Demographical Based Sentiment Analysis for Detection of Hate Speech Tweets for Low Resource Language

KAMAL SAFDAR, Department of Computer Software Engineering, National University of Sciences and Technology (NUST), Islamabad-44000, Pakistan
SHIBLI NISAR, Department of Electrical Engineering, National University of Sciences and Technology (NUST), Islamabad-44000, Pakistan
WASEEM IQBAL[*], Department of Information Security, National University of Sciences and Technology (NUST), Islamabad-44000, Pakistan
AWAIS AHMAD, Information Systems Department, College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), Saudi Arabia
YAWAR ABBAS BANGASH[*], Department of Computer Software Engineering, National University of Sciences and Technology (NUST), Islamabad-44000, Pakistan

Advancement in IT and communication technology provides the opportunity for social media users to communicate their ideas and thoughts across the globe within no time as well big data promulgated in a result of the communication process itself has immense challenges. Recently, the provision of freedom of speech has witnessed immense promulgation of offensive and hate speech content on the internet aimed the basic human rights violation. The detection of abusive content on social media for rich resource language has become a hot area for researchers in the recent past. However, low-resource languages are underprivileged due to the non-availability of large corpus and its complexity to understand. The proposed methodology mainly has two parts. One is to detect abusive content and the other is to have a demographical analysis of the Indigenously developed dataset. The process starts with the development of a unique unlabeled Urdu dataset of 0.2 M from Twitter through a web scrapper tool named snscraper. The dataset is collected against the 36 districts of Punjab from Pakistan and from the duration 2018- Apr 2022. The dataset is labeled into three target classes Neutral, Offensive, and Hate Speech. After data cleaning, the feature extraction process is achieved with the help of traditional techniques such as Bow and tf-idf with the combination of word and char n-gram and word embedding word2Vec. The dataset is trained on both machine learning algorithms SVM and Logistic regression and deep learning techniques Long Short Term Memory (LSTM) and Convolutional Neural Networks (CNN). The best F score achieved through LSTM on this dataset is 64 and accuracy is 93 through CNN

[*] Corresponding Author: Waseem Iqbal (waseem.iqbal@mcs.edu.pk)
All authors contributed equally to this research.

Authors' addresses: Kamal Safdar, kamal_csguru@yahoo.com, Department of Computer Software Engineering, National University of Sciences and Technology (NUST), Islamabad-44000, , Islamabad, Pakistan, 44000; Shibli Nisar, shiblinisar@mcs.edu.pk, Department of Electrical Engineering, National University of Sciences and Technology (NUST), Islamabad-44000, , Islamabad, Pakistan, 44000; Waseem Iqbal, waseem.iqbal@mcs.edu.pk, Department of Information Security, National University of Sciences and Technology (NUST), Islamabad-44000, , Islamabad, Pakistan, 44000; Awais Ahmad, aahmad.marwat@gmail.com, Information Systems Department, College of Computer and Information Sciences, Imam Mohammad Ibn Saud Islamic University (IMSIU), , Riyadh, Saudi Arabia; Yawar Abbas Bangash, yawar@mcs.edu.pk, Department of Computer Software Engineering, National University of Sciences and Technology (NUST), Islamabad-44000, , Islamabad, , Pakistan, 44000.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2023 Copyright held by the owner/author(s). Publication rights licensed to ACM.

2375-4699/2023/8-ART $15.00
https://doi.org/10.1145/3616867

Just Accepted

algorithms. A Choropleth map is used for visualization of the dataset distributed among 36 districts of Punjab and a time series plot for time analysis covers five years duration from 2018-Apr to 22.

CCS Concepts: • **Do Not Use This Code → Generate the Correct Terms for Your Paper**; *Generate the Correct Terms for Your Paper*; Generate the Correct Terms for Your Paper; Generate the Correct Terms for Your Paper.

Additional Key Words and Phrases: snscrapper, CNN , Word Embedding, LSTM, Choropleth map

1 INTRODUCTION

The advancement of technologies and availability of rich social media platforms like Facebook, Twitter, YouTube, and Instagram allow users to connect and communicate to share their ideas, and thoughts with relatives and friends in no time with the purpose to bring the social media community under one umbrella. In old days, different communication media is used to deliver messages like smoke signals, telegraphs, carrier pigeons, and balloon mail. The main problem with using these modes is the delay factor, the message received with delay losses its importance. The dramatic rise in technologies like high-speed networks i.e., 5G supports the social media platform to share ideas within no time with the provision of freedom of speech. It permits every individual to extend hateful ideologies among the community. The use of abusive language in social media leads to hate crimes. Given the collaborative nature of social media, the detection of hate speech content and hate speech crime has become effortless. The traditional law enforcement institution somehow established laws against hate speech content to reduce the spread of hate speech crimes, but the problem is still there. It affects the mental and emotional health of the target group, i.e., Shia, Sunny, Political, Ethnic group, e.t.c.. The life cycle of hate speech content comprises of four steps defined by Chatty and Alathur 2018, First step, hate speech remains high on social media, then it's gradually reduce after a few days in the second step. After some days the hate speech remains zero and then in the fourth stage, the hate speech again returns subject to content type, location, target class, etc. According to a recent online hate speech report of Pakistan, the most of hate speech promulgated is religiously and culturally motivated. 42 % from religious, 16% Sex/gender / sexual orientation, 22 % from race/ ethnicity, and 23 % from nationality. The Root cause of hate speech promulgation on social media is the lack of awareness of hate speech and improper proper legislation and implementation by law enforcement agencies. Social media platforms like Facebook, Twitter, etc. Somehow formulate and implement AI-based hate speech detection Algorithms that automatically detect and remove the contents from their platform but there is limitation subject to diversity of content, language, and location. Legislation from different countries including the USA, Australia, Denmark, and the UK to protect their people from harassment and hate speech content. EU code of conduct was launched in 2016 and was implemented by four internet social platforms (Facebook, Twitter, YouTube, and Microsoft) with the purpose to control and stop hate speech content on the internet. Pakistan has formulated similar policies and laws to encounter hate speech. Pakistan Penal Code (PPC) states any violation against race, ethnicity, community, religious group, and any cast will result in five years of imprisonment. The anti-terrorism act 1997 [5], declares the individual guilty if he or she is found with threatening or abusive language or words. Every citizen have a right to freedom of speech with some limitation imposed by the law. The Prevention of Electronic Crimes Act (PECA) 2016 restricts users from posting/sharing hate speech content on social media that leads to interfaith, sectarian, or racial hatred.

Since the increase of online hate speech through social media companies like Twitter and Facebook, they were under public and political pressure from many anti-hate government agencies. Germany has passed a law that could fine Twitter, Facebook, and other social media companies up to 40 million for failing to remove defamation, violence, and hate speech within 24 hours. The European Commission has issued a code of conduct to combat online hate speech. According to a report, Facebook removes hate speech content faster than Twitter and YouTube. Facebook accessed 95 % of hate speech notifications in less than 24 hours, while Instagram responded 62 %, Twitter 44%, and YouTube 9% on hate speech notifications. In 2020 Mark Zukerberg announced a comprehensive

Just Accepted

policy on hate speech content used in Ads, Facebook will remove all such contents that target a specific group (Race, national origin, gender, sexual orientation.).

2 LITERATURE REVIEW

A comprehensive literature review has been carried out separately for traditional and deep learning techniques. The first part of the Literature review for both approaches encompasses English including other low-resource languages except Urdu and another part is for the Urdu language.

2.1 Traditional Approaches for English and others Low resource Language except for Urdu

Burnap and Williams [7] used a Bag of words with n-grams ($n = 1 - 5$) and Algorithms ruled-based and spatial-based classifiers and achieved a 98% accuracy. Waseem and Hovy [29] used extra-linguistic features and n-grams ($n = 1 - 4$) and achieved 64.58% efficiency. Linguistic features are used to identify the sense of a word. In [8] Davidson et al. implemented the part of speech tag (POS), bigrams, unigrams, trigrams, and tf-idf using machine learning algorithms SVM, Naïve Bayes (NB), Logistic Regression (LR), Random Forest (RF), Decision Tree (DT) and linear SVM and achieved efficiency 90% on English tweets. Gamback and Sikdar [11] used a char n-gram and word2vec model and achieved 78.3% efficiency. The word2vec is a word embedding technique used to learn word association from a large dataset. Malmasi and Zampieri [31] focused on hate speech profanity and anti-social behavior with char n-gram, n skip-gram and uses a linear SVM model and achieved 78% efficiency. Garima Koushik and Mr. Suresh Kannan Muthusamy [19] used BOW and TF-IDF approaches to train machine learning models, after conducting exhaustive experiments on the Twitter dataset the logistic regression outperforms with the accuracy of 94.11 % on detecting binary classes either hate or not hate. In [18] Kelvin, George, Richard, Kennedy develops an approach for detecting hate speech content by the self-identified hateful community, Naive bayse classifier gives better results with precision, recall, and accuracy values of 58%, 62%, and 68%, respectively. HAJIME and MONDHER in [30] used a unigram approach on a small dataset of 2010 tweets. The experiments are conducted based on binary and ternary classification. Results show that accuracy achieved 87% on binary classification and 78.4% on ternary classification. Trisna and Arif worked on hate speech and cyber pulling detection Indonesian language on the data promulgated during the election 2019 [23]. The paper comprehensively describes the process of developing a dataset with more than 1 Million tweets using Twitter developer API. In the basic preprocessing and implementing machine learning algorithms the Latent Dirichlet Allocation LDA is used to extract the topic from collected tweets and detail sentiment analysis on each category applied to generate a polarity score on balance data. Naïve bayse classifier achieved an accuracy level of 78.7%. Yasemin and Rehime [27] works emphasize hate speech on women. The Turkish data is collected from Twitter with the approach to search tweets from the specific hashtag on a choice of clothing of women. In their research, they applied five machine learning algorithms for the detection of hate speech content against women including Support vector machine, J48, Naïve Bayse, Random Forest, and Random tree. Results show that Naïve bayse performed best with an f score of 62 % among all. OLUWAFEMI and EDUAN [22] targeted to develop of an English corpus from South African tweets to find the hate of offensive content by implementing different machine learning algorithms. Character n-grams, word n-grams, and negative sentiment are used to extract useful features from the dataset. In machine learning, support vector machines, random forest, logistic regression, and gradient boosting are used. Preliminary results show that support vector machine with n-gram is best in the detection of hate speech with a true positive rate of 89.4% and optimized gradient boosting with word n-gram performs best with a positive rate of 86%. The comprehensive analysis presented that multi-tier learning models could overcome the misclassification error rate by 34%. In [13] Purnama et al. used a multinomial logistic regression classifier with a tf-idf feature extraction technique that achieved the best average score of precision of 80.02 %, recall of 82%, and accuracy of 87.66%. Sattam et al. [4] used a supervised classifier including a support vector machine,

Just Accepted

Gaussian naïve bayse, Decision tree, nearest neighbors, and random forest and the target language is English and Spanish. Results show that Naïve bayse, support vector machine and random forest performs wells into account all features with an average f score of 77%. In [20], the authors used n-grams, word n-grams, and word skip grams with a supervised learning model on the annotated dataset with hate speech tweets 2399, offensive 4836 and ok tweets with 7247 out of 14509 tweets and it is found that Support vector machine has been outperformed well for native and variety language identification. The SVM achieved an accuracy of 78% with char 4 gram and with word unigrams SVM achieved 77.5% accuracy. Tom De Smedt and Guy De Pauw [9] examine the quantitative and qualitative analysis of Twitter data containing Jihadist hate speech. The data corpus was collected in compliance with the online procedure. The total data collected is 45K tweets from 2014-2016 covering a region Syria, Iraq, France, United States, Israel, Russia, Jorden, Iran, Egypt, Yemen, Damascus, and London. The SVM model trained on the balanced training set of 45K hates speech tweets and the same for safe tweets. The accuracy achieved is 82% (F1 Score) by applying 3-fold cross-validation.

2.2 Traditional approaches to the Urdu language

In [17] Moin et al. worked on hate speech detection in roman Urdu tweets, 5000 roman Urdu tweets were collected. Tweets are further classified into three classes' Neutral-Hostile, Simple-Complex, and Offensive-Hate speech. Five different machine learning techniques. The results show that logistic regression outperformed all with an F1 score of 0.756 for offensive hate speech tweets. The authors in [2] contributed to improving hate speech detection of Urdu tweets using sentiment analysis. The research addressed the challenges and problems including dimensionality, sparsity, and high skewed classes. The data is annotated in five classes (Neutral, positive, highly positive offensive, highly offensive). The target category is national security and religion. The SMOTE, variable global feature selection techniques are used to handle the sparsity, class imbalance problem. The two machine algorithms SVM and naïve bayse are used. Initial baseline results show that SVM performed well with an F Score (0.626), after improving the performance of the classifier, the results improved with an f score of 0.93. Pervez et al. in [1] collected 5000 tweets of roman Urdu and Urdu respectively. The N-grams technique is used on character and word levels. The seven machine learning algorithms are used to detect offensive or non-offensive tweets from a corpus. The experiment shows that regression models perform best with n-grams about process Urdu tweets. Logitboost and simple logistics outperform others with a score of 95%. In [24], the authros used machine learning algorithms including logistic regression, begging, decisions tree, and ANN to detect abusive language within-corpus of 2400 tweets (1187 Abusive and 1213 no abusive). After performing the classifier task, results show that logistic regression performs best with an f score of 83%.

2.3 Deep Learning Techniques for English and Low resource languages except for Urdu

Badjatiya et al. in [6] worked on a 16K annotated dataset 16K with three target class's racist, sexist, and neither. In their research, extensive experiments were conducted with multiple deep learning architectures in contrast with word embedding to handle the complexity. Results on the benchmark dataset showed that deep learning methods outperform the char/word gram method by 18 f points. [10] contributed well to hate speech detection from multiple languages that appeared in tweets. The experiments were performed on a Convolutional Neural network (CNN) with character-level representation. The result with the best parameter was 0.889 for the dataset containing five languages and 0.83 for the dataset containing seven languages. Lin et al. in [14] did experiments on two different datasets with different sizes (Dataset A containing 9925 and Dataset B containing 31962 records). Traditional approaches (Logistic regression, SVM) and deep learning (LSTM, Stacking, and GRU) were applied to two different datasets. The result showed that Logistic regression outperformed dataset A with an f score of 43% and LSTM on dataset B with an f score of 67.30%. The authors in [12] introduced a deep learning-based hate speech model. The text was classified into four categories, i.e., racism, sexism, both, and not hate speech

Just Accepted

from the dataset of 9K. The experiments showed that CNN performed well with word2vec with an f score of 78%. M. Umar et al. in [21] proposed a combination of CNN and LSTM for performing sentiment analysis for the detection of hate speech on three datasets. The model is analyzed with traditional models, i.e., SVM, Logistic regression, voting classifier, Random forest, and SGD. This study also investigated two different feature extraction techniques TF-IDF and word2vec to determine their impact on accuracy. The results showed that CNN –LSTM performed well among all. Roy et al. [26] developed a deep Convolution neural network for hate speech detection. In this, they used Glove embedding to analyze the semantics of tweets promulgated on Twitter and achieved precision, recall, and f score values of 0.97, 0.88, and 0.92, respectively. In [28] the authors have used an artificial neural network with a back propagation method. The case study identified the hate speech in the sentence. The random accounts were analyzed who involved in hate speech had almost 1235 tweets of which 626 tweets were categorized as hate speech and 583 tweets were classified as non-hate speech. The result was analyzed with hypermeters like Epoch size and learning rate and has been found that results were improved with the tuning of hyper parameters. The overall result obtained an average recall of 90.03%, a precision of 80.6%, and an accuracy of 89.4%.

2.4 Deep learning techniques for the Urdu language

Raza et al. [3] worked on the Urdu tweets dataset of 10K. The different machine learning algorithms are used for hate speech detection and transfer learning to exploit fast text and Bert multi-lingual embedding model. The result shows that Bert improves the f scores of 0.67, 0.68, and 0.69 respectively. Hammad et al. [25] developed annotated roman Urdu dataset of 10K and proposed a CNN-gram deep learning architecture. The results show that transfer learning is better and more beneficial as compared to training a dataset from scratch. Lal et al. in [16] worked on multi-class sentiment analysis of Urdu text using Word / Char n-gram, fastText, and BERT. The result shows that BERT pre-trained embedding outperformed Deep learning and achieved an f score of 81%. Anas et al. in [15] worked on offensive language detection for low resource language using deep sequence model and achieved an overall accuracy of 97.21%.

3 PROPOSED METHODOLOGY

In this section, the comprehensive proposed methodology describes as shown in Fig. 1. The research has been carried out in two parts. The first part is about the detection of abusive content and the second is about the demographical analysis of the dataset.

3.1 Data Collection and Preprocessing

The main challenge encountered in research is the data collection as no dataset is available with demographical properties such as location and time. The web scrapper tool named snscraper has been used for the extraction of tweets from Twitter. Snscrapper tool provides the provision to extract tweets with the support of parameters like search keyword, geolocation, time, and language. The data is collected against each district of Punjab and then combined after finalization of having a dataset of all Punjab districts of Pakistan. The duplicates and empty entries have been removed from the dataset to start the feature extraction process. The annotation of the dataset has been carried out by three expert annotators. The final annotation is based on an agreement of at least two annotators on the same target class (offensive or hate speech or neutral) as shown in Figure Fig. 2. The preprocessing of data encompasses the removal of URL links, hashtags, special characters, emojis, and stop words.

3.2 Features Extraction

Feature extraction is an important step while a step forward in training the model. The feature plays an important role in classifying problems. There is no requirement to learn the model on raw data. Multiple feature extraction

Just Accepted

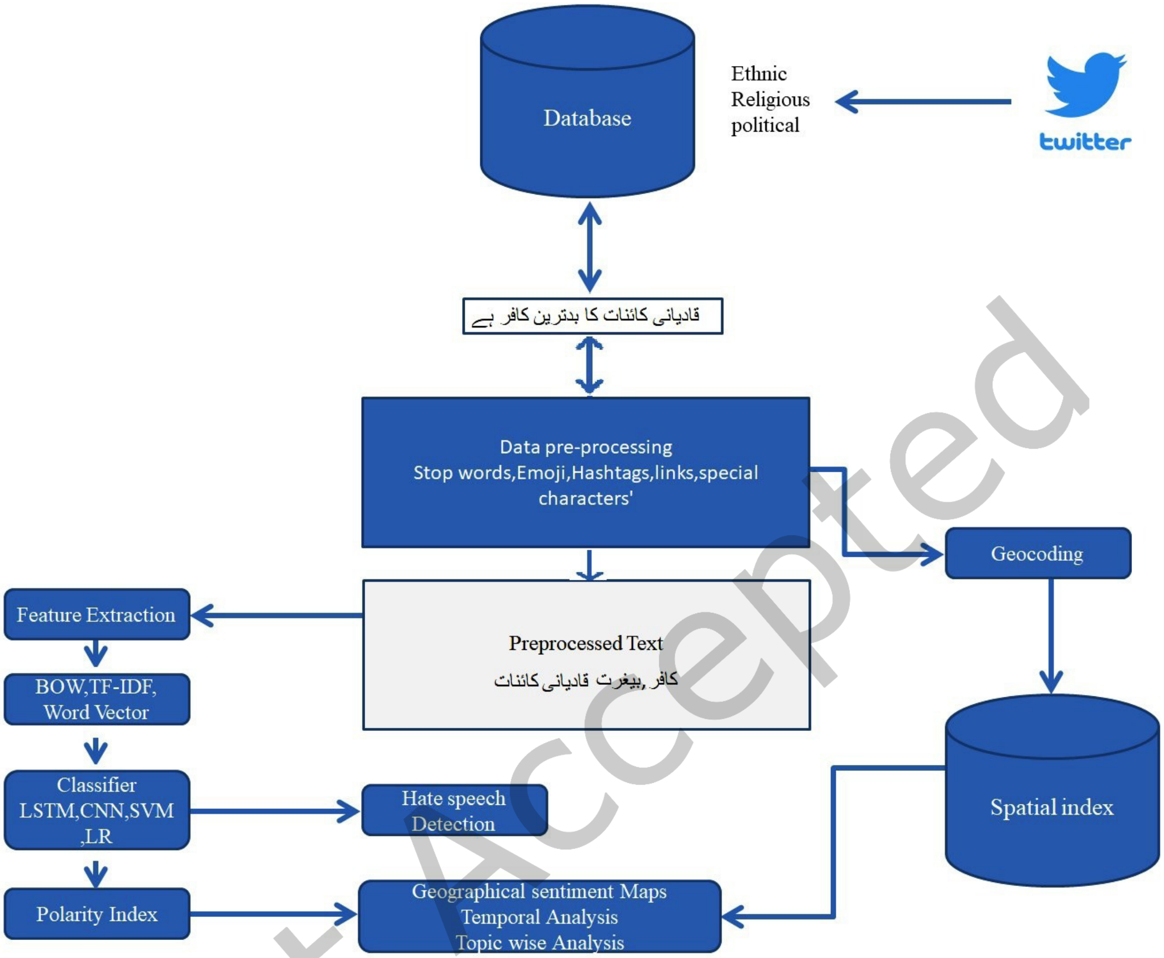


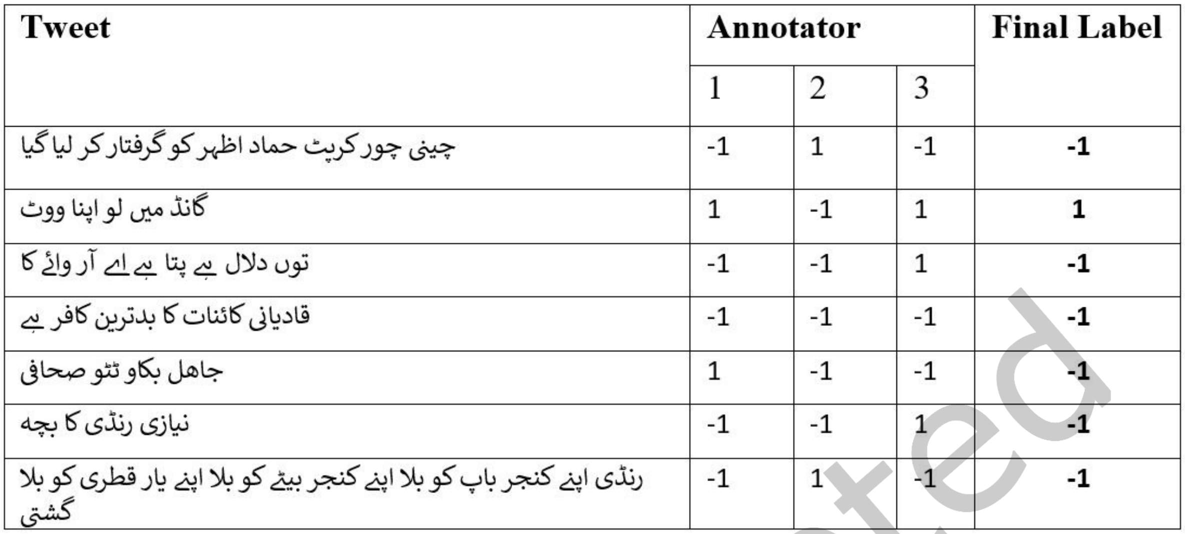
Fig. 1. Proposed Methodology

Just Accepted

techniques are available such as a bag of words (Bow), Term frequency-inverse document frequency (tf-idf), and sequential models. The bag of words is a simple representation of text in numerical form. Tf-idf assigned the value based on the occurrence of the term. Both models do not cover the sequential information about text. Sequential information is very important to capture the semantics of a text. The sequential model is used to capture the contextual information about text.

3.2.1 Word Index Dictionary. Initially, data is in textual format, deep neural networks understand the numeric data format. The dataset is required to transform into numerical or vector representation and then converted the transform data into a word index dictionary. The word index dictionary possesses sequential information about each term.

3.2.2 Input Sequence Padding. The transformation of data into vector representation have different length and shape. The Input data should be in the same size and shape before input into the training and testing process.



Tweet	Annotator			Final Label
	1	2	3	
چینی چور کرپٹ حماد اظہر کو گرفتار کر لیا گیا	-1	1	-1	**-1**
گانڈ میں لو اپنا ووٹ	1	-1	1	**1**
توں دلال ہے پتا ہے اے آر وائے کا	-1	-1	1	**-1**
قادیانی کائنات کا بدترین کافر ہے	-1	-1	-1	**-1**
جاھل بکاو ٹٹو صحافی	1	-1	-1	**-1**
نیازی رنڈی کا بچہ	-1	-1	1	**-1**
رنڈی اپنے کنجر باپ کو بلا اپنے کنجر بیٹے کو بلا اپنے یار قطری کو بلا گشتی	-1	1	-1	**-1**

Fig. 2. Annotation Methodology

Just Accepted

The input sequence is padded in equal length and shape. There are two types of sequence padded available one is pre-pad and the others are post-pad. In Pre padding sequence the vector is padded with zero in beginning and in post-padding the zero is padded after the input vectors.

3.2.3 Word Embedding. Embedding is used to reduce the complexity of data by translating the data into vectors. It is very challenging to do experiments on non-numeric data. The embedding converts the high-dimensional data into low-dimensional data by preserving its meaningful information. One more benefit of embedding is that it captures the semantics from the input. The data is converted into numeric or vector form based on the distance. We used the Word2vec model in our research. We trained our large dataset containing 0.2 M tweets with the dimensions m = 128.

3.3 Classification Models

There are mainly two approaches used for classification problems one is machine learning and the other is deep learning techniques. Machine learning (ML) focuses on allowing computers to perform tasks without explicit programming and deep learning (ML) is a subset of machine learning based on Artificial Neural Networks (ANN).

3.3.1 Machine Learning Techniques (Support Vector Machine and Logistic Regression). Support vector machine is very efficient and useful in multi-class problems, memory efficient, and very effective in high dimensional data. The SVM takes the data points as input and output hyperplanes that best separate the points. The Hyper plane equation is represented as

$$\mathbf{W}^t\mathbf{X} = 0 \,, \tag{1}$$

where, **W** represents normal to the hyperplanes. The kernel function is used to calculate the data point's separations. Given n feature vector f for three classes $[1, 0, -1]$, the hyperplanes can be defined

$$w.fn + b = 1 \tag{2}$$

$$w.fp + b = 1 \tag{3}$$

$$w.fp + b = 1 \tag{4}$$

The distance between the positive and negative hyperplanes is $2/||\mathbf{W}||$ and the margin size is $1/|\mathbf{W}|$
Logistic regression works well on independent variables. The outcome of logistic regression is the basic probability so the dependent variable remains bounded in the range between 0 and 1. For the input vector Fi, weighted matrix S, and bias values b, the probability that Fi relates to class 'K' is the value of the variable y, mathematically written as

$$h\theta(Fi) = P(y = K|Fi, s, b) \tag{5}$$

Where h is the hypothesis and θ represents parameters s and b. The probabilities for the input vectors can be determined by the softmax function as represented as

$$P(y = K|Fi, s, b) = softmax(s.Fi + b) \tag{6}$$

3.3.2 Deep Learning Techniques (LSTM and CNN). Long short-term memory consists of four layers, the Embedding layer also known as the Input layer, the LSTM layer, the dense layer, and the Output layers. The embedding layer has some predefined parameters like Input dimensions we assigned a vocab size that is 68671 for our dataset, output dimensions assigned as 64 and a maximum input length is 108 for our dataset.
We used a hidden layer to have stable and effective results. Rectified linear unit (ReLu) is used in the dense layer and on the output layer Softmax function is used for prediction, the number of neurons used in these layers is equal to the number of target classes. Sparse categorical entropy is used to calculate the cost of learning algorithms. We use the callback to monitor the overfitting we set the threshold as 3 which means if the validation loss did not change for 3 consecutive iterations the iterations automatically stops. Categorical cross entropy/ negative log-likelihood has been used to compute the cost of the learning algorithm and RMSprop as the optimization algorithm.. Let gt denote the gradient at time step t and λ denote the momentum, the running average denoted by $g_2(t)$ at time step t depends on the current gradient and the previous average, i.e., for our dataset, output dimensions assigned as 64 and a maximum input length is 108 for our dataset.

$$E\{g_2(t)\} = \lambda E\{g_2(t-1)\} + (1-\lambda) g_2(t) \ . \tag{7}$$

Convolutional Neural Network CNN is a deep neural network useful for detecting features automatically minimizing human effort. CNN Architecture consists of an input layer, a Convolutional layer, pooling, fully connected, and an output layer. Input layer that extracts useful information from the input for our case we set the parameter with the size of vocabulary, i.e., 68671 with embedding dimensions 64. We set max pooling value 2 to keep salient features. Convolutional layer that is used for useful feature extraction. We Used the ReLu activation function in this layer. We use dense layers with units 1024 and 512. All extracted features are concatenated to form a feature vector and passed as input to the output layer using the Softmax activation function to classify the sentence. We set the dropout value to 0.02, the learning rate to 0.000055, and 10 epochs.

3.4 Demographical Analysis

To our best knowledge, there is no work done in demographical analysis of promulgated within in Pakistan. We have conducted a spatial and temporal analysis of tweets against 36 districts of Punjab within the duration of 2018-Apr22.

Just Accepted

3.4.1 Temporal Analysis. This section is about the temporal analysis of our research dataset. The overall dataset contains the time in which the tweet has been recorded from the duration Jan 2018 – Apr 2022. Table 1 shows the overall statistics of data w.r.t time.

Figure 3 depicts the class distribution w.r.t time from 2018-2022. It has been observed that hate speech and offensive tweets are found more in 2019, and 2022 compared with the year 2018, 2020, and 2021. The hate speech (3903 in no) tweets in the first 4 months of the year 2022 stand high with offensive containing 12258 tweets. The trend shows that the growth in hate speech tweets rapidly increase as we moved from 2018 - 2022.

Table 1. Temporal Analysis

Years	**H S**	**Offensive**	**Neutral**	**Total**
2018	768	2483	15727	18978
2019	1130	2948	16845	20923
2020	471	1590	16863	18924
2021	833	2786	20199	23818
2022	3903	12258	100831	116992
Total	**7105**	**22065**	**170465**	**199635**

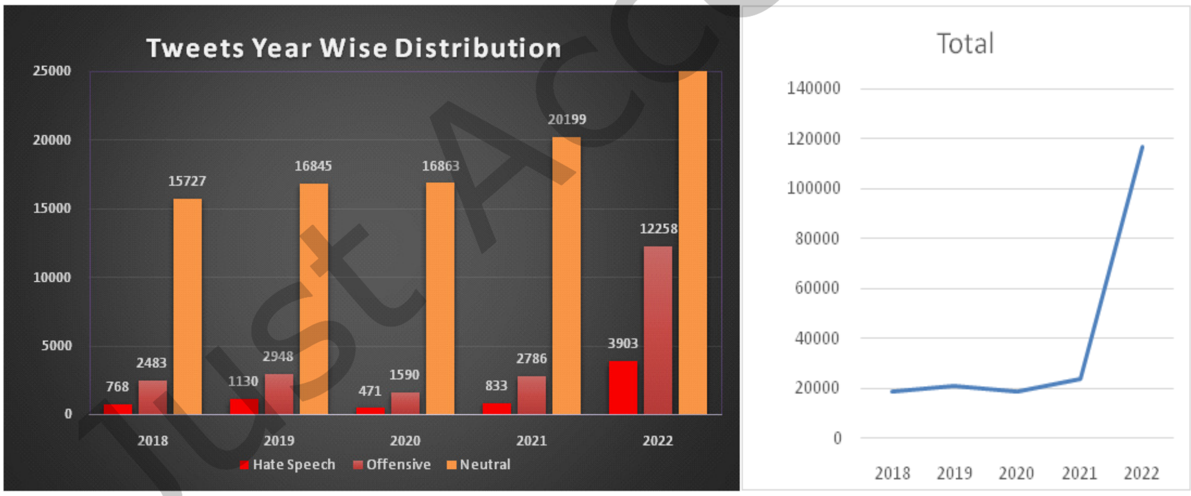


Fig. 3. Temporal Analysis

3.4.2 Spatial Analysis. This section represents the spatial analysis of our dataset. We extracted the Latitude and longitude of each Punjab district against the tweet. Table2 shows the spatial data statistics.

Figure 4 is the statistical map used to provide the visualization of the variable varies across the geographical location. We used the same map (as shown in Fig. 4) for visualization of the hate speech data promulgated in Pakistan from 2018-22. It has been observed that hate speech remains high in Lahore (962), Gujranwala (784),

Just Accepted

Table 2. Spatio Data Statistics

District	Total	Hate Speech	Offensive	Neutral
Attock	619	14	76	529
RYK	2319	142	318	1859
RajanPur	530	21	47	462
BWP	5384	231	718	4435
Lodhran	1509	41	82	1386
Bahwalnagur	1264	69	225	970
Chakwal	1513	72	132	1309
Vehari	4421	80	215	4126
Chinot	2247	59	189	1999
DGK	3644	137	654	2853
FSB	12451	593	1722	10136
GJW	17954	784	2183	14987
Gujrat	8407	202	601	7754
Hafizabad	1004	72	135	797
Jhang	1231	36	72	1123
Jhelum	3800	86	266	3448
Kasur	2619	116	340	2163
Khanewal	3796	81	289	3426
Khushab	913	29	115	769
Lahore	29103	962	3341	24800
Layyah	210	8	14	188
Mandi	5822	134	449	5239
Mianwali	2208	89	250	1869
Multan	22855	596	2274	19985
Muzafargar	1699	97	335	1267
Nanka	425	13	43	369
Norwal	507	31	114	362
Okara	3780	181	593	3006
Pakpattan	934	83	133	718
RWP	24225	784	2811	20803
Sahiwal	4174	123	431	3620
Bakhar	3595	114	339	3142
Toba	454	26	85	343
Shiekhpura	2631	83	218	2330
Sargodha	8446	341	863	7242
Sialkot	12769	575	1543	10651
Total	199635	7101	22065	170465
Ratio		3.52 %	11.01 %	84.50 %

Faisalabad (593), Rawalpindi (784), Multan(596), Sialkot(575), and Sargodha (341) districts. Figure 5 shows the overall offensive tweets recorded in the Punjab district. It has been recorded that offensive tweets in Punjab districts such as Lahore (3341), Rawalpindi (22838), Multan (2274) Gujranwala (2183), Faisalabad (1722), and Sialkot (1543) remained high, especially in 2022 the overall offensive tweets were found 22065 overall.

4 EXPERIMENTS AND RESULTS

The labeled dataset having 0.2 M tweets is trained on machine learning techniques Support vector machine Logistic regression and deep learning algorithms long short-term memory and Convolutional neural network. The evaluation metrics used in our research are precision, recall, f score, and accuracy. The experiments have been conducted on both imbalanced and balanced data.

Just Accepted

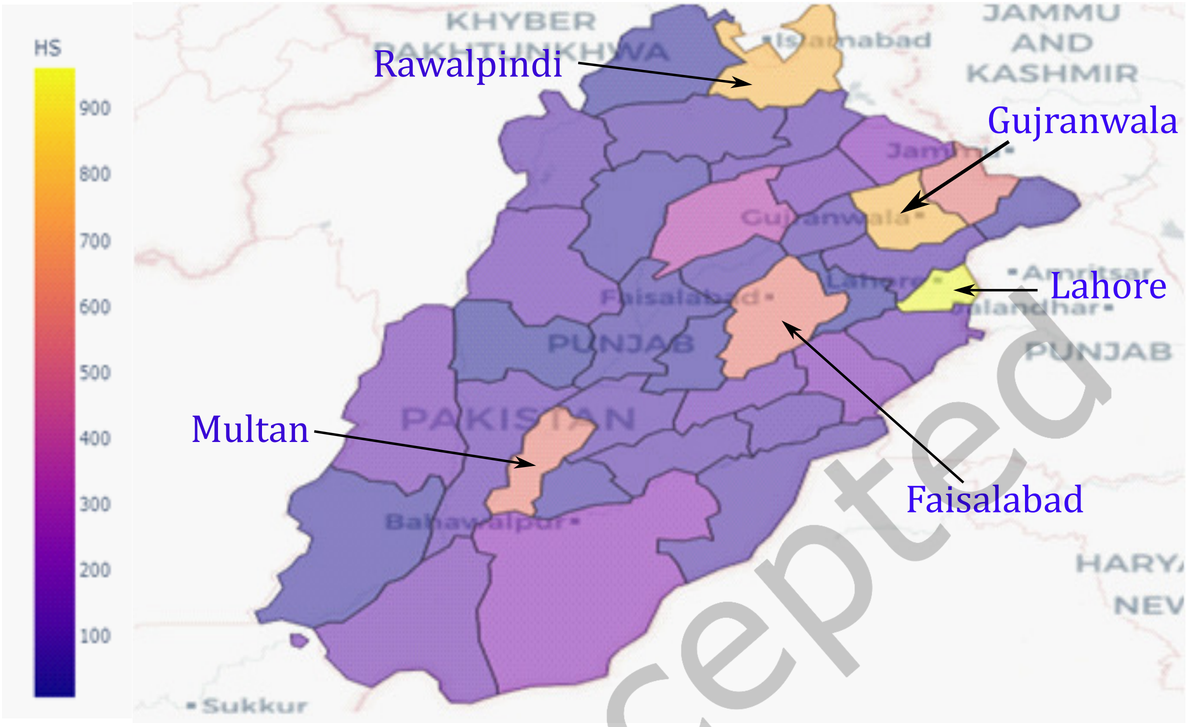


Fig. 4. Choropleth Map for visualizing Hate Speech Data

Just Accepted

4.1 Imbalanced Data

The experiments have been performed on data having about 0.2 M tweets including religious, political, and ethnic groups. Two different approaches Support Vector Machine (SVM), Logistic Regression (LR) from Machine learning and Long Short Term Memory (LSTM), Convolutional Neural Networks were used for the detection of offensive and hate speech content in the dataset. Bag of Words (Bow), TF-IDF, and word2vec are used for features engineering. The Precision, Recall, and F Scores are obtained against each and compared to the result. The result highlighted in bold represents the Highest F Score achieved against respective algorithms. Table 3 shows below the results of all 3 target classes against each algorithm.

The above results show that F scores for target class Neutral labeled as 0 achieved a maximum F score of 96 with all three algorithms Support vector Machine, Logistic Regression, and LSTM. LSTM Sequential model performs outclass in detecting offensive and hate speech contents in imbalanced data containing 0.2 M Tweets. The Highest F score achieved against the offensive type through LSTM sequential model is 75 and for hate speech is 64. It has been observed during experiments all deep learning algorithms Perform well on large data because they need more data to learn, and train. The experiments through Deep learning algorithms remained outstanding as compared to traditional approaches. Figure 6 shows the boxplot depicting the F score against all four classes.

4.2 Balanced Data

The experiments were performed on balanced data for comparison of results. Oversampling and under-sampling of data have been implemented. The distribution of class in overall data is shown in Table 4

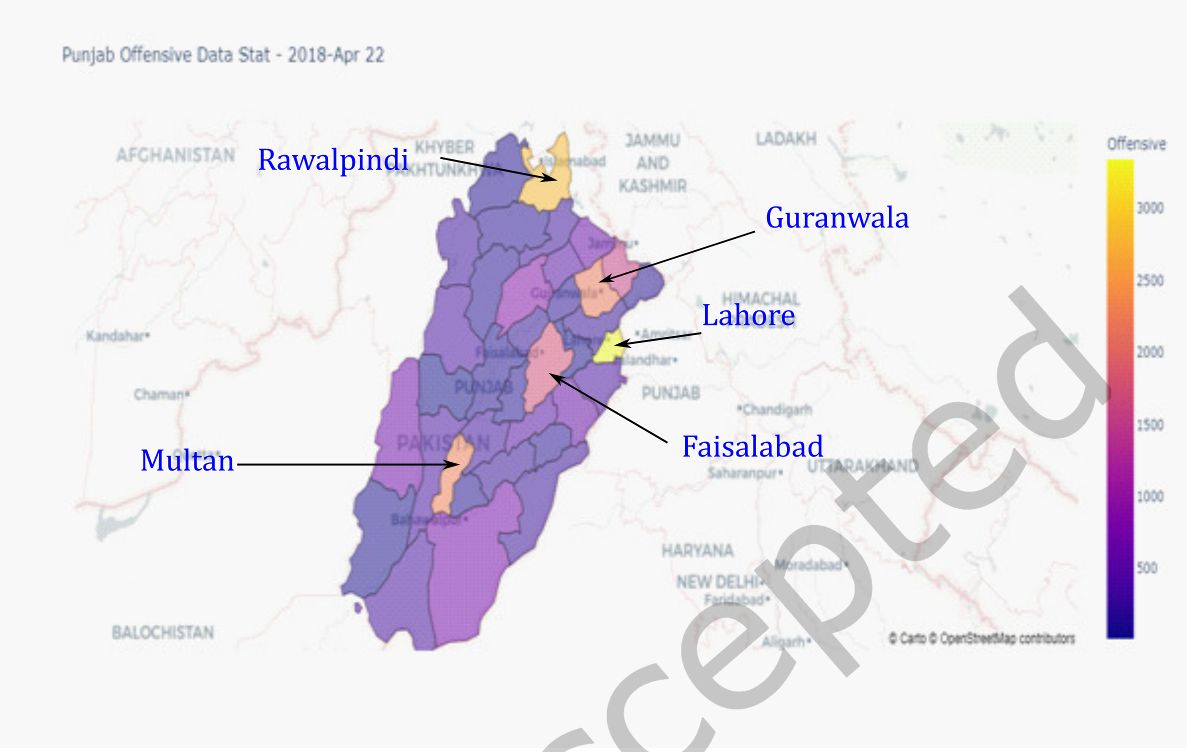


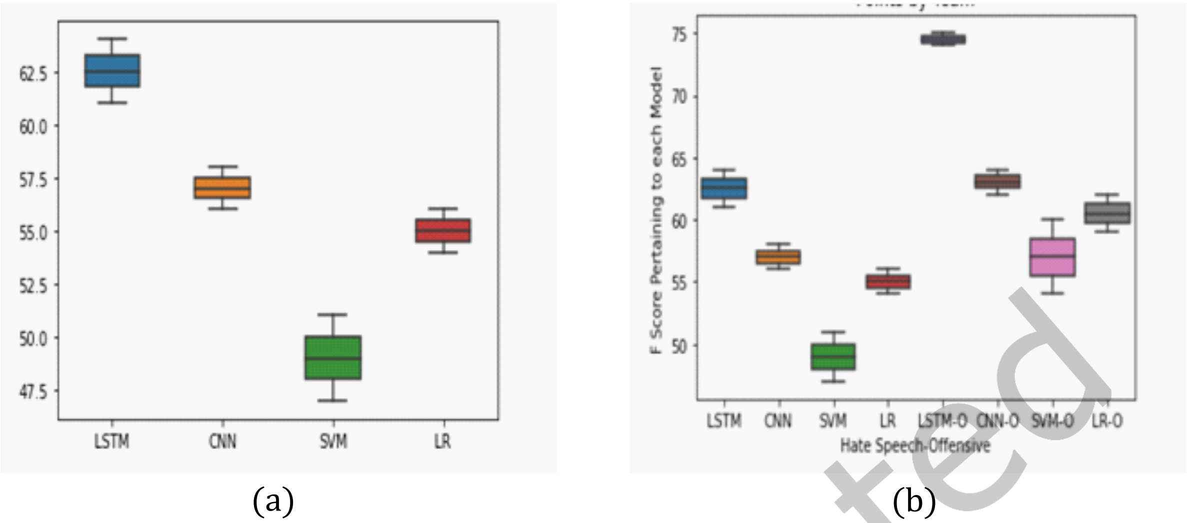
Fig. 5. Choropleth Map for visualizing Offensive Speech Data

Just Accepted

Table 3. Results for all Classifier – Imbalanced Data

classifier	**Features**	**Neutral**			**Offensive**			**Hate Speech**			**accuracy %**
		P	R	F	P	R	F	P	R	F	
SVM	**BOW**	97	95	96	57	64	60	45	59	51	91
	TF-IDF	95	94	94	55	54	54	44	51	47	88
LR	**BOW**	98	94	96	57	69	**62**	48	68	**56**	**91**
	TF-IDF	96	95	96	51	61	59	51	57	54	90
LSTM	-	90	89	89	70	80	**75**	75	54	**64**	**89**
	word2vec	82	91	87	73	74	**74**	72	52	61	87
CNN	**word2vec**	96	96	96	60	63	62	64	47	56	**92**

The result depicts that improvement in achieving a High F Score for offensive and hate speech class on balance data. CNN performs well on balance data with a yielded F score of 93. Logistic regression and Long Short term memory models performed well in detecting offensive contents with F scores of 77 and 82 respectively. The overall accuracy achieved against balanced data and imbalance data is shown in Table 5.



(a) (b)

Fig. 6. Boxplot yield F score Vs. (a) hate speech and (b) offensive and hate speech

Just Accepted

Table 4. Balanced Data

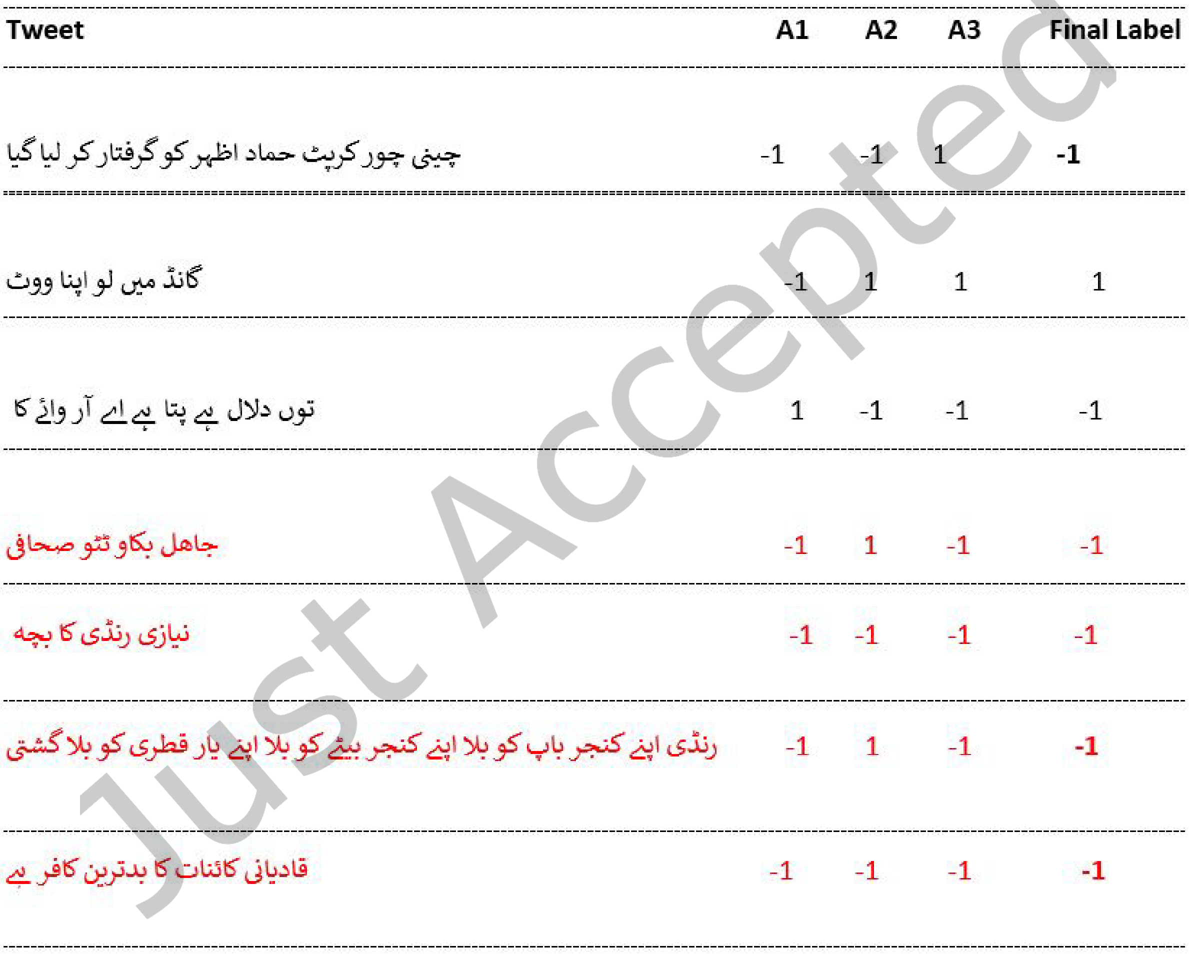
Class	Balanced Data	Original	Operation
Neutral	23000	170465	**Under Sampling**
Offensive	22225	20065	-
Hate Speech	21315	7105	**Oversampling**

Table 5. Results for all Classifier – Balanced Data

classifier	Features	Neutral			Offensive			Hate Speech			accuracy %
		P	R	F	P	R	F	P	R	F	
SVM	**BOW**	91	87	89	68	82	74	92	81	86	88
	TF-IDF	86	85	85	67	76	71	90	80	85	87
LR	**BOW**	93	87	90	72	82	**77**	88	84	86	**89**
	TF-IDF	89	87	88	71	82	**77**	93	83	88	88
LSTM	-	87	91	89	79	85	**82**	97	88	**92**	**91**
	word2vec	82	91	87	78	85	**82**	96	88	92	88
CNN	**word2vec**	90	85	87	84	79	81	88	99	**93**	87

4.3 Annotation Process

Annotation process was a very cumbersome job as we had to annotate the dataset contained 0.2 M tweets. Three annotators including one domain expert started annotation on the combined dataset. The process was started by dividing the dataset into 3 parts each annotator got 67K tweets to be annotated. Annotation guidelines were already formulated in Chapter 3. To annotate such huge data we mutually decided to complete the annotation task within 3 months timestamp. It was decided to label a dataset in three classes Hate speech labeled as -1, Offensive as 1 and Neutral / Positive as 0. As the initial annotation by each annotator, the file of Annotator A handed over to Annotator C and vice versa for cross-verification of the annotation process and omission of any human mistake (if any). The voting system was maintained while finalizing the labeling. For example, to finalize the tweet label, there should be a minimum of 2 annotators agreed on the same label either Hate speech or Offensive. Figure 7 shows the annotation process.

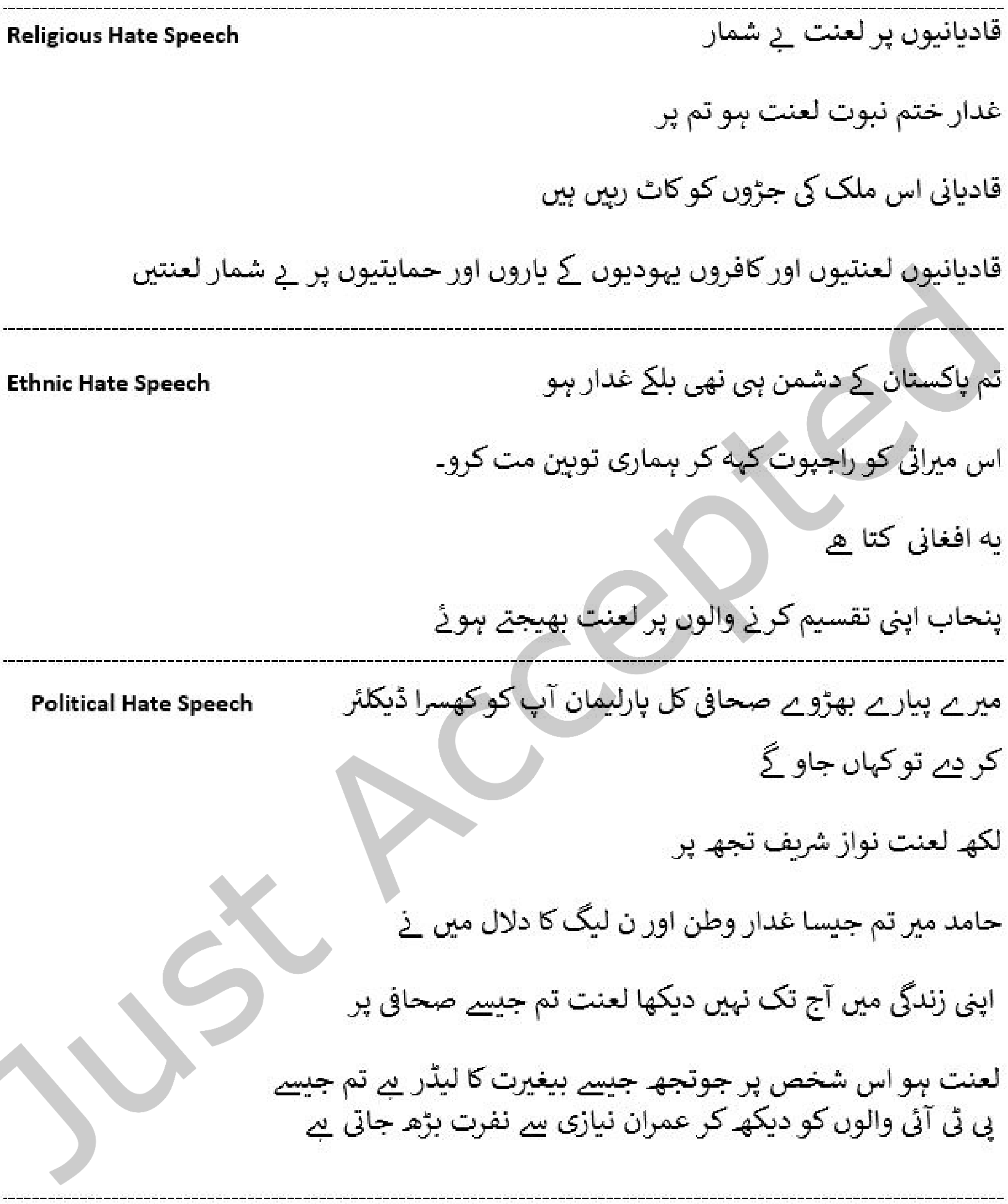


Tweet	A1	A2	A3	Final Label
چینی چور کرپٹ حماد اظہر کو گرفتار کر لیا گیا	-1	-1	1	**-1**
گانڈ میں لو اپنا ووٹ	-1	1	1	1
توں دلال ہے پتا ہے اے آر وائے کا	1	-1	-1	-1
جاہل بکاو ٹٹو صحافی	-1	1	-1	-1
نیازی رنڈی کا بچہ	-1	-1	-1	-1
رنڈی اپنے کنجر باپ کو بلا اپنے کنجر بیٹے کو بلا اپنے یار قطری کو بلا گشتی	-1	1	-1	**-1**
قادیانی کائنات کا بدترین کافر ہے	-1	-1	-1	**-1**

Just Accepted

Fig. 7. Annotation process

Figure 8 shows the group wise hate speech tweets in dataset such as Ethnic – Hate Speech, Political - Hate Speech, Religious - Hate Speech



Religious Hate Speech

قادیانیوں پر لعنت بے شمار

غدار ختم نبوت لعنت ہو تم پر

قادیانی اس ملک کی جڑوں کو کاٹ رہیں ہیں

قادیانیوں لعنتیوں اور کافروں یہودیوں کے یاروں اور حمایتیوں پر بے شمار لعنتیں

Ethnic Hate Speech

تم پاکستان کے دشمن ہی نہی بلکے غدار ہو

اس میراثی کو راجپوت کہہ کر ہماری توہین مت کرو۔

یه افغانی کتا ھے

پنحاب اپنی تقسیم کرنے والوں پر لعنت بھیجتے ہوئے

Political Hate Speech

میرے پیارے بھڑوے صحافی کل پارلیمان آپ کو کھسرا ڈیکلئر
کر دے تو کہاں جاو گے

لکھ لعنت نواز شریف تجھ پر

حامد میر تم جیسا غدار وطن اور ن لیگ کا دلال میں نے

اپنی زندگی میں آج تک نہیں دیکھا لعنت تم جیسے صحافی پر

لعنت ہو اس شخص پر جوتجھ جیسے بیغیرت کا لیڈر ہے تم جیسے
پی ٹی آئی والوں کو دیکھ کر عمران نیازی سے نفرت بڑھ جاتی ہے

Fig. 8. Category wise Hate speech Tweets

Just Accepted

5 CONCLUSION AND FUTURE WORK

Hate Speech becomes a global problem on social media nowadays. A variety of languages are used for expressing and sharing ideas on social media which makes the detection of hate speech content a challenge. Machine learning and deep learning algorithms have witnessed effective countermeasures in detecting and removal of such abusive content on social media. Several studies have been carried out on this problem, especially in the English language is the most spoken language in the world. Urdu, being a low-resource language very less amount of work has been carried out either with the small dataset or in roman Urdu. To our best knowledge, there is no work carried out on Urdu's large dataset and demographical parameters in Pakistan. To Our best knowledge, we developed a large corpus having 0.2 M tweets. The corpus is collected against 36 districts of Punjab for the period 2018-Apr 2022. The other contribution to our research is to annotate such a large dataset that takes an ample amount of time. We introduced a new definition of hate speech for our data and annotate the data accordingly. We explored the useful features of Urdu and implement the machine and deep learning algorithms. We observed that deep learning algorithms are most effective and efficient on a large dataset. Embedding features perform well in detecting infrequent patterns of hate speech. The traditional model outperforms deep learning models. It may be due to class imbalance problems, Data Sparsity, and high dimensionality and it is a challenging task to reduce and overcome the problems before moving further in the detection process. That is why we think that deep learning algorithms contribute well in this case. We carried out an error analysis of these algorithms and found it challenging to make the process more effective and efficient as we encountered the overfitting problem for our dataset. Our research establishes a baseline for the detection of hate speech in the Urdu language. Future work should address the challenges identified in our research like data sparsity, High Skew, and high dimensionality problems. Another aspect is to incorporate advanced techniques to distinguish between different degrees of language such as sarcasm, implicit hate speech, word sense, and target of abuse. To annotate the large dataset it is necessary to develop a comprehensive sentiment dictionary for the Urdu language. Secondly, the focus should be on minority classes by analyzing every hidden pattern. The language sense is also important, especially for Low resource language like the implementation of word Segmentation, etc. Advanced embedding features should be applied to more data to have more effective and accurate results.

REFERENCES

[1] Muhammad Pervez Akhter, Zheng Jiangbin, Irfan Raza Naqvi, Mohammed Abdelmajeed, and Muhammad Tariq Sadiq. 2020. Automatic detection of offensive language for urdu and roman urdu. *IEEE Access* 8 (2020), 91213–91226.

[2] Muhammad Z Ali, Sahar Rauf, Kashif Javed, Sarmad Hussain, et al. 2021. Improving hate speech detection of Urdu tweets using sentiment analysis. *IEEE Access* 9 (2021), 84296–84305.

[3] Raza Ali, Umar Farooq, Umair Arshad, Waseem Shahzad, and Mirza Omer Beg. 2022. Hate speech detection on Twitter using transfer learning. *Computer Speech & Language* 74 (2022), 101365.

[4] Sattam Almatarneh, Pablo Gamallo, Francisco J Ribadas Pena, and Alexey Alexeev. 2019. Supervised classifiers to identify hate speech on English and Spanish tweets. In *Digital Libraries at the Crossroads of Digital Information for the Future: 21st International Conference on Asia-Pacific Digital Libraries, ICADL 2019, Kuala Lumpur, Malaysia, November 4–7, 2019, Proceedings 21*. Springer, 23–30.

[5] Aisha Azhar, Muhammad Nasir Malik, and Asif Muzaffar. 2019. Social network analysis of Army Public School Shootings: Need for a unified man-made disaster management in Pakistan. *International journal of disaster risk reduction* 34 (2019), 255–264.

[6] 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*. 759–760.

[7] Pete Burnap and Matthew L Williams. 2016. Us and them: identifying cyber hate on Twitter across multiple protected characteristics. *EPJ Data science* 5 (2016), 1–15.

[8] Thomas Davidson, Debasmita Bhattacharya, and Ingmar Weber. 2019. Racial bias in hate speech and abusive language detection datasets. *arXiv preprint arXiv:1905.12516* (2019).

[9] Tom De Smedt, Guy De Pauw, and Pieter Van Ostaeyen. 2018. Automatic detection of online jihadist hate speech. *arXiv preprint arXiv:1803.04596* (2018).

[10] Aya Elouali, Zakaria Elberrichi, and Nadia Elouali. 2020. Hate Speech Detection on Multilingual Twitter Using Convolutional Neural Networks. *Revue d'Intelligence Artificielle* 34, 1 (2020).

Just Accepted

[11] Björn Gambäck and Utpal Kumar Sikdar. 2017. Using convolutional neural networks to classify hate-speech. In *Proceedings of the first workshop on abusive language online*. 85–90.

[12] Björn Gambäck and Utpal Kumar Sikdar. 2017. Using convolutional neural networks to classify hate-speech. In *Proceedings of the first workshop on abusive language online*. 85–90.

[13] Purnama Sari Br Ginting, Budhi Irawan, and Casi Setianingsih. 2019. Hate speech detection on twitter using multinomial logistic regression classification method. In *2019 IEEE International Conference on Internet of Things and Intelligence System (IoTaIS)*. IEEE, 105–111.

[14] Lin Jiang and Yoshimi Suzuki. 2019. Detecting hate speech from tweets for sentiment analysis. In *2019 6th International conference on systems and informatics (ICSAI)*. IEEE, 671–676.

[15] Anas Ali Khan, M Hammad Iqbal, Shibli Nisar, Awais Ahmad, and Waseem Iqbal. 2023. Offensive Language Detection for Low Resource Language Using Deep Sequence Model. *IEEE Transactions on Computational Social Systems* (2023).

[16] Lal Khan, Ammar Amjad, Noman Ashraf, and Hsien-Tsung Chang. 2022. Multi-class sentiment analysis of urdu text using multilingual BERT. *Scientific Reports* 12, 1 (2022), 5436.

[17] Muhammad Moin Khan, Khurram Shahzad, and Muhammad Kamran Malik. 2021. Hate speech detection in roman urdu. *ACM Transactions on Asian and Low-Resource Language Information Processing (TALLIP)* 20, 1 (2021), 1–19.

[18] Kelvin Kiema Kiilu, George Okeyo, Richard Rimiru, and Kennedy Ogada. 2018. Using Naïve Bayes algorithm in detection of hate tweets. *International Journal of Scientific and Research Publications* 8, 3 (2018), 99–107.

[19] Bhusarapu Lohitha, V Mogana, and J Jegan Amarnath. 2022. A Comparison of Different Models for the Detection of Hate Speech. In *2022 1st International Conference on Computational Science and Technology (ICCST)*. IEEE, 492–496.

[20] Shervin Malmasi and Marcos Zampieri. 2017. Detecting hate speech in social media. *arXiv preprint arXiv:1712.06427* (2017).

[21] Usman Naseem, Imran Razzak, and Ibrahim A Hameed. 2019. Deep Context-Aware Embedding for Abusive and Hate Speech detection on Twitter. *Aust. J. Intell. Inf. Process. Syst.* 15, 3 (2019), 69–76.

[22] Oluwafemi Oriola and Eduan Kotzé. 2020. Evaluating machine learning techniques for detecting offensive and hate speech in South African tweets. *IEEE Access* 8 (2020), 21496–21509.

[23] Shatakshi Raman, Vedika Gupta, Preeti Nagrath, and KC Santosh. 2022. Hate and aggression analysis in NLP with explainable AI. *International Journal of Pattern Recognition and Artificial Intelligence* 36, 15 (2022), 2259036.

[24] Muhammad Owais Raza, Qaisar Khan, and Ghulam Muhammad Soomro. 2021. Urdu Abusive Language Detection using Machine Learning. (2021).

[25] Hammad Rizwan, Muhammad Haroon Shakeel, and Asim Karim. 2020. Hate-speech and offensive language detection in roman Urdu. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*. 2512–2522.

[26] Pradeep Kumar Roy, Asis Kumar Tripathy, Tapan Kumar Das, and Xiao-Zhi Gao. 2020. A framework for hate speech detection using deep convolutional neural network. *IEEE Access* 8 (2020), 204951–204962.

[27] Havvanur Şahi, Yasemin Kılıç, and Rahime Belen Sağlam. 2018. Automated detection of hate speech towards woman on Twitter. In *2018 3rd international conference on computer science and engineering (UBMK)*. IEEE, 533–536.

[28] Nabiila Adani Setyadi, Muhammad Nasrun, and Casi Setianingsih. 2018. Text analysis for hate speech detection using backpropagation neural network. In *2018 international conference on control, electronics, renewable energy and communications (ICCEREC)*. IEEE, 159–165.

[29] 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.

[30] Hajime Watanabe, Mondher Bouazizi, and Tomoaki Ohtsuki. 2018. Hate speech on twitter: A pragmatic approach to collect hateful and offensive expressions and perform hate speech detection. *IEEE access* 6 (2018), 13825–13835.

[31] Marcos Zampieri, Shervin Malmasi, Preslav Nakov, Sara Rosenthal, Noura Farra, and Ritesh Kumar. 2019. Predicting the type and target of offensive posts in social media. *arXiv preprint arXiv:1902.09666* (2019).

Just Accepted