IITH BrainStorm @ Constraint 2021 Shared Task : A Transformer Based Ensemble Approach to Hostility Detection in Hindi

Arkadipta De
ai20mtech14002@iith.ac.in
Indian Institute of Technology
Hyderabad, Telangana, India

Venkatesh E
ai20mtech14005@iith.ac.in
Indian Institute of Technology
Hyderabad, Telangana, India

ABSTRACT

Hostility detection is a very important and relevant problem in the field of Natural Language Processing. As the usage of social media like Facebook, Twitter increase, the problem of detecting posts that go against the community standards become more and more relevant. Most of the current works on hostility detection are in English language, which has lead to the lesser usability of hostility detectors outside of English sources. This paper proposes an effective neural model based on ensemble of the multilingual Bidirectional Encoder Representations of Transformer (BERT) for domain-agnostic hostility detection in Hindi language. A large variety of experiments including different varieties of architectures coupled with different pre-processing strategies are conducted. Proposed model achieves high accuracy in detection of hostile post which we coin as Coarse grained classification between non-hostile and hostile, outperforming the current state-of-the-art models. Moreover proposed model achieves high accuracy in classifying the hostile posts as fake, hate, defamation, offensive i.e the finer class of hostile posts by employing one versus rest approach. The proposed model outperforms the current state-of-the-art models in both Coarse Grained Evaluation as well as Fine Grained Evaluation of hostile posts detection.

CCS CONCEPTS

• **Computer systems organization → Neural networks**; • **Computing methodologies → Natural language processing**.

KEYWORDS

Neural Networks, Hostility Detection, Hindi

ACM Reference Format:
Arkadipta De and Venkatesh E. 2020. IITH BrainStorm @ Constraint 2021 Shared Task : A Transformer Based Ensemble Approach to Hostility Detection in Hindi. In *Proceedings of IIT Hyderabad, Information Retrieval (CS6370) Final Project (Project Report IR CS6370).* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Project Report IR CS6370, June 03–05, 2018, Hyderabad, IN

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

In the twenty first century, the use of social media and various online platform has increased dramatically. The unique number of users in the social media platforms like Facebook, Twitter, Orkut, Hike, Snapchat, LinkedIN are increasing more than ever. It does not just stop here. The usage of chat rooms, gaming platforms and streaming sites are also receiving thousands of new users each day. Platforms like Twitch, YouTube are also growing as social media platforms. Each day nearly 10 Billion posts are made in each social media about different topic, different issue such as politics, technology, sports, literature, science, research, entertainment, movies and the list continues to grow.

With the introduction of the mighty law of freedom of speech imposed on social media, people are free to post, react and comment about any topic they want to but often this freedom of speech comes at a harsh cost. It is evident from the current usage of social media posts that a large number of users take this freedom of speech too lightly and knowingly or unknowingly they cross the invisible line defined by it. Certain circumstances increases hate speech towards a certain community, religion or even country. As an example from a recent survey [1] it is seen that after the spread of COVID-19 all over the world originating from Wuhan province of China, there has been an increase of at least 900% increase in hate speech towards China and it's people on the famous platform Twitter, around 200% increase in traffic created by users on platforms that promote hate, offensive speech against Asian community and about 70% increase in hate speech among teenagers and kids online, and toxicity levels in gaming community has increased by around 40%.

These reports show that the environment in social media is becoming hostile day by day. The line of freedom of speech is being crossed and for other people it is being extremely difficult to use and leverage the joy and usefulness of any social media. Moreover spreading fake news about any topic (more importantly any recent topic) or hate speech against a community, religion or country, passing on offensive remarks about a race, colour etc. are extremely harmful for any civilised society. The world has seen the harmful outcome of hate speech and offensive comments which may lead to terrific violence like mob-lynching, communal violence and racism which leads to even death of people. These issues concerns us hence and demands adequate, appropriate and strict guidelines to detect and prevent such activities. Which brings us to the main objective of the task of Hostility detection, where we detect various forms of social media posts that are not conforming

[1] https://l1ght.com/ToxicityduringcoronavirusReportL1ght.pdf

to the guidelines of the social media such as posts expressing hatred against a specific person, community, race or posts containing offensive and vulgar words, posts spreading fake news on various topics etc.

There has been a lot of work on Hostility detection in such as hate speech detection on twitter, emoticons suggesting hate or offensive content in English language[1, 10, 31, 32]. Despite of the fact that Hindi is the third most spoken language in the world, there is a lack of work in this field. We work with a notable dataset in Hindi Hostility detection dataset [3]. We tackle the work of detecting hostility of Hindi posts in two ways - at first, we tackle the problem of binary classification of flagging a post as Hostile or Non-Hostile which we call as Coarse Grained Classification. Secondly, we further classify the hostile posts into four different categories (a multilabel classification problem) among four different sub-classes viz. Fake, Hate, Defamation and Offensive. At this point it is important to define the various classes according to [20] and [31] -

(1) **Fake News**: A claim or information that is verified to be not true. Posts belonging to click bait and satire/parody categories can be also categorized as fake news also.
(2) **Hate Speech**: A post targeting a specific group of people based on their ethnicity, religious beliefs, geographical belonging, race, etc., with malicious intentions of spreading hate or encouraging violence.
(3) **Offensive**: A post containing profanity, impolite, rude, or vulgar language to insult a targeted individual or group.
(4) **Defamation**: A mis-information regarding an individual or group, which is destroying their reputation publicly.
(5) **Non-Hostile**: A post with no hostility.

We propose a transformer based multilingual ensemble model that outperforms the current baseline by a wide margin. We also perform a wide range of architecture exploration in addition to a various techniques of pre-processing techniques.

The rest of the paper is organized as following -

Section 2 - Related Works, Section 3.1 - Dataset Description, Section 3.2 - Pre-processing of Data, Section 4 - Methodology (where we describe the architectures and methods used), Section 5 - Results and Discussion and Section 6 - Conclusion and Future Works.

2 RELATED WORKS

English being the most spoken language in the world and used language in the social media, there are some notable works in hostility detection in social media platforms on English. We mention some of the notable works as follows -

Fake News Detection: One of the most developed sub-field of hostility detection is fake news detection. The problem is very relevant and useful because of absent of fact checking organizations. [27] present comprehensive review of detecting fake news on social media, including fake news characterizations on psychology and social theories, existing algorithms from a data mining perspective, evaluation metrics and representative datasets. [34] does a comparative study on social media platform and fake news detection methodology and strategies. [25] proposes a three module based deep learning model based on Recurrent Neural Network to capture the temporal pattern of user activity on a given article CSI (Capture, Score, and Integrate) to mitigate the problem of fake news detection. [18] proposes methods to combine information from different available sources and combine them to tackle the problem of Multi-source Multi-class Fake-news Detection (MMFD). Another interesting work in this line is proposed by [26] where the authors present a hierarchical attention based deep learning model for automatic multi-domain fake news detection which outperforms existing state-of-the-art multi-domain deep learning models for fake news detection.

Hate Speech Detection: With the growing user number and diversity of users in different social media, the usage of vulgar words, in general the unfortunate event of spreading hatred through posts has increased rapidly, which has made this field a very relevant for research purpose. [12] proposed a lexicon based approach to hate speech detection in web discourses viz. web forums, blogs as well as rate the polarity of sentiment expressions in the detected text pieces. [11] proposed distributed low-dimensional representation based hate speech detection in online user comments on various sites. [2] proposed deep learning architectures to learn semantic word embeddings for hate speech detection in hate speech. In 2016, [33] did extensive research and provide a list of criteria founded in critical race theory and use that to annotate publicly available corpus. They also analyze the impact of various n-gram features that are decisive for detecting hate speech.

Works in Different Languages: In addition to problems of Hostility detection in English language, the overall problem of hostility detection such as hate speech detection, usage of offensive language detection and prevention, defamatory posts detection has also seen some development and attention in non-English language such as - [13] addresses the problem of offensive language detection in Arabic language where the authors use Convolutional Neural Network (CNN) and attention based Bidirectional Gated Recurrent Unit (Bi-GRU) models to mitigate the problem. [16] proposed a novel dataset of $50k$ annotated fake news in Bengali language. [17] proposed a method for classification of offensive tweets in the Hindi language. [4] analyzed the problem of detecting hate speech in Hindi-English code-mixed social media text and alongside also propose several classifiers for detecting hate speech based on sentence level, word level, and lexicon based features.

In this paper we propose a transformer based multilingual ensemble architecture for detecting various classes of hostility in Hindi language and also classify the hostile posts in fine grained classes such as Fake, Hate, Defamation and Offensive. We work with a novel dataset proposed by [3] and our proposed model outperforms current state-of-the-models by a large margin. we also carry out extensive architecture explorations as well as discussions regarding results obtained by our models.

3 DATASET

3.1 Description and Statistics

The dataset that we focus on in proposed and benchmarked by [3] which contains about 8200 hostile and non-hostile posts from various popular and most used social media like Facebook, Twitter, Whatsapp. All the posts are annotated as either *Non-hostile* or *Hostile*. For hostile posts, they are annotated with fine grained labels such as *Fake*, *Hate*, *Defamation* and *Offensive*.

All the data related to *Fake-news* were collected by referring to some of India's top most fact checking websites like BoomLive[2], Dainik Bhaskar[3] etc to identify topics of the fake news. Then the authors collected fake-news on the topics from Facebook, Twitter, Instagram etc. Similarly for *Hate Speech* collection, the authors focus on comments and posts targeting a specific race, religion or country. They also focus on comments of users who supported and commented in support of the hate speech. From a deeper observation it is revealed that a large section of the hate speech contains referral to Asian race, China or specific religions promoting communal violence. As most of the offensive posts contain swear words or vulgar phrases, the authors used list of top used swear words in Hindi language compiled by [17] and use Twitter API[4] to fetch related posts. For *Defamatory posts* the authors identify posts publicly shaming a specific race, culture, religion and collect such posts. A brief statistics of the dataset is given in the following Table 1. Some examples from the dataset are shown in Table 2. From the

Category	Hostile				Non Hostile
	Fake	Hate	Defame	Offense	
Train	1144	792	742	564	3050
Dev	160	103	110	77	435
Test	334	237	219	169	873
Total	1638	1132	1071	810	4358

Table 1: Dataset Statistics

examples we can see that a particular instance can have multiple hostile labels but a non-hostile post only can have only one label.

3.2 Data Preprocessing

Before training, we perform several steps of preprocessing on the dataset. The instances are posts from different social media and as a consequence it is uncleaned. It can be observed from Table 2, the dataset contains many *URLs* or *hyperlinks*, a lot of *special characters* such as "*@*","*#*","_", many emojies as well as emoticons such as "*:-)*","*:-(*","*:-P*" etc. We provide details of the preprocessing steps that we performed on the dataset as follows -

(1) **Non-Alphanumeric Character Removal:** In this step we remove all non-alphanumeric characters in the posts. This includes all the special characters such as - "*@*","_","*$*" etc.
(2) **Emoticon and Emoji Removal:** Next we remove all the emoticons such as "*:-)*","*:-(*","*XD*","*:-P*". We also remove every emoji present in the posts. For this task we use Unicode Character ranges from an online source[5] for removal of emoji. We also remove only the hashtag characters "" and not the entire hashtag with corresponding text. We also remove any pictographs, maps, dingbats and symbols that are not alphabet character.
(3) **Paragraph and Newline Removal:** We also remove newline characters and paragraph characters.

After these process we get clean posts and corresponding labels. We also employ some other preprocessing techniques which we use to train our proposed model to explore how our model performs on the different versions of preprocessed dataset. We perform the additional preprocessing steps and create different version of the cleaned dataset -

(1) **Stop-words Removal:** In this step we remove all the stop-words in the posts and for this purpose we use the stop-word list available at Data Mendeley[6].
(2) **Stemming:** To check our model's performance we stem the Hindi words using the Snowball Stemmer[7] library and create a version of dataset with stemmed words.
(3) **Named Entity Removal:** We remove the Named Entities (NEs) using Polyglot[8] library and create a version of the cleaned dataset with NEs removed.

In the Section 5 we describe the results that we obtained using these different versions of preprocessed dataset along with the main preprocessed dataset.

4 METHODOLOGY

In this section we describe our proposed model for Coarse-grained and fine grained evaluation of hostility detection posts in Hindi language.

4.1 Basic Architecture

The backbone of our proposed model is Multilingual Cased Bert Base Transformer Architecture with multi-head attention mechanism [9]. The core idea behind the transformer model is *multi headed self-attention* i.e the ability to attend to different positions of the input sequence to compute a representation of that sequence simultaneously by different heads. The transformer creates stacks of self-attention layers, thus it can learn long-range dependencies. For this reason we use Bidirectional Encoder Representations from Transformers (BERT) [9] as the backbone of our architecture which produces state-of-the-art results in many Natural Language Understanding (NLU) tasks including the General Language Understanding Evaluation (GLUE) benchmark [30]. There has been a trend of increasing the complexity of neural network architectures to achieve state-of-the-art performance in this era of deep learning. There are many works that could be found that made use of complex architectures [5, 8, 22] to capture the meaning of a sentence by embedding the sentence into a vector. However, a few recent language studies such as [9, 23] showed progress on many popular Natural Language Processing (NLP) tasks like Reading Comprehension on the widely used dataset like SQuAD [24] and RACE [19]. We specifically use multilingual variant of BERT [9] as the backbone of our architecture and add new layers on top of that. The architecture of our model is shown in the Figure 1 [9]. The detailed description of the architecture is as follows -

A. Tokenization Layer: At first we perform tokenization of the input sentence before giving it as an input to the model. We tokenize the sentence into constituent subwords. The significance

[2]https://hindi.boomlive.in/fake-news
[3]https://www.bhaskar.com/no-fake-news/
[4]https://developer.twitter.com/en/docs/twitter-api
[5]https://apps.timwhitlock.info/emoji/tables/unicode
[6]https://data.mendeley.com/datasets/bsr3frvvjc/1
[7]https://pypi.org/project/snowballstemmer/
[8]https://pypi.org/project/polyglot/
[9]Diagram Courtesy: https://cs.uwaterloo.ca/~jimmylin/BERT-diagrams-public.pptx

Sl.	Post	Labels
1	मेरे देश के हिन्दु बहुत निराले है। कुछ तो पक्के राम भक्त है और कुछ बाबर के साले है जय श्री राम	Hate, Offensive
2	JEE Exam center से निकले #Students को सुन बाकी छात्रों के साथ Parents के चेहरे पर मुस्कान आ जाएगी https://t.co/TQ7nfIv0I0 https://t.co/gGCDYYEz6E	Non-Hostile
3	कांग्रेस मूल की कंगना रनौत बिहार चुनाव में भाजपा का प्रचार करेंगी! #NATIONALNEWS	Fake
4	@SalmanNizami_ राहुल गांधी – Maa मैं अगले 4 साल क्या करूंगा सोनिया गांधी – बेटा TV रिचार्ज कर दिया है बैठकर छोटा भीम देख.	Defamation

Table 2: Example of Dataset

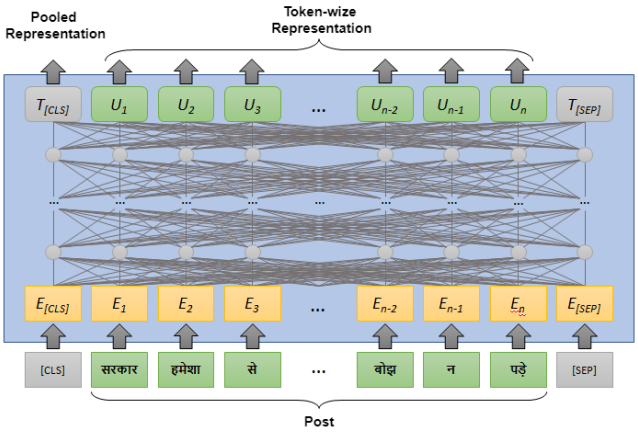


Figure 1: Basic Multilingual BERT Model

of tokenizing a sentence into subwords is to handle the out-of-vocabulary (OOV) word problems. Hence, the sentence will be tokenized as: $I = \{x_1, x_2, x_3....x_N\}$ where x_i is the i th token of the tokenized sentence. There are two special tokens used in BERT [9] i.e special *[SEP]* token which is used to denote ending of a token sequence which is appended to the end of the sequence and another special token *[CLS]*, used at the front of the sequence which corresponds to the classification token containing the learned compact representation useful for classification task and to be used in the subsequent task specific layers. So mathematically, the processed token sequence before giving it as an input to the next layer is expressed as follows:

$$I = [CLS]x_1, x_2, x_3....x_N[SEP] \tag{1}$$

BERT also uses *Positional Embeddings* to determine orders of constituent words in a given input sentence. *Positional Embeddings* thus play a crucial role where they tell the model that a word have a different meaning/syntactic function depending on its position. Either one sentence or two sentence can be given as an input to BERT. To differentiate the two sentences, it also uses *Segment Embeddings* which are simply two embeddings (for segments **A** and **B**) and it adds this *Segment Embeddings* to the *Token Embeddings* and *Positional Embeddings* before feeding them into the input layer. But when only one sentence is given as an input these *Segment Embeddings* are not used (as in our case).

B. Multilingual cased BERT Base Layer: The Multilingual cased BERT Base[10] layer is used as the backbone of this model. It is a 12 layered transformer based architecture with 12 multi-headed self-attention heads, thereby creating a layer-wise dense representation of the input sequence, which is used to learn attention [29] based deep contextual embeddings of the input text as well as focuses on long range dependencies. We feed the output of Tokenization Layer i.e I as obtained from the Equation 1 to this layer. After the input is fed into the model, we take the final hidden state (i.e., the output of the final layer Transformer) for the first token, which correspond to the special *[CLS]* token embedding. We obtain this context vector (denoted as C) with a dimension of $1 \times H$ (in case of Multilingual BERT base, $H = 768$). This is also called the hidden representation vector. We also employ 20% Dropout [28] in last layer of Multilingual cased BERT Base.

C. Feed Forward Layer: We feed the input of the previous layer or hidden representation vector i.e C as an input to the Feed Forward Layer *(FFN)* with $K \times H$ dimensional weight, denoted by W, where K is the number of labels. Here $K = 2$ as labels are *postitve* and *negative*. We denote this intermediate representation as F with dimension $1 \times K$.

$$F = C \cdot W^T + b \tag{2}$$

where $\cdot$ denotes the dot product between the weight matrix W and the context vector C and b is a bias term. Afterwards we also introduce ReLU [21] nonlinearity to the intermediate representation as described in Equation 3 and obtain the nonlinear intermediate representation denoted as Q with dimension $1 \times K$.

$$Q = ReLU(F) \tag{3}$$

D. Classification Layer: The output of the feed forward layer (*FFN*) i.e Q is fed into the final classification layer. The label probabilities are computed with a standard *softmax* function, as equation P has a dimension of $1 \times K$.

$$P = Softmax(Q) \tag{4}$$

4.1.1 Transformer Based Model for Coarse Grained Evaluation: For Coarse-grained evaluation we employ the multilingual cased Bert Base model or mBERT model [9] and XLM-Roberta [7] which is

[10]https://github.com/google-research/bert/blob/master/multilingual.md

another variation of the mBERT model that harnesses unsupervised cross-lingual representations from different languages. The diagram of the model that were used for conducting coarse grained evaluation is given in the Figure 2 -

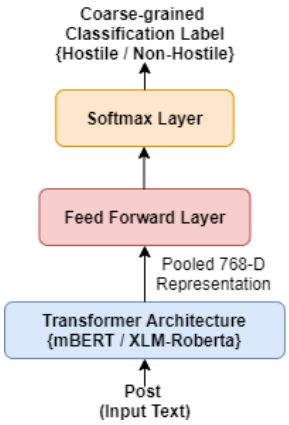


Figure 2: Block Diagram of Transformer based Coarse-grained Evaluation Model

In both mBERT and XLM-Roberta we use pooled representation of post from the last layer of the transformer based model where the pooled representation for each post is a 768 dimensional vector.

4.1.2 Ensemble Model for Coarse Grained Evaluation: To improve the performance of Coarse-grained classification of our individual models i.e mBERT and XLM-Roberta, we ensemble the two models. We get outputs from the two models and concatenate them. Finally we feed this concatenated representation through a MLP (Multi-layered Perceptron) with 3 layers and finally we use softmax classification layer to get the final output class labels for Coarse-grained evaluation. The diagram corresponding to the ensemble approach is given in the Figure 3 -

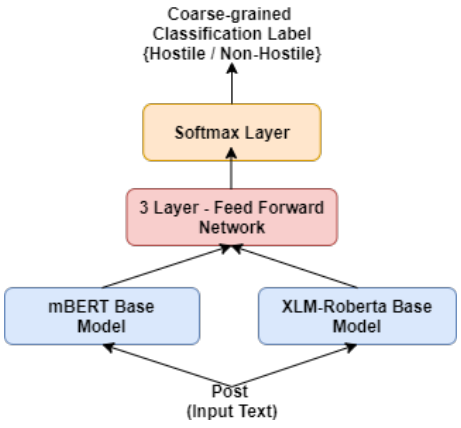


Figure 3: Block Diagram of Ensemble Model for Coarse-grained Evaluation

4.1.3 Recurrent Models for Coarse Grained Evaluation: We also explore Long Short Term Memory (LSTM) [15] and Gated Recurrent Unit (GRU) [6] based neural network models and evaluate their performance on Coarse-grained evaluation of posts. We use tokenwize representation of mBERT model which is a $m \times 768$ dimensional vector where m = maximum length of post, and feed it to the Bidirectional version of LSTM or GRU layer (i.e BiLSTM and BiGRU) layers and feed the output representation from the layer to a MLP (Multi-layered Perceptron) with 3 layers and finally we use softmax classification layer to get the final output class labels for Coarse-grained evaluation. The corresponding block diagram of the model is given in Figure 4 -

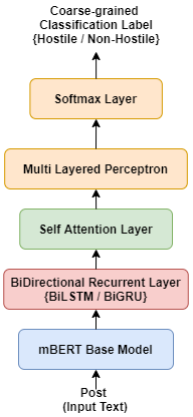


Figure 4: Block Diagram of Recurrent Neural Network based Model for Coarse-grained Evaluation

4.1.4 Machine Learning Models for Coarse Grained Evaluation: The block diagram of the machine learning models is given in the following Figure 5 -

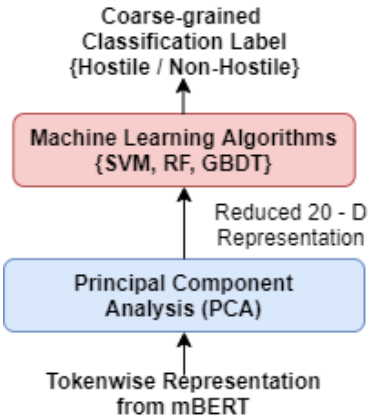


Figure 5: Block Diagram of Machine Learning Based Models for Coarse-grained Evaluation

For exploration purpose we use the tokenwise representation of mBERT model where each post is represented by a 2 dimensional

vector of size $m \times 768$. Here m = maximum length of post. We take the tokenwise representation and apply Principal Component Analysis (PCA) such that the reduced data still explains 99.85% variance of the entire representation. We found that particular **20 features** are important and sufficient for explaining 99.85% variance of the representation. Lastly we feed this reduced representation to a wide range of traditional machine learning algorithms such as Support Vector Machine (SVM), Random Forest (RF) and Gradient Boosted Decision Tree (GBDT) and measure the performance of such models. The detailed discussion of results is in Section 5.

4.1.5 Transformer Based Multi-label Model for Fine Grained Evaluation. The diagram of the multi-label architecture for fine-grained evaluation is as given in the Figure 6 -

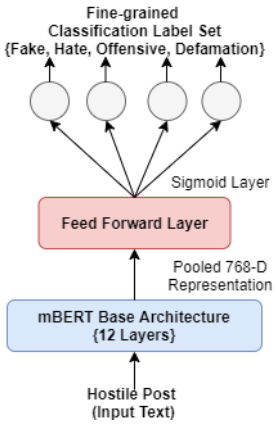


Figure 6: Block Diagram of Multi-label Based Transformer Model for Fine-grained Evaluation

For the task of Fine-grained evaluation of hostile posts we use mBERT architecture and use the 1×768 dimensional pooled representation say,H corresponding to $[CLS]$ token and feed it to a 3 layered Multi Layered Perceptron (MLP). Lastly we take the output representation from MLP say,H_1 and feed it to a Sigmoid layer with 4 independent Sigmoid neurons. The output of the Sigmoid layer is a 1×4 dimensional vector P with **independent probabilities** corresponding to the class of the input being *Fake*, *Hate*, *Offensive* or *Defamation* as follows -

$$P_{1\times4} = Sigmoid\_Layer(H_1) \tag{5}$$

From equation 5 it is clear that the 4 probabilities are independent and unaffected by each other. Hence, a particular text can be classified as multiple labels such i.e *Hate, Offensive* simultaneously.
While training the model, we took only the hostile instances (i.e instances annotated as *Fake*, *Hate*, *Offensive* or *Defamation*). Which means the *Non-hostile* instances were not given as training dataset to the model.

4.1.6 Transformer Based One vs Rest Approach for Fine Grained Evaluation. For exploration purpose we also try another approach for Fine-grained evaluation of hostile classes. We use four parallel mBERT model. For each of these models we input hostile sentences labelled as *Fake* or *Non-Fake*, *Hate* or *Non-Hate* etc. That means we change the multi-label classification problem of hostile classes into four independent binary classification problem. For each model, we take the 768 dimensional pooled representation from mBERT model and feed them to 3 layered Multi Layered Perceptron (MLP). Then we take the output representation from the MLP layer and pass them to a Sigmoid layer to get the final classification label. The diagram of the architecture is given in the Figure 7 -

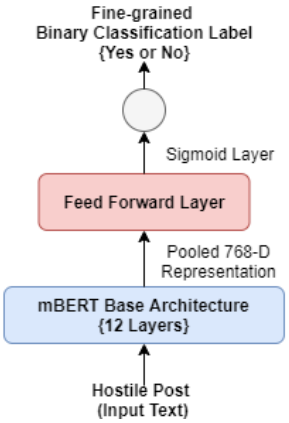


Figure 7: Block Diagram of One vs Rest Based Transformer Model for Fine-grained Evaluation

The main difference of this architecture with the architecture described in Section 4.1.5 is that, in this architecture we train four different models with four binary classification datasets which gives output as *Yes* or *No*. On the other hand, we trained the previous model with single dataset labelled as *Fake*, *Hate*, *Offensive* or *Defamation*.

4.2 Experimental Setup

In this section we briefly discuss the hyper-parameters and experimental setups that have been used to conduct the experiments.
For all the transformer based models we use batch size of 28, and run the trainings for 10 iterations. We set the maximum length of input sequence to 128. We also use a *Warmup* proportion of 0.15 where *Warmup* is a period of time where the learning rate is small and gradually increases which usually helps training. For mBERT and XLM-Roberta models we use initial learning rate of $2e-5$ and $5e-5$ respectively. We use GeLU [14] as hidden activation in each layer of the transformer architectures and use 10% Dropout [28] in the last layer of each model. All other parameters of mBERT[11] and XLM-Roberta[12] are initialized to their original values. We run the models to finetune all the parameters for our task end to end.
For all the machine learning models we use grid search[13] to tune all the hyper-parameters. For Support Vector Machine (SVM) we use Gaussian kernel (rbf kernel), for Random Forest we set number of estimators to 80.

[11]https://github.com/google-research/bert/blob/master/multilingual.md
[12]https://github.com/facebookresearch/(fairseq-py,pytext,xlm)
[13]https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html

5 RESULTS AND DISCUSSION

In this section we present detailed discussion on the results obtained by our machine learning and deep learning architectures.

5.1 Coarse-grained Evaluation

At first we feed the preprocessed dataset to mBERT model and XLM-Roberta model separately and then finetune the parameters of the models end to end using the annotated labels of the dataset. We use a 3 layered MLP and a Softmax layer as our classifier section of the architectures (see Section 4.1.1). The results obtained by our models are shown in the Table 3.

Architecture	Accuracy (%)
mBERT	91.63
XLM-Roberta	89.76
Ensemble (Best Model)	**92.60**

Table 3: Coarse-grained Evaluation Results using Transformer Based Architectures

We obtain **91.63**% and **89.76**% accuracy score on mBERT and XLM-Roberta model. For performance improvement we consider the concatenation of the output of the two models (viz. mBERT and XLM-Roberta) and then feed it to a MLP with Softmax layer on top as classifier. This ensemble model (see Section 4.1.2) obtains an accuracy score of **92.60**% which is a absolute 0.97 score improvement than the individual models. We identify the ensemble model as our best model for coarse-grained evaluation. The confusion matrix of predictions by ensemble model for validation data is given in the Figure 8.

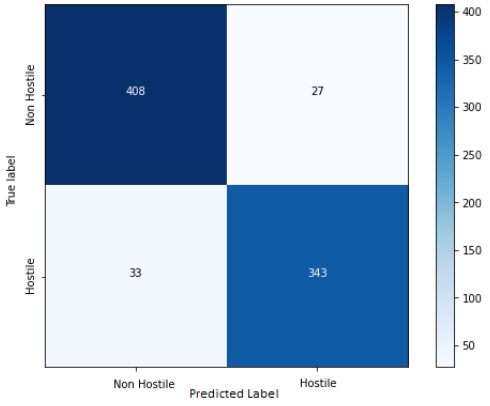


Figure 8: Confusion Matrix for Coarse-grained Evaluation for Ensemble Approach

We also run coarse-grained classification on datasets preprocessed with all the words stemmed, Named Entities removed and stop words removed. We observed that our model performed poorly with this dataset and hence we conclude that Named Entities are very crucial information regarding hostility detection. Thus for later experiments we chose to work with cleaned dataset with all the words un-stemmed and no Named Entities removed.

We use the finetuned $m \times 768$ dimensional **tokenwize representation** of the data obtained from mBERT and explore recurrent architectures such as Bidirectional LSTM (BiLSTM), Bidirectional GRU (BiGRU) with Self-Attention [29] mechanism where m = maximum sequence length of post in the datset. The results of the different experiments is given in the Table 4.

Architecture	Specifications	Accuracy Score (%)
Bi-LSTM	2 Recurrent Layers	92.11
	Self Attention Layer	91.49
Bi-GRU	2 Recurrent Layers	92.05
	Self Attention Layer	**92.36**

Table 4: Coarse-grained Evaluation Results using Recurrent Neural Network Architectures

In both of the cases it is shown that self-attention mechanism performs better than stacking multiple layers of recurrent units. In BiLSTM model the self-attention mechanism improves performance by 0.38% and In BiGRU model the self-attention mechanism improves performance by 0.21%.

For further exploration of architectures for coarse-grained evaluation we use the 768 dimensional **pooled representation** and use it as features to a variety of classical machine learning algorithms. Due to very high dimensionality of the representations we also employ Principal Component Analysis (PCA) to reduce the data where we found out that only **20 dimensions** are sufficient to explain 99.85% variance of the entire representation. Our first genuine choice for classification is the Support Vector Machine in which we use the Gaussian Kernel (also known as RBF Kernel) which theoretically transports the input dimension to an infinite dimension and finds a linear separation boundary in the higher dimension. We also employ Random Forest (RF), Gradient Boosted Decision Trees (GBDT) and XGBoost algorithm on the original representation as well as on the reduced representation. The results of the experiments are shown in the following Table 5. We achieve a high score of **91.98**% for XGBoost algorithm with PCA among all the machine learning algorithms.

Algorithm	PCA	Accuracy Score (%)
SVM	Yes	91.86
	No	91.49
RF	Yes	91.61
	No	91.46
GBDT	Yes	91.63
	No	91.46
XGBoost	Yes	**91.98**
	No	91.62

Table 5: Coarse-grained Evaluation Results using Machine Learning Algorithms

5.2 Fine-grained Evaluation

Fine grained evaluation is the evaluation and prediction of the *Hostile* posts and categorizing them to a finer classes such i.e *Fake*, *Hate*, *Offensive* or *Defamation*. A particular post can be classified as multiple labels (for example - a post can be both *Hate* as well as *Offensive*). We explore different transformer based architectures and different classifier models for achieving better performance in Fine-grained evaluation task. For this purpose we use two different models as described in Section 4.1.5 and Section 4.1.6. We achieve much better performance in the One vs Rest approach where we transform the multi-label problem statement into four different binary classification problems. The results of our best performing model is given in the Table 6.

Label	Accuracy (%)	F1 Score
Fake	81.38	81.14
Hate	69.68	69.59
Defamation	73.13	73.01
Offensive	77.39	75.29

Table 6: Fine-grained Evaluation Results using Transformer Based Architecture (One vs Rest Approach)

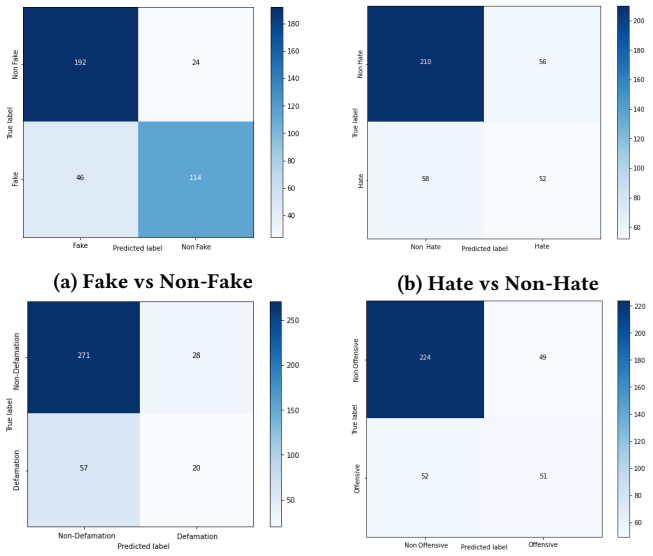
We argue that in comparison to our multi-label classification transformer model (described in Section 4.1.5), the models in One vs Rest approach (described in Section 4.1.6) performs significantly better because here we train four *different* models for classification of the corresponding labels separately. Hence features that are important and contribute more towards a specific class is not suppressed by features that are important for other class. It may be also the case that, some features that positively contribute towards classification of a particular class negatively contribute towards the classification of other class. In that case proper importance of feature will not be realized by the model. Hence by segregating the classes and using four different models for independently classifying the labels, we eliminate all such possibilities of features not being given proper importance. In the Figure 9 we show the confusion matrix of our best performing fine-grained evaluation model.

5.3 Comparison with Baseline

In this section we compare our proposed model performance with baseline [3] scores reported. All of the baseline models i.e Logistic Regression (LR), Support vector Machine (SVM), Random Forest (RF) and Multi Layered Perceptron (MLP) uses mBERT based pooled representation as input feature. In the Table 7 we show the results of our models with the baseline models.

From the Table 7, we observe that our propose models for Coarse-grained and Fine-grained evaluation performs significantly better in every aspect. We bold-font the best results reported by [3] and our results. It is seen that our model outperforms the best baseline model by a significant margin of 8.49% for Coarse-grained evaluation task and for Fine-grained evaluation task, our model achieves 33.65%, 1.44%, 33.31% and 29.44% performance increase for classification of *Fake*, *Hate*, *Offensive* and *Defamation* posts respectively.

We conclude that our proposed models (Section 4.1.2 and Section 4.1.6) outperforms the baseline models by a wide margin and establishes a new state-of-the-art architecture for Hostility Prediction in Hindi language.



(a) Fake vs Non-Fake (b) Hate vs Non-Hate

(c) Defame vs Non-Defame (d) Offense vs Non-Offense

Figure 9: Confusion Matrix for Fine-grained Evaluation for One vs Rest Approach

Model	Coarse Grained	Fine Grained			
		Fake	Hate	Offense	Defame
LR	83.98	44.27	**68.15**	38.76	36.27
SVM	**84.11**	**47.49**	66.44	**41.98**	**43.57**
RF	79.79	6.83	53.43	7.01	2.56
MLP	83.45	34.82	66.03	40.69	29.41
Ours (Dev)	**92.60**	**81.14**	**69.59**	**75.29**	**73.01**
Ours (Test)	**90.53**	**63.30**	47.34	55.83	39.37

Table 7: Comparison of Proposed Models with Baseline for Coarse-grained and Fine-grained evaluation (F1 score)

5.4 Additional Discussion and Analysis

In this section we present a brief study of the predictions by our best performing models and discuss some important facts that arose in our observation.

Using our model's predicted classes we extracted the top 10 frequent Named Entities occurring in the dataset and plotted them in Figure 10. We also tabulated some of the predicted *Fake*, *Hate*, *Offensive* and *Defamation* posts in the Table 8. From Figure 10 and Table 8 we present some observations about the predictions of our models.

(1) From the frequency plots in the Figure 10 we observe that the words "**India**" and "**Modi**" are the top frequent words in posts classified as *Fake*, *Hate*, *Offensive* and *Defamation*.

Sl.	Post	Predicted Label
1.	हमारे हिन्दू जाट भाईओ पर बोला गहलोत देख लो और वोट दो जाट भाईओ ये साले किसी के सगे नही है	Offensive
2.	दाऊद इब्राहिम और उसकी पत्नि की कोरोना संक्रमण से मौत	Fake
3.	मुंबई: खुले मेनहोल्स की शक्ल में मौजूद हैं BMC की लापरवाही के नमूने बन रहे हैं जानलेवा रिपोर्ट jitendradixit	Hate
4.	ReportForSSR शिवसेना नेता संजय राउत के हाथों से सबकुछ निकल गया है क्या भाजपा महाराष्ट्र प्रवक्ता श्वेता शालिनी ने राजनीतिक विश्लेषक अजय अरोड़ा से पूछा देखिए पूछता है भारत अर्नब के साथ रिपब्लिक भारत पर LIVE	Defamation

Table 8: Examples of Predicted Posts

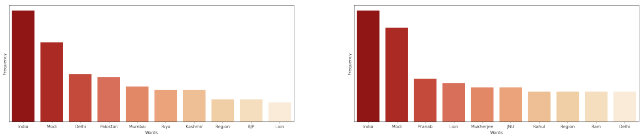
These give us clear indication that most of the *Hostile* sentences are regarding politics as political NEs like "**Modi**", "**Rahul**" are predominantly present in the Hostile posts.

(2) Specific words like "**Congress**" are associated with the classes *Offensive* and *Defamation*, whereas the word "**Pakistan**", "**Delhi**" are associated with *Fake* posts and word "**JNU**" is associated with *Hate* speech.

(3) We also observe that current events (such as Corona Virus outbreak, death of Bollywood actor, JNU attack etc.) have a very important role in deciding posts to be detected as Hostile. Example - association of word "**China**" with *Offensive* and *Defamation* posts, association of the word "**Riya**" with *Defamation* and *Offensive*.

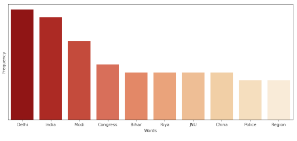
(4) By examining further we also observe that the words like "**RSS**", "**Ram**", "**Kashmir**" and "**Region**" etc increases the probability of a model to be classified as *Offensive* and *Hate*.

(5) From the Table 8 and from studying other examples we also observe that the probability of a sentence being classified as *Offensive* increases very sharply if the post contains a vulgar word for example "साले", "कुत्ते".

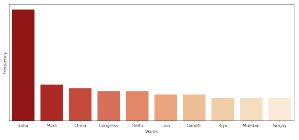


(a) Top NEs in Fake Posts

(b) Top NEs in Hate Posts



(c) Top NEs in Defamation Posts



(d) Top NEs in Offensive Posts

Figure 10: Top Frequent Named Entities in different categories of Predicted Posts

6 CONCLUSION AND FUTURE WORKS

In this paper we tackle the important and relevant problem of detection of hostility Hindi posts. Due to rise of users in social media, it is our duty to keep social media friendly and hassle-free for the people who are using it. For this purpose the work towards hostility detection must be addressed with utmost care and importance. The guidelines for preventing various forms of hostility established by social media sites helps people to not cross the finer line of freedom of speech. But due to complex structures of posts and different emoji and emoticons it becomes really challenging to detect different forms of hostility. To address these problems and to mitigate the challenges of hostility detection in Hindi language we propose a multilingual transformer based ensemble architecture which outperforms the current state-of-the-art architectures with significant margin. Furthermore we present detailed analysis of performance of various recurrent neural network and machine learning architectures and also present a brief linguistic analysis of what is causing our models to response to a particular class than other classes.

This field of research is very relevant and less developed in languages other than English and it needs extensive exploration. In future we would like to extend our current work on the challenge of hostility detection in general in the following ways -

(1) Extension of the current task to other low resource languages other than Hindi such as Vietnamese, Indonesian, Telugu, Tamil, Swahili etc. As this problem is language independent hence it must be addressed for other resource-poor languages as well.

(2) The absence of annotated dataset for hostility detection in resource poor language is restricting the growth of this research field. To address this problem we would like to propose annotated dataset for hostility detection is various low resource languages.

(3) The effect of multimodalilty in Hostility Detection is a less explored region. We would like to add multimodal features such as acoustic that consists of pitch, voice quality and video frame which captures important visual information i.e facial expressions (viz. disgust, agony, anger, happiness), gesture and posture. These would help us to detect hostility in not only audio or video posts in online social media but these would also help us in tackling the problem of hostility detection in real-time scenarios such as interviews or conversations etc.

(4) The effect of linguistic features causing neural network architectures to detect hostility in different posts is also a very

interesting research topic. It may also be the case that existence of some language independent latent (hidden) features causes the detection of hostility. The crosslingual studies such as training architectures using instances in a particular set of languages and testing the trained model with different set of instances in some different set of languages which was not present in the training set may help us detecting the important language independent latent features.

ACKNOWLEDGMENTS

This research work is executed as final project of Information Retrieval Course (CS6370) offered at Indian Institute of Technology Hyderabad (IIT Hyderabad) under the guidance of Dr. Maunendra Sankar Desarkar, Assistant Professor at the Computer Science and Engineering Department, IIT Hyderabad and under the mentorship of Kaushal Kumar Maurya, Research Scholar at the Computer Science and Engineering Department, IIT Hyderabad. The authors would like to thank the guide and the mentor for their constant support and ideas.

REFERENCES

[1] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep Learning for Hate Speech Detection in Tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion* (Perth, Australia) *(WWW '17 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 759–760. https://doi.org/10.1145/3041021.3054223

[2] Pinkesh Badjatiya, Shashank Gupta, Manish Gupta, and Vasudeva Varma. 2017. Deep Learning for Hate Speech Detection in Tweets. In *Proceedings of the 26th International Conference on World Wide Web Companion* (Perth, Australia) *(WWW '17 Companion)*. International World Wide Web Conferences Steering Committee, Republic and Canton of Geneva, CHE, 759–760. https://doi.org/10.1145/3041021.3054223

[3] Mohit Bhardwaj, Md. Shad Akhtar, Asif Ekbal, Amitava Das, and Tanmoy Chakraborty. 2020. Hostility Detection Dataset in Hindi. *ArXiv* abs/2011.03588 (2020).

[4] Aditya Bohra, Deepanshu Vijay, Vinay Singh, Syed Sarfaraz Akhtar, and Manish Shrivastava. 2018. A Dataset of Hindi-English Code-Mixed Social Media Text for Hate Speech Detection. In *Proceedings of the Second Workshop on Computational Modeling of People's Opinions, Personality, and Emotions in Social Media*. Association for Computational Linguistics, New Orleans, Louisiana, USA, 36–41. https://doi.org/10.18653/v1/W18-1105

[5] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Yun-Hsuan Sung, Brian Strope, and Ray Kurzweil. 2018. Universal Sentence Encoder. *CoRR* abs/1803.11175 (2018). arXiv:1803.11175 http://arxiv.org/abs/1803.11175

[6] J. Chung, Çaglar Gülçehre, Kyunghyun Cho, and Yoshua Bengio. 2014. Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling. *ArXiv* abs/1412.3555 (2014).

[7] Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, E. Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised Cross-lingual Representation Learning at Scale. *ArXiv* abs/1911.02116 (2020).

[8] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, 670–680. https://doi.org/10.18653/v1/D17-1070

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. *Proceedings of the 2019 Conference of the North* (2019). https://doi.org/10.18653/v1/n19-1423

[10] Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate Speech Detection with Comment Embeddings. In *Proceedings of the 24th International Conference on World Wide Web* (Florence, Italy) *(WWW '15 Companion)*. Association for Computing Machinery, New York, NY, USA, 29–30. https://doi.org/10.1145/2740908.2742760

[11] Nemanja Djuric, Jing Zhou, Robin Morris, Mihajlo Grbovic, Vladan Radosavljevic, and Narayan Bhamidipati. 2015. Hate Speech Detection with Comment Embeddings. In *Proceedings of the 24th International Conference on World Wide Web* (Florence, Italy) *(WWW '15 Companion)*. Association for Computing Machinery, New York, NY, USA, 29–30. https://doi.org/10.1145/2740908.2742760

[12] Njagi Dennis Gitari, Zhang Zuping, Hanyurwimfura Damien, and Jun Long. 2015. A lexicon-based approach for hate speech detection. *International Journal of Multimedia and Ubiquitous Engineering* 10, 4 (2015), 215–230.

[13] Bushr Haddad, Zoher Orabe, Anas Al-Abood, and Nada Ghneim. 2020. Arabic Offensive Language Detection with Attention-based Deep Neural Networks. In *Proceedings of the 4th Workshop on Open-Source Arabic Corpora and Processing Tools, with a Shared Task on Offensive Language Detection*. European Language Resource Association, Marseille, France, 76–81. https://www.aclweb.org/anthology/2020.osact-1.12

[14] Dan Hendrycks and Kevin Gimpel. 2016. Gaussian Error Linear Units (GELUs). *arXiv: Learning* (2016).

[15] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Comput.* 9, 8 (Nov. 1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

[16] Md Zobaer Hossain, Md Ashraful Rahman, Md Saiful Islam, and Sudipta Kar. 2020. BanFakeNews: A Dataset for Detecting Fake News in Bangla. In *Proceedings of the 12th Language Resources and Evaluation Conference*. European Language Resources Association, Marseille, France, 2862–2871. https://www.aclweb.org/anthology/2020.lrec-1.349

[17] Vikas Jha, Hrudya Poroli, Vinu N, Vishnu Vijayan, and Prabaharan P. 2020. DHOT-Repository and Classification of Offensive Tweets in the Hindi Language. *Procedia Computer Science* 171 (01 2020), 2324–2333. https://doi.org/10.1016/j.procs.2020.04.252

[18] Hamid Karimi, Proteek Roy, Sari Saba-Sadiya, and Jiliang Tang. 2018. Multi-source multi-class fake news detection. In *Proceedings of the 27th International Conference on Computational Linguistics*. 1546–1557.

[19] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-Scale ReAding Comprehension Dataset from Examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Copenhagen, Denmark, 785–794. https://doi.org/10.18653/v1/D17-1082

[20] Puneet Mathur, Rajiv Shah, Ramit Sawhney, and Debanjan Mahata. 2018. Detecting Offensive Tweets in Hindi-English Code-Switched Language. In *Proceedings of the Sixth International Workshop on Natural Language Processing for Social Media*. Association for Computational Linguistics, Melbourne, Australia, 18–26. https://doi.org/10.18653/v1/W18-3504

[21] Vinod Nair and Geoffrey E. Hinton. 2010. Rectified Linear Units Improve Restricted Boltzmann Machines. In *Proceedings of the 27th International Conference on International Conference on Machine Learning* (Haifa, Israel) *(ICML'10)*. Omnipress, Madison, WI, USA, 807–814.

[22] Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised Learning of Sentence Embeddings Using Compositional n-Gram Features. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 528–540. https://doi.org/10.18653/v1/N18-1049

[23] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving Language Understanding by Generative Pre-Training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf* (2018).

[24] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, 2383–2392. https://doi.org/10.18653/v1/D16-1264

[25] Natali Ruchansky, Sungyong Seo, and Yan Liu. 2017. CSI: A Hybrid Deep Model for Fake News Detection. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management* (Singapore, Singapore) *(CIKM '17)*. Association for Computing Machinery, New York, NY, USA, 797–806. https://doi.org/10.1145/3132847.3132877

[26] Tanik Saikh, Arkadipta De, Asif Ekbal, and Pushpak Bhattacharyya. 2020. A Deep Learning Approach for Automatic Detection of Fake News. *CoRR* abs/2005.04938 (2020). arXiv:2005.04938 https://arxiv.org/abs/2005.04938

[27] Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. 2017. Fake news detection on social media: A data mining perspective. *ACM SIGKDD explorations newsletter* 19, 1 (2017), 22–36.

[28] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A Simple Way to Prevent Neural Networks from Overfitting. *J. Mach. Learn. Res.* 15, 1 (Jan. 2014), 1929–1958.

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *NIPS*.

[30] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. In *7th International Conference on*

Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net. https://openreview.net/forum?id=rJ4km2R5t7

[31] Zeerak Waseem, T. Davidson, Dana Warmsley, and Ingmar Weber. 2017. Understanding Abuse: A Typology of Abusive Language Detection Subtasks. *ArXiv* abs/1705.09899 (2017).

[32] Zeerak Waseem and Dirk Hovy. 2016. Hateful Symbols or Hateful People? Predictive Features for Hate Speech Detection on Twitter. In *Proceedings of the NAACL Student Research Workshop*. Association for Computational Linguistics, San Diego, California, 88–93. https://doi.org/10.18653/v1/N16-2013

[33] Zeerak Waseem and Dirk Hovy. 2016. Hateful symbols or hateful people? predictive features for hate speech detection on twitter. In *Proceedings of the NAACL student research workshop*. 88–93.

[34] Xinyi Zhou, Reza Zafarani, Kai Shu, and Huan Liu. 2019. Fake News: Fundamental Theories, Detection Strategies and Challenges. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining* (Melbourne VIC, Australia) *(WSDM '19)*. Association for Computing Machinery, New York, NY, USA, 836–837. https://doi.org/10.1145/3289600.3291382