Vol. 6 (1), 2026
E-ISSN: 2583-4452

Identification of Bystanders from Cyberbullying Tweets: Performance Evaluation Using Multi-Feature Extraction Techniques and Neural Network Architectures

Ayushi Gupta[1], Veenu Bhasin[2] and Anamika Gupta[1*]
[1] Shaheed Sukhdev College of Business Studies, University of Delhi, Delhi
[2] PGDAV College, University of Delhi, Delhi
[*] Corresponding Author ✉ anamikargupta@sscbsdu.ac.in,

ABSTRACT

Cyberbullying involves aggressive or harmful behavior conducted by individuals or groups through online platforms. Bystanders, those who witness such incidents, can play varying roles: defenders who intervene, instigators who support the bully, or neutral onlookers who remain passive. Understanding bystander behavior is crucial to assessing the scale and impact of cyberbullying, but the lack of labeled data limits progress. To address this, the CYBY23 dataset was released on Kaggle in October 2023, featuring Twitter threads with main tweets and bystander replies where bystander roles have been manually annotated. Labeling of bystander roles is a labor-intensive task; there's a clear need for automated methods to label bystander roles. In this paper, an efficient model, based on neural network (NN) architectures, is proposed for the automatic labeling of bystander roles.

The CYBY23 dataset was first balanced using SMOTE. We then evaluated different sets of features (Toxicity, Sentiment, Empath, TF-IDF, POS) using three NN models - Feed-forward Neural Network, Extreme Learning Machine, and Multi-Layer Perceptron. The efficiency of the models was compared based on accuracy, precision, recall, and F1 score. Features TF-IDF and POS, when taken together, delivered the best performance. The feed-forward neural network gave the best performance among the network architectures.

Keywords: *Natural Language Processing, Cyberbullying, Defender, Instigator, Impartial*

1. Introduction

Cyberbullying refers to the intentional use of digital technologies, such as social media or messaging platforms, to inflict harm, humiliation, or distress on others. It often includes aggressive behaviors like harassment, threats, or the spreading of harmful content, which can significantly impact the psychological well-being of the victims (Smith, 2019). Bystanders in cyberbullying are individuals who witness or are aware of bullying behavior but are not directly involved. They can either contribute to the situation by encouraging the bully or help the victim by intervening or reporting the incident (Kowalski et al., 2020; Salmivalli, 2018; DeSmet et al., 2019).

With the increasing use of social media platforms, online communication now plays a major role in daily life. Twitter, in particular, serves as a prominent medium for individuals to express their thoughts, react to current events, and engage in public discourse. Among the numerous social behaviors exhibited on such platforms, bystander behavior - where users observe but do not participate or intervene in incidents such as cyberbullying, misinformation spread, or online harassment - has emerged as a critical area of study. Ifwe can identify and analyze how bystanders behave, we can better understand how users engage online. Further, intervention strategies can be designed to create safer online environments, and harmful behaviour can be reduced.

As per the recent research studies, bystanders play a crucial role in cyberbullying dynamics. Studies show that approximately 70% to 80% of individuals witnessing bullying, either online or offline, do not intervene (Huang et al., 2020). However, when bystanders do step in, bullying stops within 10 seconds in 57% of cases (Hawkins et al., 2019). People often hesitate to step in and help because they are afraid they might become the next target of bullying. Also, they are not sure about what they should do or how to respond appropriately (Obermaier et al., 2016).

Research that focuses on understanding the role of bystanders in cyberbullying is very important. When we identify what bystanders do (for example, whether they support the victim, ignore the situation, or encourage the bully), we can better

Received on: 27 Apr 2026 | Accepted on: 28 Jun 2026 | Published Online on: 25 July 2026

understand how cyberbullying works and continues. By studying how bystanders react and what influences their behavior, researchers can learn why cyberbullying either continues or gets reduced (Salmivalli, 2018). It can be used in developing targeted interventions and educational programs designed to empower bystanders to act. Effective bystander interventions can reduce the incidence and impact of cyberbullying (McMahon et al., 2014). Further, understanding bystanders' motivations and barriers to intervention, researchers can improve prevention strategies, making them more effective in encouraging active participation against cyberbullying (Gini et al., 2014).

The majority of publicly available datasets do not adequately address the roles of bystanders in cyberbullying. Given the significant impact that bystanders can have, it is essential to classify their roles so that there can be a better understanding and addressing of the phenomenon of cyberbullying. In this research work, we are aiming to explore the publicly available CYBY23 (Alfurayj et al., 2023) dataset, which has been manually annotated to identify the bystander's role. CYBY23 Dateset has the cyberbullying tweets and the reply tweets corresponding to bystanders.

This study focuses on designing and evaluating architectures based on neural networks for automatically identifying bystander behavior in tweets from the dataset CYBY23. To this end, we apply multiple feature extraction methods, namely, sentiments, toxicity, empath, part of speech tags, and TF-IDF, to encode tweet content into a format suitable for model training. By capturing a wide range of features, we aim to improve the discriminative power of our models. Automatic Labeling will help in enhancing the

scalability and usability of the datasets, which will contribute to enhancing the research in this area.

1.1 Objectives of the work

The main objective of this work is to provide an efficient and reliable method for automating the labelling of bystander roles in cyberbullying tweets. To achieve this objective, we will perform the following steps:

- To extract and evaluate various types of features from tweets, namely, Toxicity, Sentiment, Empath, TF-IDF, and POS.
- To apply and compare three neural network-based methods for multiclass classification, assessing their performance in detecting bystander tweets.
- To identify the optimal combination of features that enhances detection accuracy across the implemented models.

The paper is organized as follows. The next section contains the Related work and the background required, followed by section 3 describing the Methodology. The Methodology section describes the dataset, steps involved in pre-processing, and the feature extraction process, followed by the details of neural network models used and the experimental setup. Section 4 gives the results obtained, and Section 5 concludes the paper, along with giving the future scope.

2. Related Work and Background

Identifying bystanders in cyberbullying conversations or threads is very important for understanding cyberbullying and reducing its negative impact. Machine learning and neural network approaches have been widely used to improve detection accuracy in this emerging research area. Bystanders - categorized as defenders, instigators, or neutral participants- significantly influence the outcome of cyberbullying incidents through their responses or lack thereof (Alfurayj and Lutfi, 2023; Alfurayj et al., 2024; Gupta et al., 2024b). When bystander roles are included in cyberbullying detection systems, it improves how accurately and thoroughly cyberbullying is analyzed. Recent studies have successfully deployed machine learning techniques for the classification of bystanders based on their roles (Gupta et al., 2024b).

The CYBY23 dataset comprises Twitter threads with labeled bystander roles, providing a comprehensive view of cyberbullying incidents and valuable data for training and evaluating detection models (Alfurayj et al., 2023b; Gupta et al., 2024b). This dataset allows for better training of machine learning models because it includes the full conversation context, such as replies and interactions between users in a discussion thread, instead of looking at individual tweets separately. Manually labeling bystander roles (having people read posts and mark who is a bystander and what role they play) takes a lot of time and effort. Because of this, there is a need for automated labeling methods (using algorithms or machine learning) that can automatically assign these labels. This would help increase the size of the dataset more easily, which is useful for conducting more extensive research.

Cyberbullying detection, as well as bystander detection in cyberbullying threads, involves various natural language processing (NLP) steps such as pre-processing and feature extraction, before employing machine learning models.

Pre-Processing: It is a crucial step in text processing in order to increase the effectiveness of the machine learning algorithms. The online chats/posts are generally informal, noisy, and may contain URLs, emojis, and hashtags. Researchers have applied standard NLP techniques – such as text cleaning, stemming, tokenization, and lemmatization

for pre-processing social media datasets (Fati et al., 2023). To prepare the dataset, Chatzakou et al. (2019) eliminated noise by removing numbers, stop words, punctuation marks, and filtering out spammers and duplicate entries. Tokenization was then employed, in which the text is divided into smaller components called tokens, such as words, numbers, and punctuation marks. Among them, 'stop words' such as *the*, *a*, *on*, *is*, *all*, and are frequently occurring terms that carry little meaningful content and are often removed during text processing. Stemming is a method used to reduce words to their base or root form. For instance, "booked" becomes "book," and both "working" and "worked" are reduced to "work." Lemmatization, unlike stemming, does not just remove word endings. Instead, it relies on lexical databases to accurately determine the proper base form of a word, for example, converting "saw" to "see" or "better" to "good". Part-of-speech tagging assigns a grammatical category to each word in the document, such as noun, verb, adjective, and others, based on both its meaning and the context in which it appears. Text normalization is done by converting text to a uniform format, such as lowercasing all characters, to ensure consistency (Khairy et al., 2024).

Data augmentation has been widely used to enhance the size and quality of datasets (Gupta et al., 2024a). The study (Risch and Krestel, 2018) employed Data Augmentation using machine translation to increase the size of the input data three times. The hashtags were tokenized into words using the dynamic programming technique. Several methods like random subsampling, random oversampling, SMOTE, and SMOTE + TOMEK have been employed to balance the dataset, thereby tackling the class imbalance issues (Khairy et al., 2024). SMOTE (Synthetic Minority Over-sampling Technique) is applied to overcome the issue of imbalanced data by synthesizing samples for the minority class (Gupta and Gupta, 2024). Borderline-SMOTE (Han et al., 2005) considers only those minority samples that are close to the borderline for synthesizing new samples.

Feature Extraction: To analyze datasets related to cyberbullying and bystanders, features that capture the textual, contextual, and behavioral dimensions of user interactions are extracted. Various studies have utilized feature extraction techniques to enhance datasets (Gupta et al., 2026). Word Embeddings techniques, including FastText, GloVe, and BERT, convert text into numerical vectors that represent semantic meaning (Albraikan et al., 2022). Term Frequency-Inverse Document Frequency (TF-IDF) highlights significant words by taking into account both their frequency and uniqueness (Zotkina and Martyshkin, 2024). Term Frequency (TF) quantifies the number of times a term occurs in a given document, while Inverse Document Frequency (IDF) is computed as the ratio of the size of the document collection to the documents containing the specific term. This measure serves as a good indicator of how unique or rare a word is across all documents in the collection. TF-IDF is computed by taking the product of TF and IDF scores, and denotes the importance or relevance of a term within a document in relation to the entire corpus (Hasan et al., 2023; Lepe-Fa´undez et al., 2021). Researchers analyze the sentiment (emotional tone) of text to help decide the feelings behind a message, which plays a crucial role in identifying instances of cyberbullying and bystander behavior (Zotkina and Martyshkin, 2024). These sentiment features, such as positivity, negativity, and objectivity, have been employed alongside TF-IDF (Tommasel et al., 2018).

Part-of-speech (POS) features analyze the functions of words within sentences (Gupta et al., 2024b). GloVe (Global Vectors for Word Representation) is an unsupervised machine learning approach created at Stanford (Hasan et al., 2023). It generates word embeddings by using a global co-occurrence matrix that tracks the co-occurrence of words based on their frequency of appearing together in a text corpus. In (Davidson et al., 2017), a variety of different features are obtained. Each tweet is transformed to lowercase and stemmed using the PorterStemmer. Further, unigram, bigram, and trigram features are produced and given weights using TF-IDF scores to capture significant patterns. To extract syntactic structures, the NLTK toolkit (Bird et al., 2009) is utilized to create unigram, bigram, and trigram features based on Penn Treebank Part-of-Speech (POS) tags. The Flesch-Kincaid Grade Level method is adjusted using Flesch Reading Ease metrics to evaluate the readability of tweets. Additional features include scores from sentiment analysis, binary indicators, and counts for the existence of hashtags, user mentions, retweets, and URLs, as well as quantitative metrics such as the number of characters, words, and syllables per tweet. The TF-IDF method was applied on n-grams to train classifiers in (Risch and Krestel, 2018). A total of 35 carefully chosen features are utilized - 25 derived from emoticons and 10 reflecting the textual size concerning the number of words, the ratio of uppercase characters to lowercase ones, and counts of negation words along with post polarity. The VADER sentiment analyzer has been employed to extract polarity scores. In the study presented in (Chatzakou et al., 2019), researchers analyzed various features, including network properties, textual characteristics, user attributes, and profile image statistics. Network properties included the number of friends, followers, hub and authority scores, centrality measures, and reciprocity. Textual characteristics examined included metrics such as hashtags, emoticons, word counts, URLs, sentiment scores, and mentions. User attributes considered included average post count, account age, verification status, and subscribed lists. Profile image statistics covered the total number, average, and median sizes, and image locations, along with information from the profile description.

In the recent literature, a range of machine learning and neural network approaches have been utilized by researchers to identify aggression in text, detect cyberbullying, and understand bystander behavior. Classical machine learning algorithms like Logistic Regression, Random Forest, Naïve Bayes, and Support Vector Machines (SVM) are commonly applied in cyberbullying detection tasks (Raj et al., 2021). Additionally, neural network models have also been utilized to analyze and interpret cyberbullying-related conversations.

A Feedforward Neural Network (FNN) represents one of the simplest types of artificial neural networks. In this structure, data flows in a single direction, starting from the input layer, passing through one or more hidden layers, and reaching the output layer. The architecture is acyclic, with no feedback or recurrent connections.

A Multi-Layer Perceptron (MLP) is a basic type of feedforward neural network. It consists of multiple fully connected layers of nodes (neurons), including an input component, several internal processing layers, and an output component. The internal hidden layers perform transformations using weights, biases, and activation functions. The output layer generates probabilistic predictions, making MLPs suitable for both classification and regression tasks. During the training process in MLPs, the backpropagation algorithm is used, which calculates gradients and updates weights accordingly.

Extreme Learning Machine (ELM) is a training algorithm specifically designed for single-layer feedforward neural networks (SLFNs) (Huang et al., 2006). Unlike traditional SLFNs that rely on iterative training methods such as backpropagation and gradient descent, ELM randomly assigns input weights and biases to the hidden neurons and then analytically calculates the output weights. This eliminates the need for adjusting input weights and biases during training, allowing ELM to achieve a globally optimal solution without requiring parameter tuning, such as learning rate or stopping criteria, thereby significantly enhancing training speed (Bhasin and Bedi, 2013). Moreover, ELM has demonstrated strong performance in multiclass classification problems (Bedi and Bhasin, 2019; Huang et al., 2012).

A range of neural network architectures has been investigated for detecting cyberbullying, such as BERT, CNN, LSTM, and hybrid models that integrate these methods (Sandoval et al., 2025). (Tommasel et al., 2018) merged neural networks with Support Vector Machines (SVM) to identify aggression in text. In (Lepe-Fa´undez et al., 2021), the authors combined SVM, Naïve Bayes (NB), and Random Forest (RF) using a majority voting mechanism. The study (Risch and Krestel, 2018) employed an ensemble approach with a recurrent neural network (RNN). A bidirectional Gated Recurrent Unit (GRU) layer, combined with max pooling and average pooling operations, as well as three logistic regression units, is employed alongside Gradient Boosting Decision Trees to evaluate levels of aggression.

Various machine learning models have also been evaluated for their effectiveness in detecting cyberbullying and bystander roles. Techniques such as RF Classifier (Gupta et al., 2024b), BERT, and RoBERTa (Sandoval et al., 2025) have shown promising results. For instance, fine-tuning RoBERTa with oversampled data achieved an F1 score of 89.3%. Models that incorporate bystander roles have demonstrated substantial performance improvements. The classifier chain combining BERT and LSTM networks achieved a 25.35% increase in performance, with an F1-score of 89% (Alfurayj et al., 2024). Robust detection capabilities have been achieved using oversampling techniques and fine-tuning of models like RoBERTa, with high F1-scores (Sandoval et al., 2025). The BBGC model integrates BERT, BiGRU, and CNN to extract deep contextual, sequential, and local features. (Cao et al., 2026). Experimental results demonstrate that this fusion approach significantly outperforms individual pre-trained models and other hybrid neural networks (like RoBERTa and ALBERT) in cyberbullying detection. Further, researchers work on the real-world psychological toll cyberbullying takes on people (Jazzar and Duridi, 2025). This paper uses machine learning and deep learning techniques to identify cyberbullying on different social media platforms.

Evaluation Metrics: As highlighted in the reviewed literature, multiple evaluation metrics have been employed to assess the performance of models in accurately identifying bystander roles. These include class-level and overall weighted averages of recall (rec), precision (prec), F1 score (F1), and the weighted Area Under the ROC Curve (AUC) (Chatzakou et al., 2019). Additionally, researchers have utilized confusion matrices, accuracy, and the Wilcoxon test to further validate the models' effectiveness (Alfurayj et al., 2023b).

3. Methodology

Fig. 1 depicts the methodology followed in the current research. As a first step, research works available on categorizing bystanders into bullying and non-bullying were studied. In this phase, we identified various feature extraction models being used. Then, the corpus from tweets to be used was chosen. The dataset CYBY23, downloaded from Kaggle, is described in detail in subsection 3.1. Data Pre-processing involved data cleaning followed by removing the imbalance

Fig. 1: *Research Methodology*

using the Borderline Synthetic Minority Oversampling Technique (SMOTE) (Han et al., 2005). Pre-processing of the data was done so that it can be used for classification using Neural Network models. Data preprocessing is briefly explained in subsection 3.2. Some of the features were categorical, which were converted to numeric values, and different feature sets were extracted from the data. The details of all the feature sets are given in subsection 3.3. Deployment of three Neural Network models was performed on the Bystanders pre-processed data. Evaluation of the models was on the basis of standard metrics like accuracy, precision, recall, and F1 score. The details of the Neural networks used are given in subsection 3.4. Finally, subsection 3.5 outlines the experimental setup employed in this study.

3.1 Description of the Dataset

The original dataset, CYBY23, was downloaded from Kaggle (Alfurayj et al., 2023b; Alfurayj and Lutfi, 2023). Twitter API was used to extract 1,024 tweets from 150 conversation threads over the span of one year (January 2022 to January 2023). The downloaded information includes the tweet ID, tweet date, the user ID, the user's screen name, likes count, retweets count, and the tweet's text. Keywords and hashtags associated with racial and discriminatory remarks that could lead to harassment were used to gather this data.

The labeling of bystanders was achieved through a manual annotation process based on established guidelines (Alfurayj et al., 2023). This process involved assessing individual tweets' aggressiveness, identifying the role of bystanders in responses, thereby judging the aggressiveness of the whole thread, including the original post and its replies. Only those threads that received agreement from five or more annotators were included for further analysis. This approach reduced the total number of tweets to 639 and the number of tweet threads to 112. The dataset now contains between 10% to 20% instances of bullying, with only 11.6% representing cyberbullying characterized by high aggression. Instigators were prominently represented across both categories of bullying.

The study looked at bystander contagion, which is the idea that people who observe bullying may start behaving similarly. The researchers found a positive correlation - when there were more instigators (people who start or encourage bullying), the number of bullying incidents increased in the dataset. So, the presence of instigators can influence bystanders and lead to more bullying behavior spreading in the conversation. Due to the labor-intensive and time-consuming process of manual role annotation, the researchers emphasized the importance of automating the identification of bystander roles. As a result, they made the CYBY23 dataset publicly available on the Kaggle platform (Alfurayj et al., 2023).

The dataset "CYBY23" consists of 112 threads from Twitter, including main posts as well as replies from bystanders. Each of the 639 tweets is accompanied by its text and various general tweet-level features, along with the labels indicating the roles of bystanders. Additionally, toxicity features were extracted using the Perspective API, while sentiment features were obtained using TextBlob. The dataset comprises 17 attributes, which are listed below:

1. tweet id: An identifier for each tweet.
2. reply id: The identifier of the main tweet to which this reply tweet belongs.
3. created at: The date of tweet creation.
4. text: The text of the tweet.
5. retweet count: Denotes the count of retweets.
6. favorite count: The total number of favorites.
7. class label: It denotes the aggression level of the thread - the main tweet and its replies, on a scale of 0-2. Hence, the main tweet only has this assignment.
    (a) 0 - Aggression
    (b) 1 - Low-aggression bullying
    (c) 2 - High aggression bullying
8. bystander role label: The label assigned to the bystander who has replied to the main tweet. Hence, it was assigned to the reply tweets only.
    (a) 0 - Instigator who agrees with the main post
    (b) 1 - Defender who disagrees with the main post
    (c) 2 - Impartial who is not taking any sides
    (d) 3 - Others
9. Three sentiment features - Polarity, Subjectivity, and Sentiment.
10. Six Toxicity Features - Identity Attack, Insult, Threat, Profanity, Toxicity, and Severe Toxicity.

3.2 Pre-Processing

The original dataset is not suitable for classification by intelligent models in its raw form because it includes URLs, HTML tags, mention tags, chat slang, non-standard English words, and spelling errors. Therefore, preprocessing of the dataset is essential in order to prepare the data for better modelling results. The data cleaning steps carried out on the text attribute are listed below.

1. All the URLs beginning with {https, www}were removed.
2. All the HTML tags contained in <> were removed.
3. All the mentioned tags beginning with @ were removed.
4. The emoji representations were originally separated by {:, }. These delimiters were replaced by the space character, and the separate words were kept since emojis provide useful information on the nature of the tweet.
5. All the punctuation characters except the {?, !}were removed. These two were retained since these punctuation marks can provide cues about the tone or emotion of a sentence. Also, they help in understanding the context and intent of the tweet.
6. Lemmatization was performed to convert the words into their meaningful base forms.
7. Chat words such as LOL were converted to their full text forms.
8. Spell Checker was applied to correct the word spellings.
9. The target variable is the bystander role label. It had four textual values. These textual values were label encoded to values on a scale of 0 to 3 - 0 (Instigator), 1 (Defender), 2 (Impartial), 3 (Others).

High imbalance was observed in the data vis-a-vis the target variable 'the bystander role label' and preliminary experiments showed overfitting. Out of a total of 524 tweets, 228 were labeled as Instigator, 191 as Defender, 96 as Impartial, and only 9 belonged to the Others or Irrelevant category. After the data cleaning process, this imbalance in the dataset was handled using the Borderline-SMOTE technique. This technique identifies the borderline samples and uses them for generating synthetic data using SMOTE. The distribution of classes before and after applying SMOTE is shown in Fig. 2.

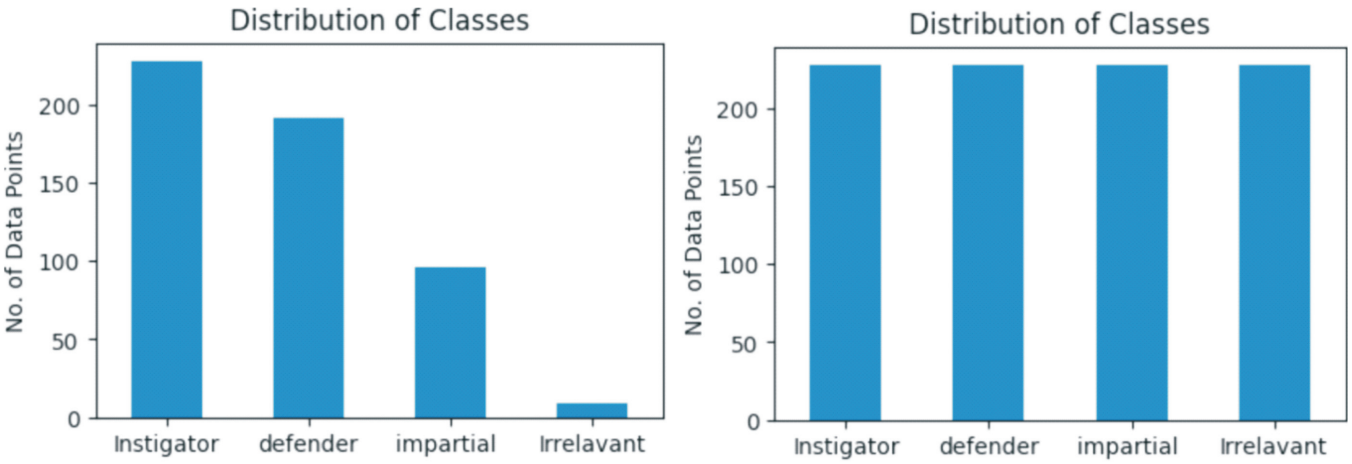

Fig. 2: *Distribution of Bystander Roles in the CYBY23 Dataset before and after Resampling*

3.3 Feature Extraction

After cleaning and preprocessing, various features were extracted from the tweet content. Initially, attributes from both the main tweet and its corresponding reply were merged to assess how the original tweet affects the reply, resulting in the creation of a new enriched dataset. This enriched dataset included:

- 6 general features
- 6 features related to toxicity from main tweet
- 6 features related to toxicity from reply tweet
- 3 features related to sentiments from main tweet
- 3 features related to sentiments from reply tweet
- class label of main tweet.
- bystander role label of reply tweet.

Due to the column-wise merging of main and reply tweets, the total tweet count reduced from 639 to 524. Thus, the updated dataset consisted of 26 attributes and 524 records.

Three features, reply ID, tweet id, and created at, were eliminated since these features had no role in detecting bystanders. Furthermore, the text of the tweet was not considered since its information was already captured through toxicity (via the Perspective API) and sentiment (via TextBlob) features. The finalized dataset comprised 22 features including target label: bystander role model. To enrich the dataset further, three more feature sets extracted through TF-IDF, POS tags, and Empath were incorporated.

- Features derived from Empath Package: The text of both the original tweet and the bystander tweet is used to extract Empath features, as outlined in (Fast et al., 2016). Empath is a tool that analyzes text by looking at the words used (lexical analysis) and placing them into different meaning-based categories. It examines text using 194 predefined categories. These categories are based on research and knowledge about human emotions, behavior, and topics. The categories cover many types of content, such as daily life topics – home, work, family, emotions – anger, sadness, government terms – embassy, democrat, technology terms – iPad, Android, and so on. By grouping words into these categories, Empath helps researchers understand the themes, emotions, and behaviors expressed in a piece of text. For example, the small text "I love you but I hate you also" yields the following values across various categories: 'anger': 0.125, 'anticipation': 0.0, 'disgust': 0.125, 'fear': 0.125, 'joy': 0.125, 'negative': 0.125, 'positive': 0.125, 'sadness': 0.125, 'surprise': 0.0, and 'trust': 0.0.
- TF-IDF Features: The method for computing Term Frequency-Inverse Document Frequency (TF-IDF) features is one of the simplest and most traditional approaches to feature extraction from text, commonly used in Natural Language Processing (NLP). TF-IDF helps create a feature vector for the text by determining the importance of each word based on its frequency in the corpus. In this study, a total of 2,071 features are extracted from the tweets' text.
- Part of Speech (POS) Features: The tweet text is tokenized and categorized into 10 part-of-speech categories. The categories in which all the tokens from text are categorized are: Noun, Adjective, Verb, Adverb, Pronoun, Preposition, Conjunction, Article, Negation terms, and Auxiliary terms. These ten categories form 10 POS features.

To summarize, the final dataset had 2296 features as mentioned in Table 1.

Table 1: *Number of Extracted Features from Tweets*

Number of Features	Tool/API
3	General Features
12	Toxicity attributes (obtained through Perspective API)
6	Sentiment attributes (obtained through TextBlob)
194	Emotion features based on Empath
2071	Features based on TF-IDF
10	Features based on Part of Speech (POS)

Further, these individual feature sets were grouped based on their nature, resulting in two more sets, as shown in Fig. 3.

- Toxicity, Sentiment, and Empath features were concatenated together because they all describe the feelings or emotions in the text. They help understand if the text is angry, kind, rude, or emotional.
- TF-IDF and POS features were grouped into a second set because they tell us about the words and grammar in the text. TF-IDF shows which words are important, and POS tells us about word types like nouns, verbs, etc. These are more about the structure and meaning of the words themselves.

3.4 Neural Network Models

The current research employs three NN models - Feedforward Neural Network (FNN), ELM, and MLP. The configurations of the three models are as follows:

- FNN - The network consists of an input layer with as many neurons as the input features. It is then followed by four dense layers each consisting of 64 hidden neurons and ReLU activation function. Finally, the network consists of an output layer with four units and softmax activation for the four bystander labels. The sparse categorical cross-entropy loss function was employed to handle the multiclass classification task. This function

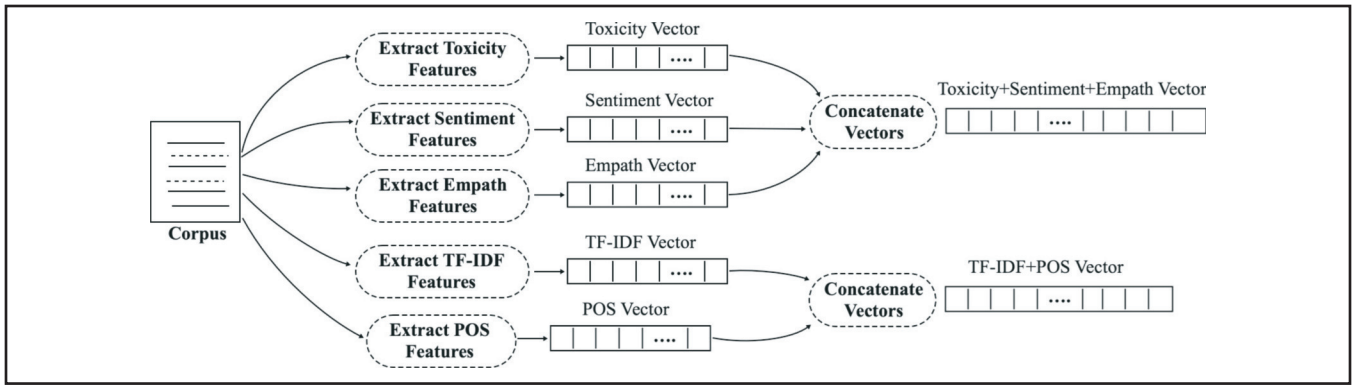

Fig. 3: *Extraction and Concatenation of Features*

is appropriate when integer-labeled target classes are involved. Finally, the model has been optimized with Adam optimizer and accuracy metric has been maximized over 1000 epochs. It is implemented using Keras' Sequential API which uses TensorFlow's backend and provides customizable training (layers, loss function, optimizers). The use of explicit epoch control ensures that the model is fully trained to convergence.

- ELM - As in FNN, the network consists of an input layer with as many neurons as the input features, followed by a single hidden layer with 128 neurons, and an output layer for predictions. The tanh activation function has been used on the hidden neurons.
- The MLP model employs a feedforward structure similar to that of the FNN, beginning with an input layer and followed by four hidden layers, each consisting of 64 neurons that utilize the ReLU activation function. To address multiclass classification, the output layer applies the softmax function. The training process is conducted with the Adam optimizer over a maximum of 1000 iterations. This model is constructed using Scikit-learn's MLP Classifier, which is user-friendly; however, it cannot customize the loss function in the way that Keras does. Additionally, the model may terminate early if convergence criteria are satisfied or if it is progressing slowly, which might result in underfitting.

For improved and stable results, Stratified 5-fold validation has been utilized for dividing the data into training and test sets.

3.5 Experimental Setup

Fig. 4 illustrates the experimental pipeline followed in this study. After extracting features from the corpus, various feature sets are constructed as described in subsection 3.3. These feature sets are then balanced using the Borderline SMOTE technique to address class imbalance. The resulting balanced numeric datasets are fed into three different neural network architectures (explained in subsection 3.4), which are trained and evaluated using stratified 5-fold cross-validation wherein 4 folds are used in training and 1 fold is used in testing for the prediction of a new label. The process of selection of 4 folds for training is repeated 5 times. Performance metrics such as accuracy, precision, recall, and F1-score are analyzed to determine the most effective architecture. The Python library Scikit-learn (Pedregosa et al., 2011) and Keras (Chollet et al., 2015) were utilized for the implementation of the neural network architectures. All experiments were conducted on a system running macOS Big Sur Version 11.3.1 with 8GB of RAM.

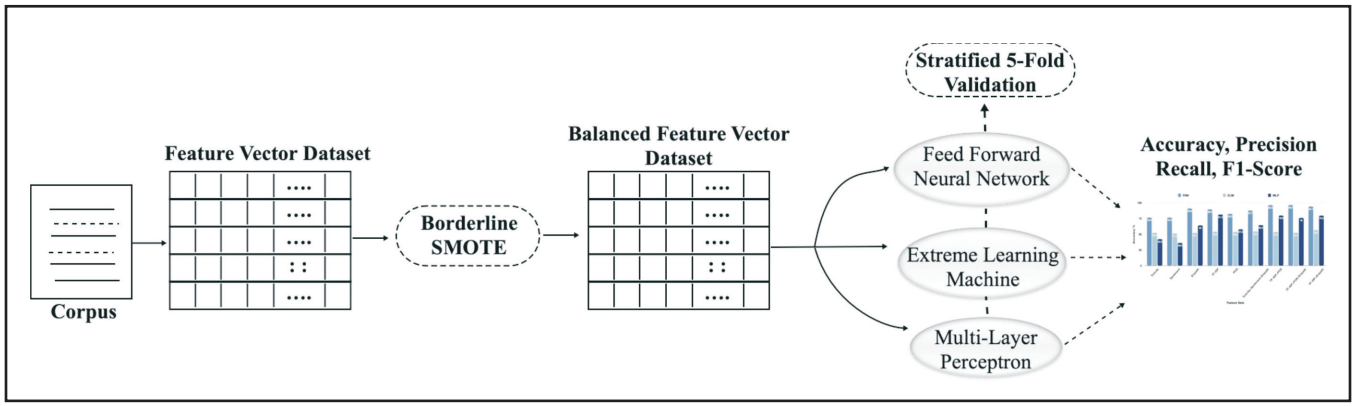

Fig. 4: *Experiment Pipeline*

4. Results

The performance of the three neural networks across all five individual feature sets: Toxicity, Sentiment, Empath, TF-IDF, and POS is presented in Table 2, evaluated using Accuracy, Recall, Precision, and F1-Score. For the four bystander role labels, the precision, recall, and F1-scores are also provided separately. The weighted averages are also reported - weighted precision (WP), weighted recall (WR), weighted F1 (WF). The following observations are evident:

Table 2: *Performance of Neural Networks on Individual Feature Sets*

Feature Set	Models	Accuracy	Precision				WP	Recall				WR	F1-Score				WF
			0	1	2	3		0	1	2	3		0	1	2	3	
Toxicity	FNN	0.7642	0.68	0.71	0.69	0.97	**0.76**	0.67	0.78	0.62	0.99	**0.77**	0.68	0.74	0.65	0.98	**0.76**
	MLP	0.4221	0.31	0.35	0.39	0.58	**0.41**	0.16	0.46	0.40	0.67	**0.42**	0.21	0.40	0.39	0.62	**0.41**
	ELM	0.5241	0.45	0.52	0.39	0.71	**0.52**	0.33	0.35	0.50	0.91	**0.52**	0.38	0.41	0.44	0.80	**0.51**
Sentiment	FNN	0.7654	0.69	0.73	0.66	0.98	**0.77**	0.63	0.77	0.67	0.99	**0.77**	0.66	0.75	0.67	0.98	**0.77**
	MLP	0.3618	0.23	0.41	0.37	0.55	**0.39**	0.34	0.22	0.43	0.46	**0.36**	0.27	0.29	0.39	0.50	**0.36**
	ELM	0.5121	0.41	0.47	0.46	0.63	**0.49**	0.30	0.43	0.41	0.91	**0.51**	0.35	0.45	0.43	0.74	**0.49**
Empath	FNN	0.909	0.91	0.90	0.85	0.99	**0.91**	0.79	0.94	0.92	0.99	**0.91**	0.85	0.92	0.88	0.99	**0.91**
	MLP	0.6404	0.5	0.62	0.57	0.88	**0.64**	0.47	0.68	0.57	0.85	**0.64**	0.49	0.64	0.57	0.86	**0.64**
	ELM	0.5208	0.46	0.47	0.42	0.66	**0.5**	0.30	0.38	0.49	0.91	**0.52**	0.37	0.42	0.45	0.77	**0.5**
TF-IDF	FNN	0.8958	0.95	0.95	0.75	0.99	**0.91**	0.86	0.98	0.94	0.79	**0.89**	0.90	0.97	0.84	0.88	**0.9**
	MLP	0.8125	0.74	0.83	0.72	0.94	**0.81**	0.67	0.87	0.72	0.99	**0.81**	0.70	0.85	0.72	0.96	**0.81**
	ELM	0.5395	0.45	0.49	0.46	0.69	**0.52**	0.32	0.43	0.48	0.92	**0.54**	0.37	0.46	0.47	0.79	**0.52**
POS	FNN	0.8235	0.78	0.82	0.71	0.98	**0.82**	0.72	0.86	0.73	0.98	**0.82**	0.75	0.84	0.72	0.98	**0.82**
	MLP	0.5768	0.54	0.48	0.45	0.79	**0.57**	0.32	0.61	0.41	0.97	**0.58**	0.40	0.54	0.43	0.87	**0.56**
	ELM	0.5362	0.44	0.51	0.44	0.67	**0.52**	0.30	0.43	0.46	0.96	**0.54**	0.36	0.47	0.45	0.79	**0.52**
Toxicity + Sentiment + Empath	FNN	0.8772	0.84	0.84	0.84	0.99	**0.88**	0.79	0.90	0.83	0.99	**0.88**	0.81	0.87	0.84	0.99	**0.88**
	MLP	0.6436	0.54	0.60	0.52	0.88	**0.64**	0.43	0.63	0.55	0.97	**0.65**	0.47	0.61	0.53	0.93	**0.64**
	ELM	0.545	0.48	0.48	0.46	0.68	**0.53**	0.34	0.38	0.51	0.95	**0.55**	0.40	0.43	0.48	0.80	**0.53**
TF-IDF + POS	FNN	0.9627	0.96	0.97	0.92	1	**0.96**	0.90	0.98	0.97	1	**0.96**	0.93	0.98	0.94	1	**0.96**
	MLP	0.7982	0.72	0.82	0.67	0.97	**0.8**	0.64	0.84	0.73	0.99	**0.8**	0.67	0.83	0.70	0.98	**0.8**
	ELM	0.5351	0.45	0.46	0.45	0.69	**0.51**	0.33	0.46	0.39	0.95	**0.53**	0.38	0.46	0.42	0.80	**0.52**

- Across all five individual feature sets, the FNN model consistently outperformed the other two models, achieving the highest values for all four performance metrics. The highest accuracy was achieved on the Empath feature set at 90.9%, followed by 89.58% on the TF-IDF feature set.
- The FNN model achieved the highest precision for classes 0 and 1 using the TF-IDF feature set, class 2 using the Empath feature set, and class 3 with an equivalent precision of 0.99 using both the Empath and TF-IDF feature sets.
- For recall, the FNN performed best on classes 0 and 1 using the TF-IDF feature set and on classes 2 and 3 using the Empath feature set. Additionally, a close recall value of 0.98 for class 3 was achieved on the Toxicity, Sentiment, and POS feature sets.
- FNN attained the highest F1-Score values for classes 0, 1, and 2 using the TF-IDF feature set, and a score of 0.99 for class 3 using the Toxicity, Sentiment, and Empath feature sets. While the MLP model also attained 0.99 as an F1 score on class 3 using the TF-IDF feature set, it underperformed across other classes.

- Both the MLP and ELM models showed poor performance across most cases, with ELM yielding accuracy values close to 50%, while the MLP performed slightly better. Both models performed the worst on the Sentiment feature set and showed their best performance on the TF-IDF feature set.

Additionally, the feature sets were combined, resulting in two new sets: Toxicity + Sentiment + Empath and TF-IDF + POS. The results for these combined feature sets are also presented in Table 2. It is evident from the table that:

- On the Toxicity + Sentiment + Empath feature set, the accuracy of the FNN model dropped to 87.72% when compared to its performance on the standalone Empath feature set. However, MLP and ELM showed slight improvement on the combined set as compared to the individual sets.
- The recall and precision, thus the F1-score value of FNN and MLP, slightly dropped on the Toxicity + Sentiment + Empath feature set when compared to the standalone Empath feature set. However, ELM showed a slight improvement on the combined set.
- On the TF-IDF +POS feature set, significant improvement is seen in all four performance metrics of the FNN model, achieving the highest accuracy of 96.27% across all the feature sets. However, all four measures of MLP and ELM slightly decreased in most of the cases when compared to the individual TF-IDF and POS sets.

Based on the performance of the FNN model across individual and combined feature sets, the highest accuracy was observed with the Empath feature set and the TF-IDF + POS combination. To further explore potential improvements, two additional combinations — TF-IDF +POS+Empath and TF-IDF + Empath were evaluated. The accuracy values of the three NNs over all the feature sets are illustrated in Fig. 5, and the following insights can be drawn:

- A marginal improvement of only 0.2% was observed when using the TF-IDF + POS + Empath feature set compared to TF-IDF + POS. This small gain does not justify the dataset size, as well as the running time of the model being increased. Therefore, TF-IDF + POS remains the more efficient and practical choice.
- A decline in performance by 2.2% was observed for the TF-IDF +Empath feature set compared to TF-IDF + POS, indicating that POS features contribute more significantly to model performance than Empath features in this combination.

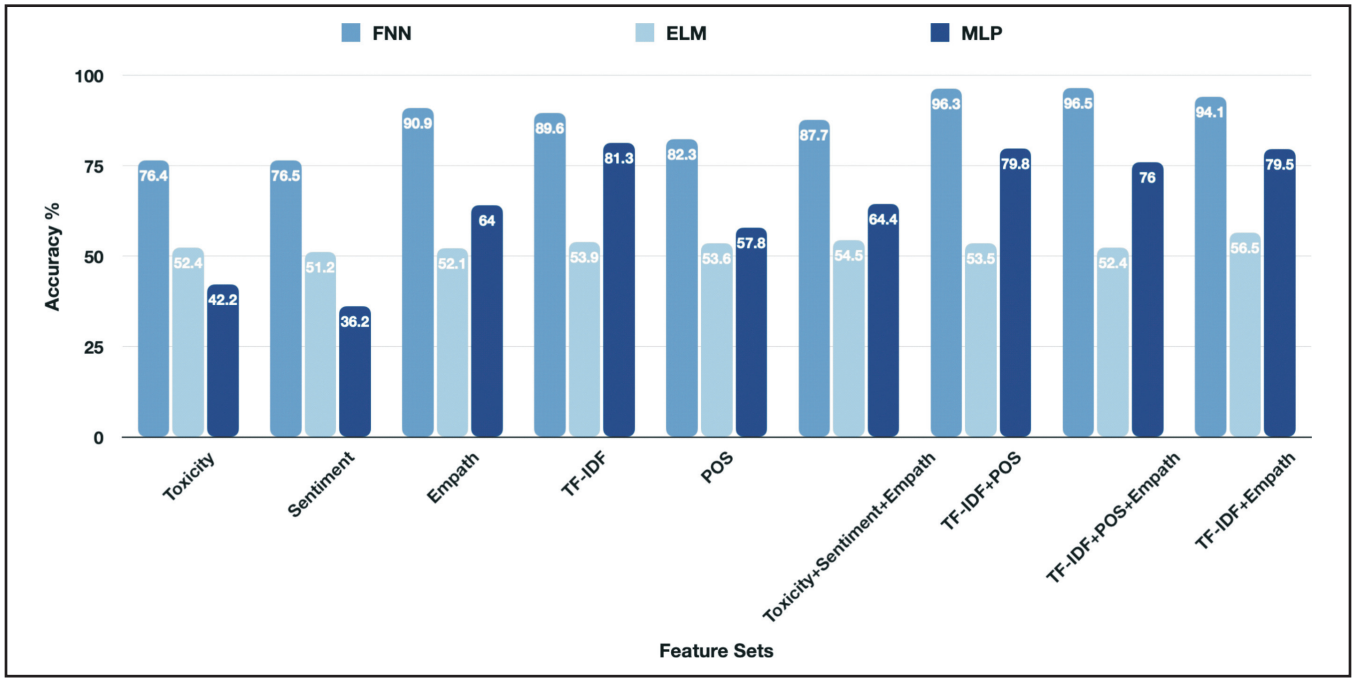

Fig. 5: *Accuracy of the models on different Feature Sets*

In conclusion, the FNN model gives the best overall performance, measured in terms of accuracy, precision, recall, and F1-score, when using the TF-IDF + POS feature set.

The confusion matrices of the FNN model for all five basic feature sets - Toxicity, Sentiment, Empath, POS, and TF-IDF, as well as on the best performing combined feature set TF-IDF +POS, are presented in Figure 6. In each matrix, the bottom-right cell displays the overall classification accuracy (in green) and the misclassification rate (in red). Additionally, the rightmost column shows the recall for each class, while the bottom row indicates the precision for each class.

FNN Performance on Toxicity Features

Actual \ Predicted	Instigator	Defender	Impartial	Irrelevant	Recall
Instigator	153 16.78%	31 3.40%	42 4.61%	2 0.22%	228 67.11% 32.89%
Defender	28 3.07%	177 19.41%	21 2.30%	2 0.22%	228 77.63% 22.37%
Impartial	41 4.50%	41 4.50%	142 15.57%	4 0.44%	228 62.28% 37.72%
Irrelevant	2 0.22%	0 0.00%	1 0.11%	225 24.67%	228 98.68% 1.32%
Precision	224 68.30% 31.70%	249 71.08% 28.92%	206 68.93% 31.07%	233 96.57% 3.43%	697 / 912 76.43% 23.57%

FNN Performance on Sentiment Features

Actual \ Predicted	Instigator	Defender	Impartial	Irrelevant	Recall
Instigator	144 15.79%	34 3.73%	48 5.26%	2 0.22%	228 63.16% 36.84%
Defender	20 2.19%	176 19.30%	31 3.40%	1 0.11%	228 77.19% 22.81%
Impartial	43 4.71%	30 3.29%	153 16.78%	2 0.22%	228 67.11% 32.89%
Irrelevant	2 0.22%	1 0.11%	0 0.00%	225 24.67%	228 98.68% 1.32%
Precision	209 68.90% 31.10%	241 73.03% 26.97%	232 65.95% 34.05%	230 97.83% 2.17%	698 / 912 76.54% 23.46%

(a) Toxicity(b) Sentiment

FNN Performance on Empath Features

Actual \ Predicted	Instigator	Defender	Impartial	Irrelevant	Recall
Instigator	181 19.85%	16 1.75%	30 3.29%	1 0.11%	228 79.39% 20.61%
Defender	5 0.55%	214 23.46%	8 0.88%	1 0.11%	228 93.86% 6.14%
Impartial	12 1.32%	6 0.66%	209 22.92%	1 0.11%	228 91.67% 8.33%
Irrelevant	1 0.11%	2 0.22%	0 0.00%	225 24.67%	228 98.68% 1.32%
Precision	199 90.95% 9.05%	238 89.92% 10.08%	247 84.62% 15.38%	228 98.68% 1.32%	829 / 912 90.90% 9.10%

FNN Performance on POS Features

Actual \ Predicted	Instigator	Defender	Impartial	Irrelevant	Recall
Instigator	165 18.09%	19 2.08%	44 4.82%	0 0.00%	228 72.37% 27.63%
Defender	9 0.99%	196 21.49%	22 2.41%	1 0.11%	228 85.96% 14.04%
Impartial	37 4.06%	22 2.41%	166 18.20%	3 0.33%	228 72.81% 27.19%
Irrelevant	1 0.11%	2 0.22%	1 0.11%	224 24.56%	228 98.25% 1.75%
Precision	212 77.83% 22.17%	239 82.01% 17.99%	233 71.24% 28.76%	228 98.25% 1.75%	751 / 912 82.35% 17.65%

(c) Empath (d) POS

FNN Performance on TF-IDF Features

Actual \ Predicted	Instigator	Defender	Impartial	Irrelevant	Recall
Instigator	197 21.60%	6 0.66%	24 2.63%	1 0.11%	228 86.40% 13.60%
Defender	2 0.22%	224 24.56%	2 0.22%	0 0.00%	228 98.25% 1.75%
Impartial	8 0.88%	5 0.55%	215 23.57%	0 0.00%	228 94.30% 5.70%
Irrelevant	1 0.11%	1 0.11%	45 4.93%	181 19.85%	228 79.39% 20.61%
Precision	208 94.71% 5.29%	236 94.92% 5.08%	286 75.17% 24.83%	182 99.45% 0.55%	817 / 912 89.58% 10.42%

FNN Performance on TF-IDF+POS Features

Actual \ Predicted	Instigator	Defender	Impartial	Irrelevant	Recall
Instigator	205 22.48%	4 0.44%	19 2.08%	0 0.00%	228 89.91% 10.09%
Defender	4 0.44%	223 24.45%	1 0.11%	0 0.00%	228 97.81% 2.19%
Impartial	4 0.44%	2 0.22%	222 24.34%	0 0.00%	228 97.37% 2.63%
Irrelevant	0 0.00%	0 0.00%	0 0.00%	228 25.00%	228 100.00% 0.00%
Precision	213 96.24% 3.76%	229 97.38% 2.62%	242 91.74% 8.26%	228 100.00% 0.00%	878 / 912 96.27% 3.73%

(e) TF-IDF(f) TF-IDF+PO

Fig. 6: *Confusion Matrices of FNN on Different Feature Sets*

The confusion matrices reveal the following important observations:

- Majority of the data points belonging to the Irrelevant category are correctly classified by FNN irrespective of the type of feature vector set used, with the TF-IDF + POS combination achieving perfect classification of all samples in this category.
- Among the individual feature sets, TF-IDF results in the highest number of correctly classified samples for the Instigator, Defender, and Impartial categories. This is followed by Empath, which ranks second, and POS, which ranks third in terms of classification performance for these categories.
- The samples belonging to the Instigator category turn out to be the most challenging to classify accurately. Even with the best-performing feature set, TF-IDF + POS, the FNN model fails to correctly classify 23 samples from this category.

The feature set TF-IDF + POS turns out to be the optimal choice among all, since TF-IDF tells us which words are important or unique in a tweet. It helps the model understand what the tweet is talking about. Also, POS tags show the grammatical role of words, which helps the model learn how the message is constructed. It is useful for understanding the intent and tone of the tweet. Further, both of these individual feature sets provide non-overlapping information and thus result in a more useful feature set combination. Therefore, the FNN model achieves the highest classification performance on the Bystander role dataset when using the combined feature set TF-IDF + POS.

5. Conclusion and Future Scope

This paper presents research on extracting various types of features from tweets related to bystanders. We utilized the dataset CYBY23, which contains tweets from bystanders, and conducted experiments with different combinations of features. A total of 2,284 features were extracted using various tools and APIs, including sentiment features obtained from TextBlob, emotion features from Empath, toxicity features via the Perspective API, TF-IDF features, and features derived from Part of Speech (POS) Tagging. For the

classification task, we employed three types of deep neural network models: a feedforward neural network, a Multilayer Perceptron (MLP), and an Extreme Learning Machine (ELM). To evaluate the models, we employed metrics such as accuracy, precision, recall, and F1-score. The experiments showed that choosing the right features (important characteristics from the text data) is very important for getting good results in a machine learning model. Among all the feature combinations tested, the best performance was achieved when TF-IDF features were combined with POS features. Additionally, the feedforward neural network outperformed all other deep learning models.

In the future, we plan to work with a larger dataset. We also aim to explore multimodal data, including emojis, emoticons, images, and videos. Furthermore, we intend to extend our research to identify bystander roles in tweets that feature multilingual data, incorporating various low-resource languages.

Data Availability Statement

The Cyberbullying Bystander Dataset 2023 that supports the findings of this study is openly available at: https://www.kaggle.com/datasets/haifasaleh/cyberbullying-bystander-dataset-2023.

Conflict of Interest Statement

The authors declare no conflict of interest.

Ethical Declarations

The authors declare that this research does not involve human participants, animals, or any procedures requiring ethical approval.

References

1. Albraikan, A. A., Hassine, S. B. H., Fati, S. M., Al-Wesabi, F. N., Hilal, A. M., Motwakel, A., Hamza, M. A., and Al Duhayyim, M. (2022). Optimal deep learning-based cyberattack detection and classification technique on social networks. *Computers, Materials and Continua*, 72(1):907 - 923.
2. Alfurayj, H., Yee, N. S., and Lutfi, S. L. (2023a). Cyberbullying bystander dataset 2023.
3. Alfurayj, H. S. and Lutfi, S. L. (2023). Exploring bystanders' roles in labeled cyberbullying threads on twitter: A preliminary analysis. In *TENCON 2023-2023 IEEE Region 10 Conference (TENCON)*, pages 1018-1023. IEEE.
4. Alfurayj, H. S., Lutfi, S. L., and Perumal, R. (2024). A chained deep learning model for fine-grained cyberbullying detection with bystander dynamics. *IEEE Access*, 12:105588 -105604. Cited by: 0; All Open Access, Gold Open Access.
5. Alfurayj, H. S., Yee, N. S., and Lutfi, S. L. (2023b). Bystanders unveiled: Introducing a comprehensive cyberbullying corpus with bystander information. In *TENCON 2023-2023 IEEE Region 10 Conference (TENCON)*, pages 1012-1017. IEEE.
6. Bedi, P. and Bhasin, V. (2019). Multilayer ensemble of elms for image steganalysis with multiple feature sets. *International Journal of Computational Science and Engineering*, 20(4):558 - 569.

7. Bhasin, V. and Bedi, P. (2013). Steganalysis for jpeg images using extreme learning machine. In *2013 IEEE International Conference on Systems, Man, and Cybernetics*, pages 1361-1366.
8. Bird, S., Klein, E., and Loper, E. (2009). Natural Language Processing with Python. *O'Reilly Media*.
9. Cao, J., Xiong, Y., Wang, W., & Chen, G. (2026) Cyberbullying Detection Based on Hybrid Neural Networks and Multi-Feature Fusion. *Information*, *17*(2), 205.
10. Chatzakou, D., Leontiadis, I., Blackburn, J., Cristofaro, E. D., Stringhini, G., Vakali, A., and Kourtellis, N. (2019). Detecting cyberbullying and cyberaggression in social media. *ACM Transactions on the Web (TWEB)*, 13(3).
11. Chollet, F. et al. (2015). Keras. https://keras.io.
12. Davidson, T., Warmsley, D., Macy, M., and Weber, I. (2017). Automated hate speech detection and the problem of offensive language. *Proceedings of the International AAAI Conference on Web and Social Media*, 11(1):512-515.
13. DeSmet, A., Bastiaensens, S., Van Cleemput, K., Poels, K., Vandebosch, H., and De Bourdeaudhuij, I. (2019). The efficacy of a serious game on bystander behavior in cyberbullying: A randomized controlled trial. *Computers in Human Behavior*, 57:196-206.
14. Fast, E., Chen, B., and Bernstein, M. S. (2016). Empath: Understanding topic signals in large-scale text. In Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems, CHI '16, page 4647-4657, New York, NY, USA. *Association for Computing Machinery*.
15. Fati, S., Muneer, A., Alwadain, A., and Balogun, A. (2023). Cyberbullying detection on twitter using deep learning-based attention mechanisms and continuous bag of words feature extraction. *Mathematics*, 11:3567. Gini, G., Pozzoli, T., Sala, M., and Elia, N. (2014). The role of empathy in the bystander effect in bullying situations. *Aggressive Behavior*, 40(6):522-533.
16. Gupta, A., Chug, A., and Singh, A. P. (2024a). Multi-disease identification in single citrus leaf image using sigmoid activation and optimised threshold. In 2024 11th International Conference on Signal Processing and Integrated Networks (SPIN), pages 285-290. IEEE.
17. Gupta, A., Chug, A., and Singh, A. P. (2026). Feature extraction in sensor plant disease datasets using reformed membership functions independent of class variables. Scientific Reports, 16(1):4115.
18. Gupta, A. and Gupta, S. (2024). Enhanced classification of imbalanced medical datasets using hybrid data-level, cost-sensitive and ensemble methods. International Research Journal of Multidisciplinary Technovation, 6(3):58-76.
19. Gupta, A., Thakkar, K., Bhasin, V., Tiwari, A., and Mathur, V. (2024b). Bystander detection: Automatic labeling techniques using feature selection and machine learning. International Journal of Advanced Computer Science and Applications, 15(1):1135 - 1143. Cited by: 0; All Open Access, Gold Open Access.
20. Han, H., Wang, W.-Y., and Mao, B.-H. (2005). Borderline-smote: a new oversampling method in imbalanced data sets learning. In *International conference on intelligent computing*, pages 878-887. Springer.
21. Hasan, M. T., Hossain, M. A. E., Mukta, M. S. H., Akter, A., Ahmed, M., and Islam, S. (2023). A review on deep-learning-based cyberbullying detection. Future Internet, 15(5).
22. Hawkins, D. L., Pepler, D. J., and Craig, W. M. (2019). *Naturalistic observations of peer interventions in bullying. Social Development,* 10(4):512-527.
23. Huang, G.-B., Zhou, H., Ding, X., and Zhang, R. (2012). Extreme learning machine for regression and multiclass classification. *IEEE Transactions on Systems, Man, and Cybernetics*, Part B (Cybernetics), 42(2):513-529.
24. Huang, G.-B., Zhu, Q.-Y., and Siew, C.-K. (2006). Extreme learning machine: Theory and applications. Neurocomputing, 70(1):489-501. *Neural Networks*.
25. Huang, Y. Y. and Chou, C. (2020). An analysis of multiple factors of cyberbullying among junior high school students in Taiwan. *Computers in Human Behavior*, 73:221-228.
26. Jazzar, M., Duridi, T. (2025). A Comprehensive Review of Machine Learning and Deep Learning Techniques for Cyberbullying Detection. In: Poonia, R.C., Sharma, S., Hameed, I.A., Upreti, K. (eds) Smart Cyber Physical Systems. ICSCPS 2024. *Smart Innovation, Systems and Technologies*, vol 435. Springer, Singapore.
27. Khairy, M., Mahmoud, T. M., and Abd-El-Hafeez, T. (2024). The effect of rebalancing techniques on the classification performance in cyberbullying datasets. *Neural Computing and Applications*, 36(3):1049 - 1065. Kowalski, R. M., Giumetti, G. W., Schroeder, A. N., and Lattanner, M. R. (2020). Bullying in the digital age: A critical review and meta-analysis of cyberbullying research among youth. *Psychological Bulletin*, 140(4):1073-1137.
28. Lepe-Fa´undez, M., Segura-Navarrete, A., Vidal-Castro, C., Mart´ýnez-Araneda, C., and Rubio-Manzano, C. (2021). Detecting aggressiveness in tweets: A hybrid model for detecting cyberbullying in the spanish language. *Applied Sciences*, 11(22).
29. McMahon, S. D., Florin, P., and Kivlighan, D. M. (2014). Prevention of bullying and victimization through bystander education and intervention. *School Psychology Review*, 43(4):510-527.
30. Obermaier, M., Fawzi, N., and Koch, T. (2016). Bystanding or standing by?how the number of bystanders affects the intention to intervene in cyberbullying. *New Media and Society*, 18(8):1491-1507.

31. Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825-2830.
32. Raj, C., Agarwal, A., Bharathy, G., Narayan, B., and Prasad, M. (2021). Cyberbullying detection: Hybrid models based on machine learning and natural language processing techniques. *Electronics (Switzerland)*, 10(22). Cited by: 80; All Open Access, Gold Open Access, Green Open Access.
33. Risch, J. and Krestel, R. (2018). Aggression identification using deep learning and data augmentation. In Kumar, R., Ojha, A. K., Zampieri, M., and Malmasi, S., editors, Proceedings of the First Workshop on Trolling,

❖ ❖ ❖