no-pca	0.544692

5.2. Analysis

Our LDA Model is the best performing model closely followed by MLP, Logistic Regression, Perceptron and XGBoost. When we compare with the SOTA which is an XLM-RoBERTa model we see we are in the same neighborhood with the best model only having a difference of 0.23 F1 Points. MLP, Logistic Regression and Perceptron all cross the 0.92 Mark which was the score obtained by the 2nd Ranked Teams in the 2021 Shared Task. This tells us that proper care was not taken by the teams in using Word Embeddings and non-DL Models. The "Better Embedding" from Sentence BERT consistently performs well for our use case.

5.3. Conclusion

From our experiments we conclude that the techniques we used were on par with, if not better than, many deep learning SOTA performances which were reported for the task. Our best performing model (LDA) was only about a 0.23 Macro-F1 score behind the reported SOTA (XLM RoBERTa) for this task.

Our best performing model was when we applied Linear Discriminant Analysis (LDA) on the Sentence-BERT embeddings generated from the better pre-trained model, without performing PCA and it achieves a whopping 40 Weighted F1 score better than our worst performing model (K-Medoids).

6. Timeline

We've followed our Timeline very closely and also done much more than was initially proposed. First 2 weeks were spent on Data Analysis and Data Preprocessing post which we started with creating Word Embeddings. The next 5 Weeks included creating a total of 8 Word Embedding Configurations and training models with each of them while also using Grid Search wherever possible. We've used a total of 15 different Model Architectures with 8 Embeddings Each Resulting in 120 Models in total which we compare using Weighted and Macro F1 Scores. We also perform Topic Modelling using LDA, Top2Vec and BERTopic to better understand and analyze our data.

6.1. Future Timeline

1. SVMs
2. Deep Learning Models (RNN, LSTM, Transformers, PreTrained Models)
3. Custom Embeddings using Contrastive Learning Based Models such as Word2Vec & GLoVe
4. Try Data Augmentation Techniques to Balance the Dataset
5. Try Explainability techniques

7. Contributions

Neemesh:
1. Word Embeddings
2. Data Preprocessing
3. Random Forests, Perceptron, MLP, K-Medoids
4. Report and Presentation

Aflah:
1. GLoVe, FastText Embeddings
2. EDA
3. Decision Trees, Random Forests, Extra Trees, Linear & Quadratic Discriminant Analysis
4. Report and Presentation

Raghav:
1. EDA
2. XGBoost, AdaBoost, KMeans, KNN
3. TF-IDF Embeddings
4. Report and Presentation

Diksha:
1. Data Preprocessing
2. Logistic Regression, Naive Bayes, Nearest Centroid
3. Word2Vec Embeddings
4. Report and Presentation

References

[1] Shriphani Palakodety, Ashiqur R. KhudaBukhsh and Jaime G. Carbonell. Hope Speech Detection: A Computational Analysis Of The Voice Of Peace. In 24th European Conference on Artificial Intelligence - ECAI 2020

[2] Bharathi Raja Chakravarthi. 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 Emotions in Social media, pages 41–53

[3] Bharathi Raja Chakravarthi and Vigneshwaran Muralidaran. Findings of the Shared Task on Hope Speech Detection for Equality, Diversity, and Inclusion. Proceedings of the First Workshop on Language Technology for Equality, Diversity and Inclusion, pages 61–72

[4] Jacob Devlin, Ming-Wei Chang, Kenton Lee and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of NAACL-HLT 2019, pages 4171–4186

[5] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach.

[6] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzman, Edouard Grave, Myle Ott, Luke Zettlemoyer and Veselin Stoyanov. Unsupervised Cross-lingual Representation Learning at Scale. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 8440–8451

[7] Bharathi Raja Chakravarthi, Vigneshwaran Muralidaran, Ruba Priyadharshini, Subalalitha Cn, John McCrae, Miguel Ángel García, Salud María Jiménez-Zafra, Rafael Valencia-García, Prasanna Kumaresan, Rahul Ponnusamy, Daniel García-Baena and José García-Díaz. Overview of the Shared Task on Hope Speech Detection for Equality, Diversity, and Inclusion. Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion, pages 378 - 388

[8] Jeffrey Pennington, Richard Socher and Christopher D. Manning. GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)

[9] Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch and Armand Joulin. Advances in Pre-Training Distributed Word Representations. Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)

[10] Tomas Mikolov, Kai Chen, Greg Corrado and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space.

[11] Radim Rehurek and Petr Sojka. Software Framework for Topic Modeling with Large Corpora. The LREC 2010 Workshop On New Challenges For NLP Frameworks

[12] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing

[13] Pedregosa et al. Scikit-learn: Machine Learning in Python. JMLR 12, pp. 2825-2830, 201