An Intelligent Hate Speech Detection System For Safetalk Using Bidirectional LSTM

Akash R Nair[1]
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Ashik Das TH[3]
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Arathy Krishna[2]
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Diya Merin Babu [4]
B-tech Student
Computer Science &Engineering
Mangalam Colleg Of Engineering

Anjana Sekhar[5]
Assistant Professor
Computer Science &Engineering
Mangalam Colleg Of Engineering

***Abstract*— Hate speech is clearly directed at social tensions and violence. Recognition becomes increasingly difficult when emotions overlap. However, there are still some unsolved problems with informal and indirect targeting of negative communication, such as sarcasm, misrepresentation, and glorification of immoral behavior of the target audience or society. In this study, we proposed a case selection method based on thevisualization of attention networks. The purpose is to classify, modify, and scale the number of training instances. To do this, we first used hate speech dictionaries and online forums to practice embedding using transfer learning. We then used synonym expansion of the semantic vector. An active learning approach was used to train the task using pairs of outcome labels. Entropy- based selection and visualization techniques help select unlabeled text for each active learning cycle. To improve model accuracy, the approach is improved and the number of training instances is increased. The active learning cycle repeats until all unlabeled text is converted to labeled text. Semantic embedding and lexicon expansion improve the receiver operating characteristic (ROC) ofthe model from 0.89 to 0.91. A bidirectional LSTM with attention and active learning scored 0.90 on Precision – Recall. A trained model can visualize position-weighted terms to explain why hate speech is classified**

Keywords—Deep learning, ethnic hate, explainable machine learning (ML), hate speech detection.

I. INTRODUCTION

With the advent of new communication technologies, the communication process has become very easy for internet and social network users all over the world. Remarkable advances in technology are driving the adoption of new media. With the increasing use of social media, the phenomenon of online hate speech is also gaining attention. In recent years, social media platforms, such as Twitter and Facebook, have gained popularity among the masses. They are filled with user-generated content, including text, social media data, photos, and videos. Given the large amount of user-generated content on the Internet, especially on social media, it is becoming increasingly important to identify and potentially prevent the transmission of hate speech, i.e. fight against racism and sexism. With the vast amount of user-generated information on the Internet, especially on social media, it is becoming increasingly important to identify and potentially limit the spread of hate speech. Hate speech and defamatory comments against another person's religion, ethnic origin, or sexual orientation are prohibited by law. In many countries, anyone who incites violence or genocide is considered a criminal. In addition, many governments prohibit the use of symbols of totalitarianism and restrict freedom of assembly in the case of fascism or communism. However, not everyone has equal access to this public space and not everyone has the right to express themselves without fear. Hostile and disrespectful communication on the Internet drowns out the voices of marginalized and underrepresented groups in the public conversation. This helps us to understand this and mitigate .

Hate speech on the Internet and social media not only causes friction between groups of people, but it can also causeharm businesses and cause really important problems. For these reasons, websites such as Facebook, YouTube, and Twitter are limited hate speech. However, tracking and filtering all content always problems. For this reason, many tests have been conducted to learn how to automatically detect hate speech. Most of this hate speech detection work attempts to create dictionaries of hate phrases and expressions or categorize hate speech into two categories: "hate" and "don't hate". However, assessing whether a sentence contains hateful content is always difficult, especially when hate speech is masked under sarcasm or when hate is not clearly expressed. race or prejudice. The goalof this study was to extract hate speech from social media content in an online forum. We have proposed a hate speech visualization and recognition system based on the deep attention technique. In a study of online trends, users communicated hate speech in response in the

Communication or an online system during the pandemic.

II. RELATED WORK

Deep learning techniques have proven to be very effective in classifying hate speech. The performance of deep learning-based approaches has outperformed classical machine learning techniques such as support vector machines (SVMs), gradient-enhanced decision trees (GBDTs), and logistic regression. Among the deep learning-based classifiers, convolutional neural networks (CNNs) record local patternsin the text. A long-term memory model (LSTM) based on a Deep Learning Model or Gated Recurrent Unit (GRU) captures it.

[1] Deep learning to detect hate speech in tweets: Detecting hate speech on Twitter is essential for applications such as extracting controversial events, creating AI chatbots, content recommendations, and sentiment analysis. We define this task as being able to classify a tweet as racist, sexist ornot. The complexity of natural language constructs makes this task very difficult. We perform extensive tests with several deep learning architectures to learn how to embed semantic words to address this complexity. Our tests on a benchmark dataset of 16,000 annotated tweets show that such deep learning methods outperform modern character/word n-gram methods by about 18 F1. With the dramatic increase in social interactions on online social networks, there has also been an increase in hate activity aimed at exploiting theseinfrastructures. On Twitter, hate tweets are tweets containing abusive language aimed at individuals (online followers, politicians, celebrities, products) or specific groups (a country, LGBT, etc.) religion, gender Such hate speech detection is important to analyze the overall sentiment of one group of users towards another and to prevent related illegal activities.

[2] CNN on hate speech and identifying offensive content in Hindi: describes the best group solution for task 1 for Hindi in HASOC competition organized by FIRE 2019. The mission is to identify hate speech and offensive language in Hindi. Specifically, it's a binary classification problem where a system has to classify tweets into two classes:
(a) hateful and offensive (HOF) and (b) not hateful or offensive (NOT). Contrary to the popular idea of pre-training word vectors (aka word embedding) with a large corpus of corpus from a common domain such as Wikipedia, here we have used a relative collection small relevant tweets to practice in advance. Here, they trained a convolutional neural network (CNN) on pre-trained word vectors. This approach allows us to be ranked first for this mission out of all the teams. Otherwise it is labeled as NO. There has been significant research on hate speech and the identification of offensive content in several languages, particularly in English [3,2,6,25,24]. However, there is a lack of work in most other languages. The proposed method is based on very little preprocessing and feature engineering compared to many existing methods.

[3] Development of an online hate classifierfor multiple social media platforms: The growth of social media allows people to express their emotions.At the same time, however, it leads to the emergence of conflict and hatred, making the online environmentunattractive for users. Although researchers have found that hate is a cross-platform problem, there is a lack of online hate detection models that use cross-platform data. To fill this research gap, we are collecting a total of 197,566 reviews fromfour platforms: YouTube, Reddit, Wikipedia and Twitter, with80% of comments labeled as non-hate and the remaining 20% labeled as hateful. Then we test some classification algorithms (Logistic Regression, Naïve Bayes, Support Vector Machine, XGBoost and Neural Network) and feature representation (Bag-of-Words, TF-IDF, Word2Vec, BERT and their combinations). Although all models significantly outperformed the benchmark keyword-based classifier, XGBoost using all the features would perform best (F1 = 0.92). Feature importance analysis indicated that BERT features had the most impact on predictions.

[4] Hate me don't hate me: Detect hate speech on Facebook: While promoting communication and facilitating information sharing, social networking sites are also used to launch harmful campaigns against specific groups and individuals. Cyberbullying, inciting self-harm, sexual assault are just some of the serious effects of large-scale online attacks. In addition, attacks can be made against groups of victims and can escalate into physical violence. In this work, we aim to prevent and stop the alarming spread of such hate campaigns. Using Facebook as a benchmark, we looked at the text content of comments that appeared on a set of Italian public pages. First, we introduce multiple hate categories to distinguish the type of hate. Discovered comments are then annotated by up to five separate annotators, depending on the identified taxonomy. Leveraging the features of syntactic morphology, affective polarization, and word-integrated lexicon, we design and implement two classifiers for the Italian language, based on different learning algorithms: the first is based on support vector machines (SVM) and the second is based on a specific recurrent neural network called Long Short Term Memory (LSTM).

[5] Hate Speech Detection: A problem solved? Long Tail's hard case on Twitter: This work makes several contributions to the state of the art in this field of study. First, in-depth data analysis to understand the extremely imbalanced nature and lack of discriminatory characteristics of hate speech in thetypical datasets one faces in tasks. such service. Second, new DNN-based methods are proposed for such tasks, specifically designed to capture latent features that are potentially useful for classification. Finally, the methods have been carefully evaluated on Twitter's largest data collection of hate speech, to show that they can be particularly effective at detecting and classifying content. hate speech (as opposed to non-hateful content) that we have shown is more effective than hard and arguably more important in practice. The end results set a new standard in this field of research. With the growing popularityof deep learning-based NLP models, interpretable systems are needed.

III. METHODOLOGY

A. *Proposed System* III

To build a hate speech detection model using LSTM, we can follow these general steps:

1. Data Collection: Collect a large dataset of labeled data that contains examples of hate speech and non-hate speech. There are several publicly available datasets for hate speech detection that can be used.
2. Data Preprocessing: Preprocess the data by removing irrelevant information such as stopwords, punctuation, and special characters. Then tokenize the text and convert it to a sequence of integers to feed into the LSTM model.
3. Word Embeddings: Use pre-trained word embeddings such as GloVe, FastText or Word2Vec to represent each token in the text with a dense vector. This will help the LSTM model to learn better semantic relationships between words.
4. LSTM Model Architecture: Define the LSTM model architecture with an embedding layer followed by one or more LSTM layers. The output of the LSTM layers will be fed into a fully connected layer with a sigmoid activation function to produce a binary classification output.
5. Training: Train the LSTM model on the preprocessed and embedded data using the back propagation algorithm with cross-entropy loss. Adjust hyper parameters such as learning rate, batch size, and number of epochs to optimize the model's performance.
6. Evaluation: Evaluate the performance of the LSTM model on a separate test set using metrics such as accuracy, precision, recall, and F1 score. Tweak the model parameters and architecture as needed to improve its performance.
7. Deployment: Deploy the trained LSTM model as a service or integrate it into an application for real-time hate speech detection.

It is also important to note that hate speech detection is a complex and nuanced problem, and it may be necessary to incorporate additional techniques such as topic modeling, sentiment analysis, and user profiling to improve the accuracy of the model.

B. *Algorithm*

1. Data preparation: Collect and prepare a dataset of text labeled as either hateful or non-hateful. This dataset should be large enough to train the algorithm.
2. Tokenization: Convert the text into a numerical format that can be processed by the algorithm. Tokenization involves breaking up the text into words or subwords, andassigning each token a unique numerical value.
3. Embedding: Transform the tokens into a dense numerical representation using a word embedding technique. This step captures the semantic relationships between words and their contexts.
4. Bidirectional LSTM: Train a bidirectional LSTM neural network to classify the text as hateful or non-hateful.
1. Bidirectional LSTMs process the text in both forward and backward directions, allowing them to capture long-term dependencies in the text.
2. Output layer: The output layer of the LSTM network is a binary classifier that outputs a probability of the text being hateful or not.
3. Evaluation: Evaluate the performance of the algorithm on a test dataset. Common evaluation metrics include accuracy, precision, recall, and F1 score.
4. Tuning: Fine-tune the hyper parameters of the algorithmto optimize performance. This can include adjusting the number of LSTM layers, the number of neurons in each layer, and the learning rate.
5. Deployment: Deploy the trained model to classify new text as hateful or not.

Overall, using a bidirectional LSTM algorithm to detect hate speech involves a combination of natural language processing techniques and machine learning algorithms. It is important to use a large and diverse dataset to train the algorithm and to carefully evaluate its performance to ensure its effectiveness.

C. *System Architecture*

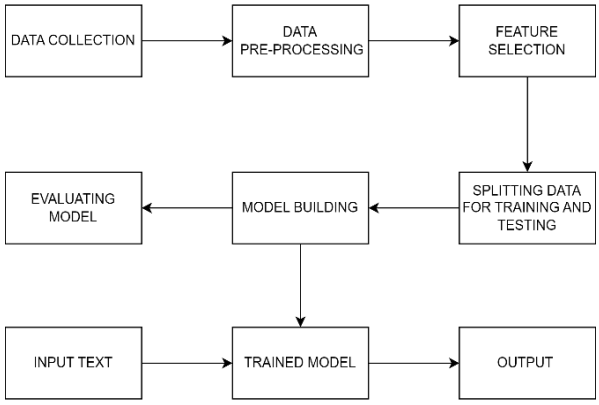

Fig.1.system architecture

Data Collection: The first step is to gather relevant data that will be used to train the model. This data can be obtained from various sources like APIs, databases, CSV files, etc.

Data Preprocessing: The raw data obtained in the first step is often not suitable for machine learning algorithms. This step involves cleaning, transforming, and encoding the data so that it can be fed into the machine learning model. Common preprocessing techniques include normalization, feature scaling, missing value imputation, and categorical encoding.

Feature Extraction: This step involves selecting the most relevant features or attributes from the preprocessed data that will be used to train the model. Feature extraction can be done manually or automatically using techniques like principal component analysis (PCA), independent component analysis (I C A), and linear discriminant analysis

(LDA).

Model Training: Once the relevant features are selected, the model is trained on the data using a suitable machine learning algorithm. There are many algorithms to choose from, such as linear regression, decision trees, neural networks, etc. The model learns from the input data and adjusts its parameters to minimize the error or loss function.

Model Evaluation: The trained model is evaluated on a separate set of data to assess its performance and accuracy. The evaluation metrics depend on the type of problembeing solved. For example, for a classification problem, the metrics could be accuracy, precision, recall, and F1 score.

Model Deployment: Once the model is trained and evaluated, it can be deployed in production to make predictions on new data. The deployment process involves integrating the model into the existing system, testing its functionality, and monitoring its performance over time.

The architecture of the ML pipeline typically consists of a data storage layer, a data preprocessing layer, a feature extraction layer, a model training layer, and a model evaluation layer. These layers can be implemented using various tools and technologies like Python, scikit-learn, TensorFlow, Keras, PyTorch, etc.

IV.RESULT

The proposed model aims to reduce the number of data annotation operations. Therefore, this technique contributes generalization of the apprenticeship system. Word-classified semantic vectors combine word information the context in which they occur. From the combined result uses semantic information to help select a subset of unlabeled text. This approach identifies unlabeled text-based cases of active learning. Method of integrating new learning points into model training. Hate speech detection using a BiLSTM (Bidirectional Long Short-Term Memory) model will typically be a binary classifier of whether a given text should be considered hate speech. The output is usually a probability score between 0 and 1, where 0 indicates that the text is not hate speech and 1 indicates that the text is hate speech. The BiLSTM model will be trained on a dataset of labeled examples of hate speech and non-hate text, using techniques such as word embedding, recurrent neural networks, and annotation mechanisms. idea. The model will then be used to predict whether new unseen texts are hate speech. It should be noted that the accuracy of hate speech detection using the BiLSTM model (or any other machine learning model) can vary depending on the quality and variety of the training data. , as well as the complexity and efficiency of the model 's architecture and parameters . In addition, determining what constitutes hate speech can be subjective and context dependent, so there may be some degree of ambiguity ordisagreement in the labeling of some texts.

V. FUTURE SCOPE

As natural language processing (NLP) and machine learning continue to evolve. BiLSTM (Bidirectional Long Short-Term Memory) is a deep learning algorithm that has been proven effective in text classification tasks such as sentiment analysis and hate speech detection. Some potential future directions for research and development in this area include: Multilingual hate speech detection: BiLSTM can be trained to detect the hate speech in multiple languages, which is especially important due to the global nature of social media and the internet. As a result, researchers can develop models capable of detecting hate speech in a variety of languages, which will have important implications for content monitoring and moderation. online. Contextualization: Hate speech detection can benefit from contextualization to better understand the underlying meaning of language

VI.CONCLUSION

In daily life, as the use of social media increases, people seem to think that they can say or write whatever they want. Due to this reflection, hate speech has increased, so there isa need to automate the process of classifying hate speech data. Interpretable natural language processing and deep learning have been adopted in recent years. Existing models are based on static data. Therefore, most traditional algorithms cannot account for significant changes. The proposed supervised learning method first labels the textand then trains the model. Two-way LSTM can achieve excellent accuracy when combined with active learning and attention networks.

VII.ACKNOWLEDGEMENT

The authors would like to thank Principal Vinodh P Vijayan, Neethu Mariya John, H.O.D, Faculty of Computer Science, for their appropriate guidance, valuable assistance and helpful comments during the proofreading process.

REFERENCES

[1] Usman Ahmed and Jerry Chun-Wei Lin SeniorMember, IEEE “Deep Explainable Hate Speech Active Learning on Social-Media Data”

[2] A. Rajkomar et al., “Scalable and accurate deep learning with electronic health records,” NPJ Digit. Med., vol. 1, no. 1, pp. 1–10, 2018.

[3] K. W. Johnson et al., “Artificial intelligence in cardiology,” J. Amer. College Cardiol., vol. 71, pp. 2668–2679, Jun. 2018.

[4] C. Krittanawong, H. Zhang, Z. Wang, M. Aydar, and T. Kitai, “Artificial intelligence in precision cardiovascular medicine,” J. Amer. College Cardiol., vol. 69, no. 21, pp. 2657–2664, 2017.

[5] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: Predicting clinical events via recurrent neural networks," in Proc. 1st Mach. Learn. Healthcare Conf., vol. 56, Aug. 2016, pp. 301–318.
[6] R. C. Feldman, E. Aldana, and K. Stein, "Artificial intelligence in the health care space: How we can trust what we cannot know," Stan. L. Pol'y Rev., vol. 30, p. 399, Jul. 2019.
[7] D. Gunning, "Explainable artificial intelligence," Defense Adv. Res. Projects Agency (DARPA), p. 2, 2017.
[8] A. Holzinger, G. Langs, H. Denk, K. Zatloukal, and H. Müller, "Caus- ability and explainability of artificial intelligence in medicine," WIREs, Data Mining Knowl. Discovery, vol. 9, no. 4, p. e1312, Jul. 2019.
[9] A. Vellido, "The importance of interpretability and visualization in machine learning for applications in medicine and health care," Neural Comput. Appl., vol. 32, pp. 18069–18083, Feb. 2019.
[10] S. L. James et al., "Global, regional, and national incidence, prevalence, and years lived with disability for 354 diseases and injuries for 195 countries and territories, 1990–2017: A systematic analysis for the global burden of disease study 2017," Lancet, vol. 392, no. 10159, pp. 1789–1858, Nov. 2018, doi:10.1016/S0140-6736(18)32279- 7.