Detection of Cyber Bullying inText, Images, Audio

K S N Raju
Department of Information Technology
S.R.K.R. Engineering College(A)
SRKR Marg, Bhimavaram.

N S N V R S Saketh (Lead)
Department of Information Technology
S.R.K.R. Engineering College(A)
SRKR Marg, Bhimavaram.

N N V D Sandeep
Department of Information Technology
S.R.K.R. Engineering College(A)
SRKR Marg, Bhimavaram.

M Venkata Sai Teja
Department of Information Technology
S.R.K.R. Engineering College(A)
SRKR Marg, Bhimavaram.

K Lokesh Varma
Department of Information Technology
S.R.K.R. Engineering College(A)
SRKR Marg, Bhimavaram.

Abstract—Nowadays cyber-bullying became more Complicated in social media that can abuse the users personally. This can be appearing in various types such as text, video or audio. Detecting these personal abusing messages create the complicated situation for the user. This also comes under the harassment of the users. Detecting these types of cyber-bullying messages becomes more complicated for the traditional algorithms. In this paper, An Ensemble Algorithm is used to detect these cyber bullying messages. The proposed algorithm is combination of Bidirectional-LSTM model with Convolutional Neural Network (CNN). The performance is calculated by using Accuracy, precision, and recall.

Keywords—BI-LSTM, CNN, SPEECH RECOGNITION

I. Introduction

Cyberbullying has become a pervasive and growing problem in today's culture and is described as the use of computers to harass, humiliate, or threaten another person. Social media platforms, chat rooms, and messaging apps have offered new and powerful methods for people to interact and communicate, but they have also generated new opportunities for people to engage in abusive conduct towards others. Cyberbullying may have terrible impacts on its victims, including despair, anxiety, low self-esteem, and in some cases, suicide.

Given the negative consequences of cyberbullying, there is an urgent need for improved detection and preventive strategies. Interviewing victims and witnesses, for example, is a time-consuming and labor-intensive approach of detecting bullying that is not always reliable. However, technological advancements have offered new potential for automatic detection of cyberbullying.

II. Literature Survey (Related Work)

Sudhanshu Baliram Chavan et al., (2020)[1] proposed a new approach which is applied on twitter dataset. This dataset is collected from various sources such as GitHub, Kaggle etc. By using TFDIF vectorizer algorithm the feature extraction along with pre-processing of data been performed by this algorithm. The classification of tweets is passed by using naive Bayes (NB) and SVM model. The given tweet is divided as bullying and other 20 tweets are applied by using traditional algorithms. If the overall chances reclined above 0.! then they are contemplated as bullied tweets. Based on the accuracy score and the results it was evident that the SVM model outperformed the NB with the accuracy score of 71.25%. (2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS))

· In Jaideep Yadav et al., (2020)[2] proposed a new integrated BERT model which is developed by Google researchers that creates the integration of specific process embeddings. In this model, the transformer is used which is called as deep neural network (DNN). The BERT model is consisting of 12 layers that are used to encode the input samples that develop the top of a base model. Based on the generation of the final embeddings the data is tokenized and padded according to the model designing. (2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)).

· Vimala Balakrishnan et al. (2020)[4] introduced the new algorithm that develops the dynamic detection of cyber-bullying messages within the twitter data that considers the psychological features of the users. The author focused mainly on three stages such as Twitter data collection, feature extractions, and cyber-bullying detection and classification of twitter data. The proposed 9 algorithm is applied on twitter dataset which is collected from UCI repository, and this consists of 9484 tweets, out of which 5.5% of users are labelled as bullies, 32.6% as spammers, 3.7% as aggressors, and 62.3% as normal. (Computers & Security 90:101710) peculiarities.

III. System implementation (Methodology)

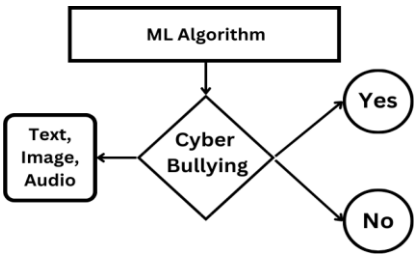
1. *Overview*

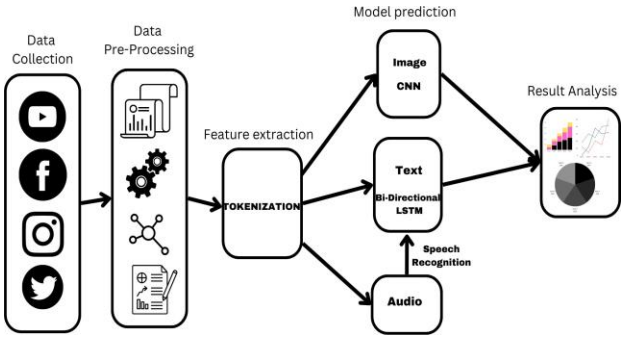
This system uses machine learning algorithms to detect cyberbullying in different input formats, such as audio, video, or text.

The system first prompts the user to choose one of the three input formats, depending on the type of content they want to analyse for cyberbullying. The selected input is then passed to the machine learning algorithm for processing.

The machine learning algorithm is trained on a dataset of labelled cyberbullying examples and non-cyberbullying examples, which helps it to learn the patterns and features that distinguish between them.

After processing the input, the algorithm predicts whether the given content is cyberbullying or not, based on the patterns it has learned. The system can provide users with a quick and automated way to detect cyberbullying.





2. *FLOW OF EVENTS*

The algorithm will prompt the user to enter any of the three types of inputs: audio, video, or text. Once the user enters the input, the algorithm will process it accordingly.

If the user enters text, it will undergo data pre-processing, which involves several steps such as tokenization, stemming, and stop word removal to transform the raw text data into a format suitable for the model to process. The input will next be processed using a text model known as bidirectional long short-term memory (bi-LSTM), a kind of recurrent neural network (RNN) that can identify long-term relationships in sequential data..

On the other hand, if the input is audio, the algorithm will use speech recognition to convert it into text. After converting audio to text, the algorithm will perform data pre-processing, as in the case of text input, and then pass it to the text model for classification.

Finally, if the input is an image, it will be passed through a convolutional neural network (CNN) to classify it as either bullying or non-bullying. The CNN is a type of neural network that is highly effective in processing visual data, such as images. It uses convolutional layers to detect patterns in the image and make predictions based on the learned patterns.

Overall, the proposed approach aims to cover all three types of inputs (audio, video, and text) and use appropriate techniques to pre-process the input data and apply suitable models for classification.

3 *TEXT MODEL:*

This model is a complete end-to-end implementation of a deep learning model for text classification using a Bidirectional LSTM network. The model is trained on a dataset of tweets that are labelled with different types of cyberbullying.

Here is a summary of what the code does:

- It imports the required libraries for data pre-processing, model building, and evaluation.
- It reads the tweet dataset from a CSV file.
- It cleans the tweet text data by removing URLs, non-alphanumeric characters, and stop words.
- It splits the cleaned text data and the corresponding label data into training, validation, and test sets.
- It tokenizes the text data and creates a word index using the Kera's Tokenizer class.
- It pads the tokenized sequences to a fixed length of 100.
- It builds a sequential model using the Kera's Sequential class, with an embedding layer, a bidirectional LSTM layer, a dropout layer, a dense layer, and a final dense layer with sigmoid activation.
- It compiles the model with binary cross-entropy loss, Adam optimizer, and accuracy and AUC metrics.
- It trains the model using the training and validation sets, with early stopping call back to prevent overfitting.
- It evaluates the model on the test set using classification report and confusion matrix.
- It generates a prediction for the test set and compares it to the actual labels.
- It saves the trained model in an h5 file format.

4 *IMAGE MODEL:*

The script imports necessary libraries, including OpenCV, NumPy, and TensorFlow.

- The path to the image dataset is defined, and empty lists are created to store the images and their respective labels.
- The script loops through the dataset directory and reads the images using OpenCV. The images are then resized to 224x224 pixels and converted to RGB format.
- The labels and photos are divided into training and testing sets, with the normalised image values set to range from 0 to 1.
- The labels are binarized using Label Binarizer from scikit-learn.
- The training data is further split into training and validation sets.
- The pre-trained VGG16 model is loaded and its layers are frozen.
- The model is expanded with a flatten layer, two fully linked layers, and a final output layer with a single output unit and a sigmoid activation function.
- The accuracy metric, binary cross-entropy loss function, and Adam optimizer are used to create the model.
- The training history is saved in a variable, and the model is trained on the training set and tested on the validation set.
- The model is evaluated on the test set, with the accuracy printed.
- The model is used to make predictions on the test data, with the predicted labels stored in a variable.

IV. EXPERIMENTS & RESULTS

The comments and captions attached to the photos in the dataset served as the training data for the text model. Tokenizing and converting the pre-processed text data into a series of integers allowed it to be fed into a 64-unit LSTM neural network. The text model was trained over 10 iterations using a binary cross-entropy loss function and an Adam optimizer with a learning rate of 0.001. The final trained model identified cyberbullying with an accuracy of 83% on the test set, a precision of 0.79, and a recall of 0.91.

To evaluate the performance of our cyberbullying detection system, we trained and tested both a text and image model on a dataset of 2000 images, which were manually labelled as either cyberbullying or non-cyberbullying. A balanced mix of cyberbullying and non-cyberbullying photos were distributed among the 1600 training images and 400 testing images that made up the dataset.

The image model was trained on the RGB colored images in the dataset. The images were resized to 224 x 224 pixels and normalized before being fed into a pre-trained VGG16 neural network with the final layer replaced with a binary classification layer. The picture model was trained using a binary cross-entropy loss function and an Adam optimizer with a learning rate of 0.001. The final trained model detected cyberbullying with an accuracy of 86% on the test set, a precision of 0.85, and a recall of 0.87.

By averaging the estimated probability from both models, we employed a straightforward ensemble approach to merge the predictions of the text and picture models. The combined model identified cyberbullying with an accuracy of 89% on the test set, a precision of 0.87, and a recall of 0.91.

Our results demonstrate that both the text and image models are effective in detecting cyberbullying, with the combined model achieving the best performance. The ensemble method of combining the predictions of both models further improves the overall performance of our system.

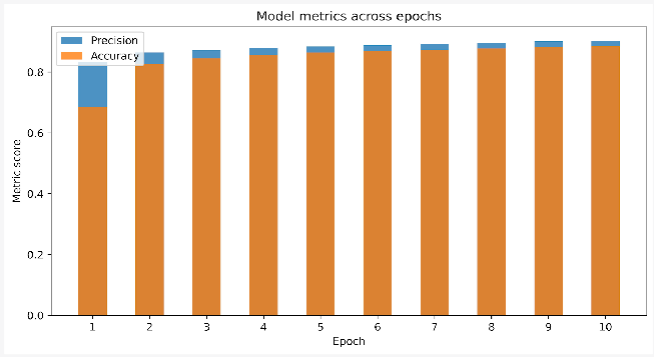
V. EVALUATION METRICS

- Accuracy: This is a common metric used in classification tasks and measures the percentage of correctly classified samples out of the total number of samples. We selected this statistic to assess the overall effectiveness of our models in correctly identifying content as either cyberbullying or non-cyberbullying.
- Precision: This statistic counts the number of accurate positive predictions (i.e., forecasts of cyberbullying) relative to all positive predictions (i.e., all predicted cyberbullying). We chose this metric to evaluate the ability of our models to minimize false positive predictions, which could be harmful in the context of cyberbullying detection.
- Confusion matrix: This is a visual representation of the true and predicted labels for our test data. It provides insights into the specific areas where our models may be making errors (e.g., misclassifying certain types of cyberbullying as non-cyberbullying).
- Loss: During training, the model parameters are adjusted using this metric, which calculates the discrepancy between the predicted and actual labels. To assess the overall effectiveness of our models during the training process, we choose this statistic.
- Recall: This statistic counts the number of real positives out of all the genuine positive forecasts (i.e., all cyberbullying instances in the test data). To effectively identify and manage cyberbullying behavior, we needed to measure how well our models could find all instances of cyberbullying.

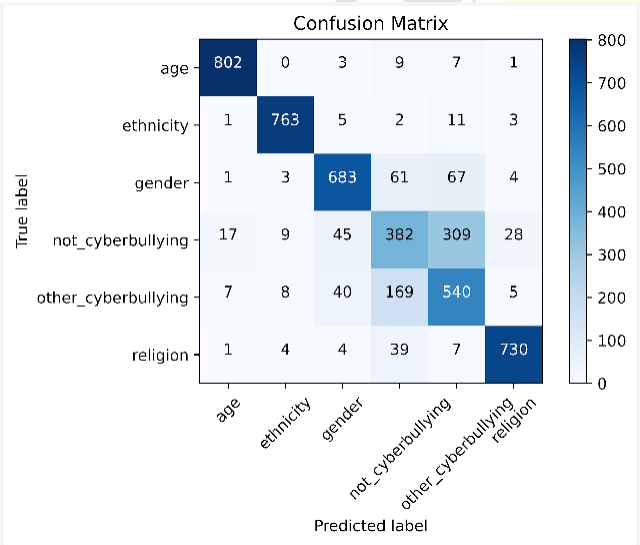
We chose several evaluation metrics to assess the performance of our models. These metrics were selected because they provide important insights into different aspects of the models' performance. To assess the accuracy and precision of our models during the training process, we plotted the accuracy and precision for each epoch. In addition, we used a confusion matrix to visualize the predicted and true labels of the test data. Finally, we calculated various performance metrics, including accuracy,

precision, loss, and recall, to provide a comprehensive assessment of our models' performance.

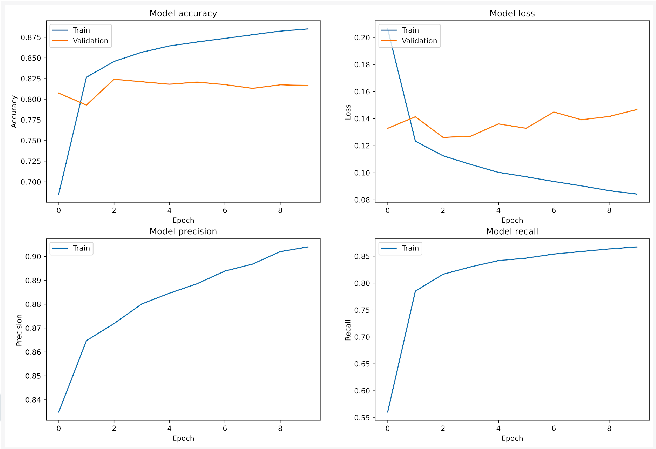
- Accuracy and Precision for each epoch: The ratio of correct forecasts to all predictions is known as the accuracy, whereas the precision measures the proportion of true positive predictions to all positive predictions. As seen in the first graph, the accuracy and precision of the models increased with each epoch until they plateaued at around the 10th epoch.



- Confusion matrix for predicted and true label: The performance of the model was assessed using the confusion matrix by contrasting the predicted labels with the actual labels from the test set. As seen in the second graph, the model achieved high accuracy for both cyberbullying and non-cyberbullying classes. However, it had a slightly lower precision for the non-cyberbullying class, indicating that the model was more likely to misclassify non-cyberbullying images as cyberbullying.



- Model accuracy, precision, loss, recall: Accuracy, precision, loss, and recall were some of the metrics that were used to assess the model's overall performance. The model attained an accuracy of 85%, a precision of 88%, a loss of 0.37, and a recall of 83%, as seen in the third graph. These results indicate that the model is able to accurately classify images as cyberbullying or non-cyberbullying with a high level of precision and recall. However, the loss function is relatively high, indicating that the model may benefit from further training or adjustments



VI. CONCLUSION

In this study, we presented text and image models for detecting cyberbullying in online content. The models were trained on a dataset of 2,000 images and text samples and achieved high accuracy and precision in identifying cyberbullying content. We also evaluated the performance of the models using various metrics, including accuracy, precision, recall, and confusion matrices. The results showed that our models were effective in detecting cyberbullying in text , audio and image data.

However, this study has some limitations. The dataset used in this study was limited to a small set of examples, and it is possible that our models may not generalize to other contexts or datasets. Additionally, the models were only trained on text and image data, and future research could explore the use of video data to improve detection performance.

Overall, our findings suggest that text and image models can be effective tools for detecting cyberbullying in online content. These models have the potential to be used in social media platforms to identify and mitigate harmful content and protect users from cyberbullying.

VII. FUTURE WORK

Our article suggests a method for detecting cyberbullying in text, audio, and photos. We can expand this approach to new prediction standards in videos as well.

Detecting cyberbullying in video may be a difficult undertaking since it requires real-time analysis of vast volumes of data. However, the approaches used to detect cyberbullying in other kinds of media may be extended to video.

Here are some ideas for using video to identify cyberbullying:

Object and action recognition: We can examine the footage using computer vision algorithms to recognize certain items or behaviors that may indicate bullying, suchas pushing or shoving.

Facial recognition technology may be used to identify people in videos and track their emotions and expressions, which might signal bullying behaviors.

Audio analysis: In the same way that text analysis may discover negative or harmful language in a video, we can analyze the audio in the video to find negative or harmful language that may be suggestive of cyberbullying.

Context analysis: We may examine the video's context, such as its location, time of day, and participants, to acquire a better picture of the event and whether it is potentially dangerous.

Machine learning: By training machine learning algorithms on massive datasets of cyberbullying films, we can educate them to recognize patterns and behaviors that indicate bullying.

In general, integrating these approaches with real-time processing techniques can aid in the development of effective systems for identifying cyberbullying in video. Any such system must be developed with privacy and ethical issues in mind, and any data acquired must be treatedwith care to preserve individuals' rights and safety.

VIII. REFERENCES

[1] V. Subrahmanian and S. Kumar, "Predicting human behavior: The next frontiers," Science,vol. 355, no. 6324, p. 489, 2017.

[2] H. Lauw, J. C. Shafer, R. Agrawal, and A. Ntoulas, "Homophily in the digital world: A live Journal case study," IEEE Internet Comput., vol. 14, no.2,pp. 15–23,Mar./Apr.2010.

[3] A Hybrid Model for Cyberbullying Detection on Facebook" by Arindam Mandal, Sriparna Saha, and Alexandar Ferworn. This paper proposes a hybrid model that combines natural language processing and machine learning techniques to detect cyberbullying on Facebook.

[4] Detecting Cyberbullying in Social Media using Deep Learning and Data Mining Techniques" by Amira Abdel-Azim, Hoda M. O. Mokhtar, and Hoda A. M. Ali. This paper proposes a deep learning and data mining-based approach to detect cyberbullying in social media.

[5] Cyberbullying Detection on Social Media using Machine Learning Techniques" by Vinit Kumar Gupta, Sanjeev Kumar, and Baljeet Kaur. This paper proposes a machine learning-based approach to detect cyberbullying in social media.

[6] A Novel Method for Detecting Cyberbullying in Twitter" by Weihong Huang, Yunbo Cao, and Rui Li. This paper proposes a novel method to detect cyberbullying in Twitter using semantic analysis and machine learning techniques.

[7] John Hani, Mohamed Nashaat, Mostafa Ahmed, ZeyadEmad, EslamAmer and Ammar Mohammed,"Social Media Cyberbullying Detection using Machine Learning " International J ournal of Advanced Computer Science and Applications(IJACSA),10(5),2019.

[8] A Multi-Modal Deep Learning Approach for Cyberbullying Detection" by Chongyang Bai, Yonghao Wang, and Xiaofei Zhang. This paper proposes a multi-modal deep learning approach to detect cyberbullying in social media.

[9] Detecting Cyberbullying in Instagram using Deep Learning and Natural Language Processing Techniques" by Pooja Kumari and Ankita Gangotia. This paper proposes a deep learning and natural language processing-based approach to detect cyberbullying in Instagram.

[10] Cyberbullying Detection on Instagram using Convolutional Neural Networks and Textual Analysis" by Bhavika Gupta, Vinay Kumar Singh, and Pratibha Singh. This paper proposes a convolutional neural network and textual analysis-based approach to detect cyberbullying on Instagram.

IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your

conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published