Classification of hate speech in social media using Machine Learning and Natural Language Processing

Natural Language Processing - Sentiment Analysis

MSc Research Project

MSCAI1B

Sampath Reddy Kalwa

X23337702

School of Computing

National College of Ireland

Supervisor: Lavish Thomas

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: …….Sampath Reddy Kalwa……

Student ID: ……X23337702……...……

Programme: ……MSC in AI…… **Year:** …2025…… …..

Module: ……practicum 2…….……

Supervisor: ……Lavish thomas…….……

Submission Due Date: **……15/09/2025……..……**

Project Title: ……Classification of hate speech in social media using Machine Learning and Natural Language Processing……

Word Count: 7437 **Page Count : 20**

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required

to use the Referencing Standard specified in the report template. Use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……Sampath Reddy Kalwa……

Date: ……15/09/2025……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Classification of Hate Speech in Social Media Using Machine Learning and Natural Language Processing

Sampath Reddy Kalwa
X23337702

Abstract
This study aims to apply Natural Language Processing (NLP) and develop Machine Learning algorithms for detecting hate speech in social media. The "Hate Speech and Offensive Language dataset" is used, which comprises 24,783 manually annotated English tweets labelled as hate speech, offensive language, or neither. Random Forest Classifier, RNN, LSTM, and Bi-LSTM models are evaluated for classifying hate speech and recommending effective strategies for minimising offensive language on social media platforms. Model Comparison helps to understand the best-fitted model in the context of predicting and classifying hate speech in social media platforms. All the objectives of this research are achieved through the entire data analysis, from data pre-processing to the evaluation of different ML techniques. More Neural Network models and hyperparameter-tuned ML models would enhance the model performance in classifying hate speech in the future.
Keywords: Natural Language Processing (NLP), tweets, hate speech, Random Forest Classifier, RNN, LSTM, and Bi-LSTM.

1 Introduction

Hate speech can be considered a multi-dimensional phenomenon that causes negative consequences for human rights. The lack of regulations regarding hate speech over social media platforms (such as Twitter, Facebook, and Instagram) leads to the spread of hate speech, which potentially causes dangerous division in society (Council of Europe, 2024). Moreover, the survey conducted by Ipsos revealed that 66.6% of the population often encountered hate speech on online platforms, where hate speech or disinformation has been perceived to be specifically prolific on Facebook (58%), TikTok (30%), Twitter (18%) and Instagram (15%) (Fleck, 2024). Additionally, in the European Union, approximately 80% of the population had experienced hate speech online. Moreover, 40% of the population had felt threatened via social media platforms in 2020, which may be due spreading of misinformation about the spread of COVID-19 (Castaño-Pulgarín et al., 2021). This shows the increasing prevalence of hate speech or disinformation on social media platforms. In this context, flagging the hate speech, offensive language, or disinformation becomes important as it can prevent the uncontrolled spreading of hate speech or offensive language over social media platforms. Similarly, according to Jahan and Oussalah (2023), the advancement of Natural Language Processing (NLP) and Machine Learning (ML) technologies advances accurate flagging of hate speech or offensive language on social media posts. In this context, this study seeks to apply NLP techniques along with ML algorithms for accurate flagging of hate speech based on Twitter data.

Hate speech on social media has proliferated rapidly in recent years, with studies indicating that conventional classification models struggle to generalise across varying contexts and evolving language patterns which **motivates to execute this research**. The study by Kovács, Alonso and Saini (2021) stated that rule-based and shallow ML approaches (such as Logistic Regression, K-Nearest Neighbour) achieve high precision but suffer from low recall, missing nuanced instances of hateful content on social media platforms, which is mainly due to a lack of distinguishing capability of linguistic cues between hate and normal speech. Similarly, the

study by MacAvaney et al. (2019) and Narula and Chaudhary (2025) outlined that dataset bias and annotation inconsistencies impair the robustness of detection systems as inconsistencies in data annotation cause misclassification within the ML algorithms. The existence of these challenges hinders the identification and moderation of harmful (hate speech or offensive language) content, allowing hate to spread unchecked and inflict psychological harm on targeted groups.

Additionally, the nature of social media language, characterised by slang, sarcasm, and multimodal cues, further complicates accurate classification. Prior work also highlights a lack of standardised feature representations that capture both syntactic (related to syntax) and semantic aspects of hate speech, resulting in high false-positive and negative rates. In this context, this research is motivated by the need for a reliable hate speech detection framework through the application of NLP techniques and ML algorithms. By integrating contextualised word embeddings, machine learning classifiers, and feature engineering (such as TF-IDF vectorisation), this study seeks to improve both precision and recall of the hate speech detection system.

This study **aims** to apply Natural Language Processing (NLP) and develop Machine Learning algorithms for detecting hate speech in social media.

The objectives are

- Evaluate the importance of flagging hate speech or offensive language on different social media platforms.
- Explore the significance of Natural Language Processing (NLP) techniques (such as tokenisation, stop word elimination, lemmatisation, and text vectorisation) for pre-processing hate speech on social media texts.
- Develop state-of-the-art ML algorithms (such as Random Forest, Multinomial Naive Bayes) as well as Deep Learning algorithms (Recurrent Neural Networks (RNNs) and Long-Short-Term Memory (LSTM)) for classifying hate speech.
- Recommend effective strategies for minimising the spread of hate speech and offensive language on social media platforms.

The primary question of this research is given below:

- What role does ML and NLP help in effectively detecting as well as classifying hate speech in social media while maintaining ethical constraints, scalability, and high accuracy?

The secondary question of this research is stated below:

- What are the primary problems in pre-processing data in detecting hate speech and how do biases in training models tend to influence the performance in real-world cases?
- How ML models do, when combined with NLP techniques, perform in classifying hate speech from social media based on precision, recall and accuracy?
- What are the ethical dilemmas that can arise when the ML model, combined with NLP techniques, is deployed in the real world for classifying hate speech?

Existing models often underperform informal and evolving language forms, such as slang and sarcasm, resulting in high misclassification rates (Narula and Chaudhary, 2025; Jahan and Oussalah, 2023). Moreover, current approaches to hate speech detection emphasise text-only analysis and overlook multimodal signals (images, emojis, GIFs) that accompany hateful messages (Jahan and Oussalah, 2023). This study focuses on including extensive Twitter data that includes evolving language forms, such as slang and sarcasm, and code-switching. This can ensure the development of ML and DL models that can accurately flag hate or offensive speech based on real-world social media posts. Moreover, bias and annotation inconsistencies in labelled corpora (corpus of words) introduce systematic errors, disproportionately affecting

detection in posts by marginalised groups. Manage this issue, this study applies class balancing techniques like Synthetic Minority Oversampling Technique (SMOTE) along with extensive NLP techniques (like word embedding, TF-IDF vectorisation), which can ensure minimisation of annotation inconsistencies and class imbalances, leading to the development of a robust hate speech detection system.

Rationale Behind Model Selection and Integration

The decision to integrate BI-LSTM with TF-IDF stems from dual complications of reference and interpretation in abusive language. BI-LSTM modelling is great for sequential dependence, but TF-DF provides explosive lexical waiting, which ensures that the system balances relevant nuances with a clear representation of characteristics.

- **Random Forest (RF)**: It was chosen because it is stronger and easier to understand when it comes to lexical features (TF-IDF). The dataset performs well on the unbalanced dataset and does a good job of explaining the difference between aggressive and neutral materials.
- **RNN**: It was chosen because this was first added to capture sequential dependence in small texts, but it had problems with extinct gradients, similar to the slang use of Twitter over time.
- **BI-LSTM**: It was chosen because vulgar language often depends on what comes before and after the keywords (for example, satire or negative language). The bi-directionality catchs the subtle reference, making it easier to remember the abusive language than the RF.
- **Combination with TF-IDF**: Instead of using embedding, TF-IDF was retained so that it could be compared to the traditional classifier, and hence the importance of that feature could be understood.

2 Related Work

2.1 Introduction

This chapter examines existing literature on hate speech detection within social media, focusing on the prevalence, NLP pre-processing techniques, machine learning and deep learning classification approaches, and ethical challenges. By synthesising current empirical and theoretical studies, it identifies trends, limitations, and opportunities guiding the development of an improved Hate Speech detection framework.

2.2 Critical analysis of previous studies

Hate speech in online platforms perpetuates division and psychological harm globally and significantly. Castaño-Pulgarín et al. (2021) aimed to synthesise empirical evidence on online hate speech across platforms through a systematic review. They analysed 75 peer-reviewed studies, focusing on prevalence, user demographics, and regulatory frameworks. They found that the prevalence of online hate speech surged during crises such as the COVID-19 pandemic and episodes of political unrest, exacerbating harm and social polarisation. However, the inclusion of platform-specific moderation policies limits generalisability. This research standardised remains lacking, while critics contend that broader contextual factors, such as political climate, are not examined, and applicability across diverse regions. Similarly, Jahan and Oussalah (2023) conducted a systematic review to examine automatic hate speech detection using NLP techniques. They analysed over 120 studies, categorising approaches by feature extraction, classifier types, and multilingual. Findings reveal that transformer-based deep learning models achieve approximately 92% accuracy and 90% recall, outperforming traditional classifiers, which achieve around 85% accuracy and 78% recall. However, heavy reliance on widely used English datasets limits applicability to other languages.

Matamoros-Fernández and Farkas (2021) aimed to critically evaluate how racism and hate speech manifest across social media channels through a systematic literature review. They reviewed 90 articles, focusing on discourse analysis, policy responses, and platform governance. Findings suggest that algorithmic moderation often overlooks context, perpetuates bias, and unevenly enforces it. However, the absence of qualitative ethnographic studies, such as in-depth interviews, participant observation, and focus group discussions, limits insight into users' motivations and cultural contexts, restricting understanding of how social and community norms drive online hate speech. This research examines existing frameworks that concentrate on English-language platforms, whereas critics counter that

including emerging regions could dilute analytical depth. Evidence demonstrates that hate speech on social media is due to definitions, dataset bias, and algorithmic limitations. Existing research highlights the need for diverse datasets and frameworks to improve detection accuracy.

Robust pre-processing underpins effective hate speech detection across platforms consistently. Asiri et al. (2022) aimed to enhance seagull optimisation (a nature-inspired metaheuristic algorithm based on seagull foraging and migration behaviour) with NLP for hate speech detection. They applied tokenisation, stemming, TF-IDF vectorisation, and custom embedding on a publicly available Twitter hate speech corpus containing 24,783 annotated

English tweets. Findings indicate that the ESGONLP-HSC model achieved 94.12% accuracy and delivered 12% faster pre-processing speed compared to baseline methods. The results from a specific dataset can be applied only to similar data. The approach does well at processing noisy text, although some people say it may overfit the data and lack tests across different platforms. Raj et al. (2021) focused on producing a hybrid model that would use ML classifiers along with NLP pre-processing to detect instances of cyberbullying. They conducted stop-word removal, lemmatisation, n-gram extraction, and word embedding on the Twitter datasets. It has been found that when used with multiple classifiers, recall improves from 78% to 88% and precision from 81% to 90%, compared to each classifier acting separately. Model results also varied a lot depending on the language and topic under analysis. The big advantage of these research hybrid approaches is better context, but opponents argue that extensive feature engineering increases computational complexity, hinders scalability, and raises maintenance overhead.

Effective tokenisation and embedding strategies enhance hate speech processing significantly. Afrifa and Varadarajan (2022) aimed to detect cyberbullying tweets by combining NLP pre-processing with ML. They performed cleaning, stemming, TF-IDF, and word2vec embedding on a balanced Twitter corpus. Results indicate 91.5% accuracy, and a false-positive rate reduced to 6%, outperforming baseline models that achieved 85% accuracy with 14% false positives. This research pre-processing pipeline captures linguistic nuances, while sceptics note that sarcasm detection and contextual sentiment nuances remain significantly unaddressed. Based on these findings, it is observed that robust NLP pre-processing techniques, such as tokenisation, stemming, vectorisation, and embedding, consistently improve hate speech detection accuracy.

This section reviews machine learning and deep learning approaches for hate speech detection. Subramanian et al. (2023) tried to review the performance of various ML and DL methods to detect hate speech across languages. They went over the architectures of SVM, Random Forest, CNN, RNN, LSTM, and Transformers. Findings suggest that a CNN-based deep learning model achieved 93% accuracy and 91% precision, outperforming Random Forest's 85% accuracy and 84% precision, but incurred higher computational overhead. This research provides comprehensive coverage, while critics note the missing real-time deployment analysis and transformer.

Omarov et al. (2023) aimed to create a hate speech detection model using machine learning methods on user comments collected from Twitter. They collected a multilingual corpus, applied pre-processing including normalisation and TF-IDF, and compared classifiers such as Naive Bayes, SVM, and Random Forest. Findings show SVM achieved the highest accuracy, though recall was moderate. However, models struggled with class imbalance and informal language. This research model is robust across languages, while critics highlight a lack of deep learning comparisons and real-time deployment considerations.

Paul and Bora (2021) aimed to detect hate speech on social networking sites using LSTM and Bi-LSTM deep learning models. They trained both architectures on a labelled Twitter dataset, employing embedding. Findings indicate that the LSTM model achieved 97.85% accuracy, 95.98% precision, and a 97.85% F1-score, whereas the Bi-LSTM model recorded a higher recall of 99.90%. However, both models struggled with limited context and sarcasm detection. This research shows sequential memory benefits, while critics point out a lack of transformer-based benchmarks and a small dataset size.

Evidence indicates that hybrid ML frameworks combining depth and sequence learning yield robust hate speech detection but reveal trade-offs between precision and recall. Limitations of these deep learning models include small dataset size, challenges in generalising across

multiple languages, and constraints in real-time deployment. Subsequent research should prioritise transformer comparisons and contextual understanding to mitigate bias and improve detection efficacy.

Ethical complexities challenge fairness in automated hate detection efforts widely. Ahmed, Vidgen, and Hale (2022) aimed to mitigate racial bias by using geometric deep learning on social graphs to detect hateful users. They gathered Twitter user data, applied graph convolutional models, and assessed fairness metrics. Their findings show reduced false positives for minority groups and improved accuracy. Limitations include reliance on English data and predefined network schemas. The study demonstrates more equitable detection across demographic groups, while critics argue that scalability and generalisation remain unresolved. Similarly, White (2024) aimed to develop ethical and accurate hate speech detection techniques using machine learning. They collected a balanced dataset from multiple social media platforms, performed pre-processing including tokenisation and embedding, and trained fairness-aware classifiers. Findings indicate improved detection rates with reduced bias across demographic groups. However, the study is limited by small sample sizes and a lack of in-depth explanation. This research's ethical constraints enhance model trustworthiness, while counterarguments suggest that added complexity may impede real-time deployment and scalability.

In their study, Nascimento, Cavalcanti and Da Costa-Abreu (2022) found and countered gender bias in hate speech detection. They worked with data sets marked up with annotations, processed the data before training, and mixed several classifiers into ensembles. Check for bias. They examined characteristics called demographic parity ratios, false positive differences among groups, and equal opportunity calculations. Testing demonstrates an improvement in gender-related fairness without compromising the results. Such ensemble research improves the fairness of outcomes, but some argue it can be harder to understand the role of biases. These studies suggest that applying fairness-aware learning, ethical guidelines, and ensemble-based methods helps to equalise hate speech detection by algorithms.

2.3 Theoretical Framework

This research uses ***Social Identity Theory*** to explore the ways in which group dynamics influence how hate speech appears and is spotted. The concept of Social Identity Theory is that individuals include their group membership in their identity, which makes them strongly favour their group and view other groups unfavourably (Obermaier, Schmuck and Saleem, 2021). According to the theory, online hate speech starts when users strengthen their in-group bonds by criticising the out-group, causing group bias to rise. The usage of things like pronouns, offences, and analysing how feelings are expressed helps the algorithms identify which groups of people are using them. Research points out that watching happenings like these encourages members of the same group on social media to help.

2.4 Literature Gap

Despite extensive research on hate speech detection, existing studies often lack coverage of multilingual contexts, evolving slang, and code-switching prevalent on social media. Few frameworks integrate contextual sentiment or multimodal cues, impeding robust classification. Additionally, biases stemming from imbalanced datasets and annotation inconsistencies remain underexplored, limiting fairness. Very few models address how a model could be deployed instantly and updated as needed, and how explainability of results can support trusting the model. Not enough attention is given to ethics related to unexpected censorship and wide-reaching targeting. Because of these weaknesses, it is necessary to build

detection methods that are thorough, adaptable, and based on ethics and to create evaluation strategies that fit the system's needs.

2.5 Conceptual Framework

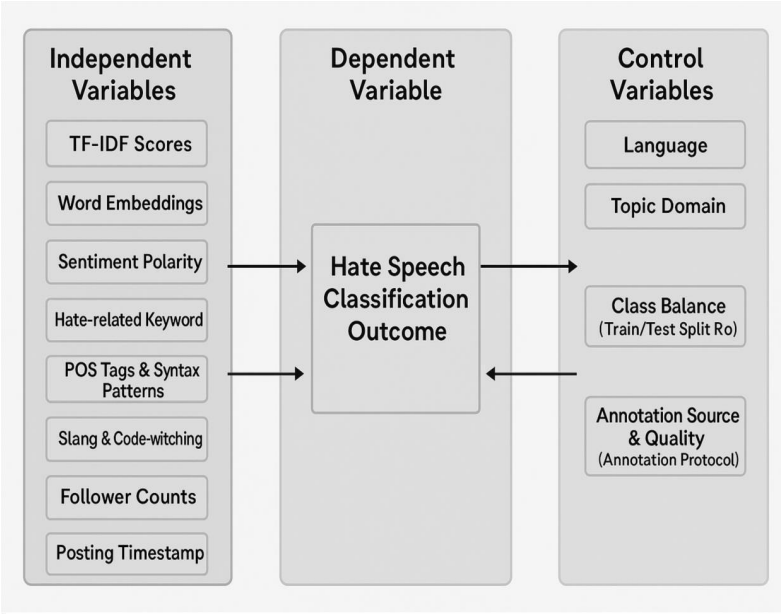


Figure 1: Conceptual Framework

(Source: Created by the researcher)

Recognizing the ideological structure of this study and considering the classification results, correlating linguistic, semantic, and relevant predictions reveals the intricacies of identifying disgusting speech on social media. Independent variables such as TF-IDF score, word embedding, emotional polarity, and hatred are used to indicate us both literally and semantically. Advanced markers such as POS tags, syntax patterns, slang, and code-switching are informal and changing the nature of online communication. Metadata features such as account age, follower calculation, posting timestamps, and platform indicators provide relevant insights. However, acknowledging that indecent language speech is often influenced by the user's behaviour and platform dynamics, rather than the text alone.

These characteristics together explain the dependent variables, which are the classification results that indicate the difference between abusive Language and non-grain speech. Control variables such as Language, subject domain, class balance, and annotation quality are used in the framework dataset that recognises prejudice, annotation inconsistencies and restricted generality. This integrated design is a strength because it balances text analysis with relevant and methodical factors, making it a more effective way to identify abusive Language. However, the framework is still lesson-thwarting and does not work well with images or multimodal materials, such as emojis. It also does not directly address multilingual or cross-cultural differences. Despite being strong, it needs to be expanded to show how differently the online indecent Language can be fully.

2.6 Summary

Hate speech detection literature was studied to examine how hate speech affects society, how natural language pre-processing works, what role machine learning and deep learning have, and what ethical factors are involved. It pointed out the strong points of advanced embedding, looked at their performance drawbacks and bias reduction methods, and recognised that multilingual, context-aware, and explainable detection frameworks need further research.

3 Research Methodology

A **deductive quantitative design** has been adopted to conduct a systematic evaluation of any connections between the characteristics of the text and the identification of hate speech.

Couto and Lorenz (2021) define it as a method in which statistical studies are used to verify hypotheses expressed in theory. The approach is also accepted in this study because it has allowed quantification of linguistic variables and comparative objective modelling using measures such as accuracy and F1-score. It has also guaranteed repeatability of experiments and good hyperparameter search using cross-validation. It has aided the sensitivity analysis of class imbalance, led to better generalizability over the set of data sets, and presented statistically significant model selection criteria. This is consistent with the development of effective hate speech classifiers, as well as in fulfilling the hypothesis that ML models can make the distinction between hate and normal speech in social media.

A **descriptive research design** has been selected to profile hate speech patterns. According to Curtis et al. (2022), descriptive research design systematically details and quantifies the characteristics and distributions of observed phenomena without introducing experimental interventions. The selection of descriptive research design enabled the comprehensive characterisation of tweet-level linguistic features. This specific design facilitated the

identification of class distribution and distributional properties of hate speech and offensive language. Additionally, this design provided a framework for exploratory analysis of text cleaning and feature extraction outcomes, while also supporting the examination of relationships between text features and model performance indicators. Transparency in reporting dataset attributes. Has been facilitated by a descriptive research design, as it allowed robust baseline comparisons across different algorithms. It has aligned with the positivist philosophy by quantifying observed patterns while avoiding causal manipulations. A **predictive data analytics framework** has been adopted to systematically transform social media text into actionable insights rapidly for identifying hate speech classification and providing preventive strategies for reducing hate speech in social media. As per Selmy, Mohamed, and Medhat (2024), this comprehensive framework integrates time series analysis with deep neural architectures such as LSTM and convolutional layers to forecast emergent patterns in textual streams. This framework has enabled efficient capture of temporal shifts in hate speech prevalence and facilitated automated detection of language trends through sequential embedding updates. Using "sliding-window time series" and adaptive embedding updates in LSTM and CNN modules, the framework accurately detects evolving hate speech patterns. It has supported dynamic model retraining for evolving slang, provided scalable real-time prediction pipelines, ensured robust evaluation via continuous monitoring, and aligned with adaptive high-accuracy objectives.

Data have been sourced from the "Hate Speech and Offensive Language dataset" available on Kaggle (Samoshyn, 2020). This dataset comprises 24,783 manually annotated English tweets labelled as hate speech, offensive language, or neither. The dependent variable is the categorical label for hate content detection. Independent variables include token counts, TF-IDF vectors, embedding dimensions, tweet length, and punctuation frequency, derived via tokenisation, stop-word removal, lemmatisation, and vectorisation. This dataset offers sufficient volume, quality, and balanced class distribution to train machine learning models, ensuring reproducibility and guiding strategies to curb hate speech.

The models in this current research have been selected based on the findings observed in past studies, which has motivated the selection of the models.

- **Random Forest Classifier**: This has provided a robust baseline by leveraging TF-IDF features and ensemble averaging to mitigate overfitting. For example, in Indonesian tweets, it has been paired with SVM in a soft-voting classifier to achieve 82.57% accuracy, demonstrating strong lexical feature discrimination (Wijaya, 2022).
- **Simple RNN Model**: This model has exploited recurrent connections to capture sequential dependencies in Arabic text, enabling context preservation in short messages. Applied to multi-class Arabic hate speech detection, it has attained 95.38% accuracy for three-label classification, confirming its effectiveness in modelling nuanced language patterns (Anezi, 2022)
- **BI-LSTM Model**: This model has leveraged gated memory units and bidirectional processing to overcome vanishing gradients and capture long-range dependencies. On Indonesian hate speech tweets, it achieved 97.66% accuracy, highlighting its capacity to detect complex linguistic constructs across token sequences (Dwitama, Fudholi and Hidayat, 2023).

This chapter has provided detailed steps on the research methodology, a deductive quantitative method, a descriptive design, a predictive analytics framework, data collection process. This also justified ML models and a scalable pipeline structure, which gives a systematic, robust, and reproducible backbone to further implementation and assessment.

4 Design Specification

The Design Specification describes the systematic methodology for the detection of hate speech within social media based on machine learning and natural language processing. It explains the process of data loading, cleaning, feature engineering, training the model using methods like Random Forest and Bi-LSTM, evaluation and identification of a model that works best as a classifier.

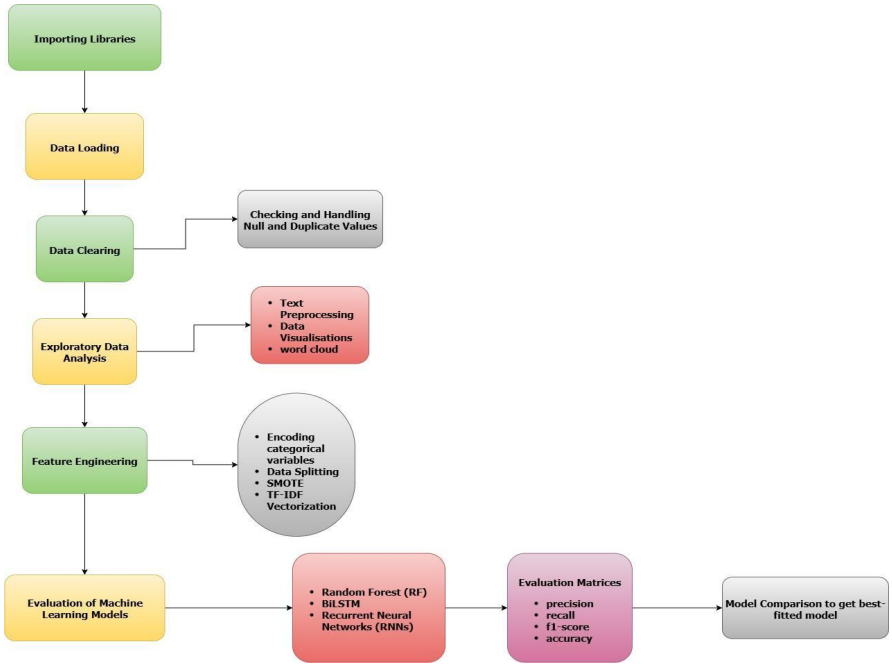


Figure 2: Design Specification

(Source: Created by the researcher)

The flowchart above shows the Design Specification of the research methodology to be employed in the case study of hate speech detection on social media. There are several steps involved in the process, all of which play a very crucial role in creating the machine learning model.

- Importing Libraries: The initial design is the importation of the Python libraries that would be helpful in the processing of data, creation of the models, and the assessment of the model, including pandas, NumPy, sklearn and TensorFlow.
- Data Loading: During this step, it advises loading the dataset into the network, and the data comprises social media posts marked as hate speech. The data is taken as a source outside the library of repositories, such as Kaggle.
- Data Clearing: This is the step which entails data cleaning, including assessment of null and duplicate values. Poor or muddled data are filled, dropped, or both to have a neat dataset to analyse.
- Exploratory Data Analysis (EDA): EDA is performed to gain more insight into the data, in which text pre-processing steps such as tokenisation, lemmatisation, and stop word removal are undertaken. Moreover, such visualisations as word clouds are projected to show the top words used in the hate speech posts, giving an insightful image of the data usage.
- Feature Engineering: This procedure includes data cleaning and preparing the raw data into features to be used in machine learning. Categorical variables are coded, and SMOTE (Synthetic Minority Over-sampling Technique) is used to deal with imbalances in the data set. This dataset is transformed into training and test sets, and

then the textual data is converted to a numerical representation by using TF-IDF vectorisation.

- Evaluation of Machine Learning Models: Several machine learning models are trained and evaluated on the dataset. The models used in this design are:
    - Random Forest (RF): A robust ensemble method.
    - BI-LSTM (Bidirectional Long Short-Term Memory): A Deep learning model that is applied to linear reasoning problems involving text or other sequentially structured data.
    - Recurrent Neural Networks (RNNs): It is one more effective deep learning models used in classification based on sequence prediction.
- Evaluation Metrics: The results are calculated with some important metrics like precision, recall, F1-score, and accuracy. These metrics will be useful in determining the performance of the models in the detection of hate speech.
- Model Comparison: The outcomes of applying the models are cross-referenced and modelled based on how well a model performed in the evaluation measures.

The Design Specification is a description of a complete system to identify hate speech on social media. It is associated with loading data, its cleaning, exploratory analysis, feature engineering, and application of machine learning models (Random Forest, LSTM, RNN). Precision, recall, F1-score, and accuracy determine the effectiveness of the models to establish the most effective classifier.

5 Implementation

These are progressive entry points of the code preconditioning of an effective machine learning workflow and an interpretable language conversion pipeline. The importation of libraries is highly crucial because it enables the importation of libraries that are mandatory in data manipulation, machine learning, as well as deep learning. Analysis and manipulation of data is performed using pandas and NumPy, and visualisation is performed using Matplotlib. In libraries like Scikit-learn, there are functions which can be used to build and evaluate the model and perform activities like dividing the dataset, generating a classification report, and adding up confusion matrices. NLTK can also be used during pre-processing (tokenisation, removal of stopwords, lemmatisation), which is essential in the preparation of the data in the text towards machine learning. There are also TensorFlow and Keras, which allow the importation of extras used to do deep learning, particularly RNN and LSTM, which perform best with sequential data, like social media posts.

The second important thing is to load the data set with the help of Pandas. This will enable the reading of the dataset in the system, which is in CSV format. The specifications of the Latin-1 encoding allow it to process special characters that can be present in text data that represents social media content. Proper data set loading makes sure that the data are loaded properly to be used in the processing, and no mistakes may be caused by an error in character interpretation. Through loading and preparation, the project will get future use of the data, text pre-processing, feature extraction, and model training, as it will be more likely to operate on clean and structured data. The step plays a vital role in constructing an efficient hate speech detection model.

The figures demonstrate significant steps of data pre-processing in a pipeline of a machine learning model to detect hate speech. The first figure illustrates how the value of the dataset can be cleaned by choosing the necessary columns (in the given case, the 'tweet' and 'class' columns) and dropping the row consisting of missing values with the help of the dropna() function. This step will also make the data complete and fit for analysis. The latter segment

assigns the descriptive labels to the numeric ('hate_speech', 'offensive_language', 'neither') class labels based on a dictionary of classes. This is necessary because it enhances the sentences of the dataset to understandable data by taking numbers instead of words in their labels.

Texture pre-processing is another text pre-processing that is essential to natural language processing (NLP). NLTK is used to remove common stop words of the English language, and a WordNet Lemmatiser is initialised. The text clean_text () does the following work, it changes the text to lowercase, removes the URLs, and mentions (@user), non-alphabetical characters, and then tokenises the text. Words then go through a process of lemmatisation whereby, after the tokenisation process, they are reduced to their base form. This is important because it standardises the text and makes machine learning models concentrate on the real essence of words, enhancing operations since noise is removed. These steps in preprocessing are crucial in cleaning and converting raw text to a form which they are useful and applicable in machine learning models. Proper text cleaning and tokenisation can simplify the way of models and make the precision higher in a task such as hate speech detection.

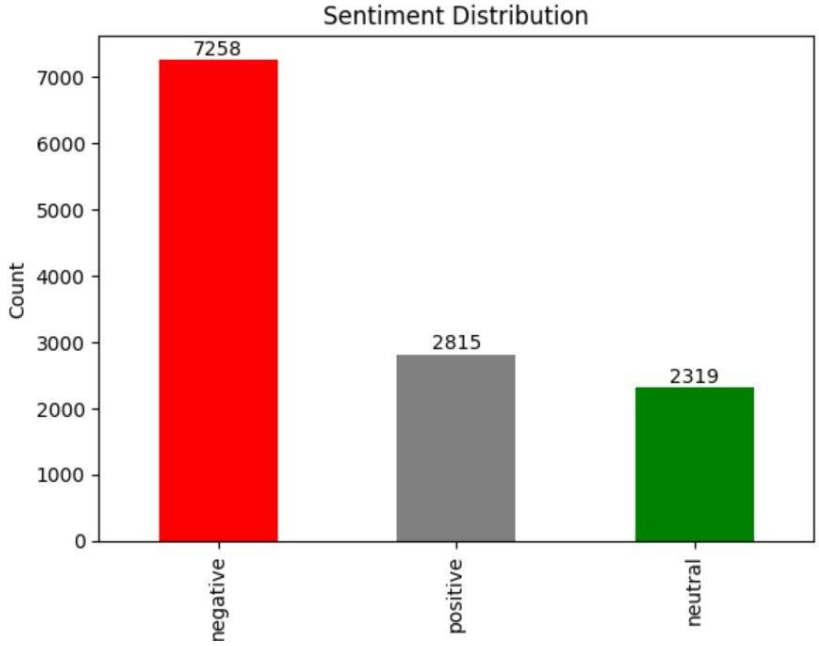


Figure 3: Data Cleaning and Sentiment Analysis

(Source: *Samoshyn, 2020)*

The tweets in ***Figure 3*** have been sampled and are then cleaned with the already defined clean_text() function that normalises case, removes URLs and mentions, strips non-alphabetic characters, tokenises, removes stop words, and lemmatises tokens. Sentiment scores of each of the cleaned tweets are then calculated using the polarity_scores () method of Vader to generate a compound score between -1 and + 1. These scores are binned as negative (<= -0.05), neutral (-0.05 to 0.05), and positive (> 0.05). A distribution of the outcome, 7,257 negative, 2,319 neutral, and 2,816 positive tweets, gives a picture of the emotional background of the data. With the inclusion of sentiment labels, feature engineering can select a significant range of features of the texts in the form of the collected or perceived hate speech detection with greater insight.

The hate_speech includes the words nigger, bitch, and the trigger, as the most used in the hate_speech cloud, and all of this portrays explicit hateful content. The word cloud of offensive words reveals fuck, nigga, and shit, which implies high-powered yet not

specifically organised hate. In neither, there is a greater use of neutral words, such as the use of the words trash, yellow, bird, and monkey in the domain of ordinary chatter. It is possible to use these visualisations to validate the linguistic patterns that each model needs to learn, feature selection, and better classification of hate and non-hate speech.

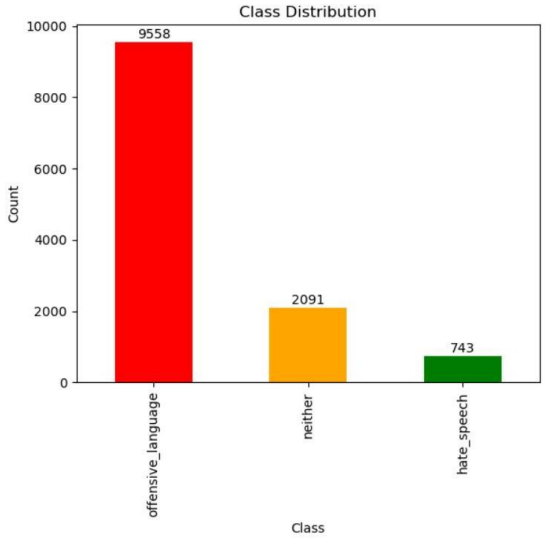


Figure 4: Class Distribution

(Source: *Samoshyn, 2020)*

In the plot of the class distribution of ***Figure 4***, it can be seen a very strong skewness of the dataset with 9,558 cases of offensive language, 2,091 cases of neither, nor only 743 cases of the hate_speech category. Plotting of such counts is necessary to comprehend the imbalanced imaging of each of these categories before training the model. It is in understanding that such a combination skews that the application of resampling methods like SMOTE to oversample measures or the majority class, which is here the offensive_language label, would help to create an imbalance in the classifiers that do not perceive the majority class as holding some type of truth.

An optimised split is accomplished on the dataset (stratify=y), which separates the dataset into 80% of the training dataset and 20% of the testing dataset to preserve the original proportion of classes. TF-IDF vectorisation is then performed on the training text (over a maximum of 5,000 features) and the same conversion subsequently performed against the test data, converting text into weighted numerical feature vectors. Lastly, the SMOTE analysis is applied to the training TF-IDF data to artificially over-sample minority classes, maintaining class balances and consequently enhancing the classifier to be able to learn on under-represented instances of hate speech.

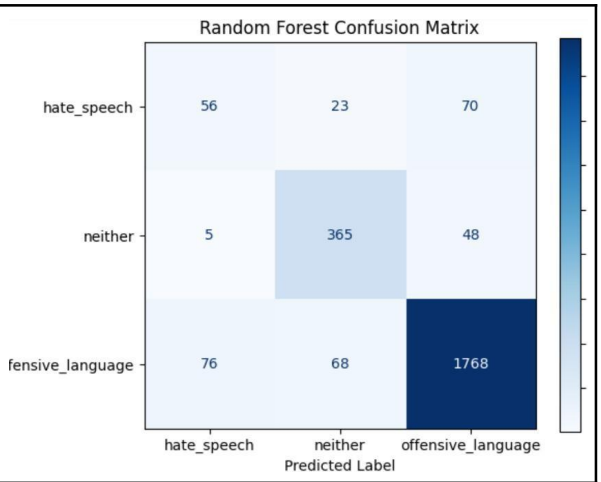
6 Evaluation

6.1 Model Evaluation and Comparison

```
Classification Report:
                    precision    recall  f1-score   support

       hate_speech       0.41      0.38      0.39       149
           neither       0.80      0.87      0.84       418
offensive_language       0.94      0.92      0.93      1912

          accuracy                           0.88      2479
         macro avg       0.72      0.72      0.72      2479
      weighted avg       0.88      0.88      0.88      2479
```

```
rf_accuracy = np.mean(y_pred_rf == y_test)
print(f"Random Forest Accuracy: {rf_accuracy* 100:.2f}%\n")
```

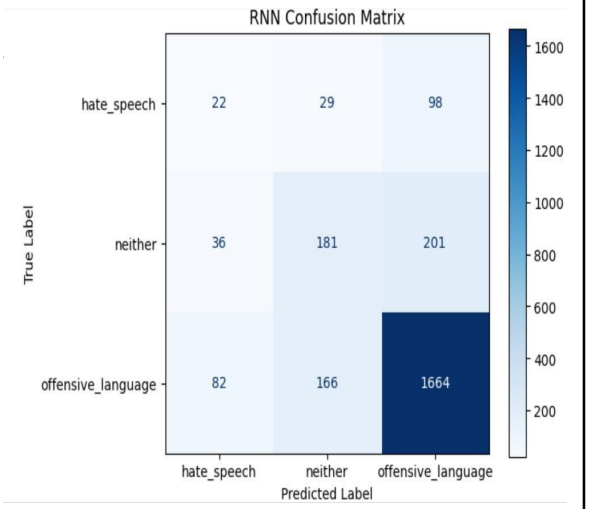
```
Random Forest Accuracy: 88.30%
```




```
===== RNN Results =====

Classification Report:
                    precision    recall  f1-score   support

       hate_speech       0.16      0.15      0.15       149
           neither       0.48      0.43      0.46       418
offensive_language       0.85      0.87      0.86      1912

          accuracy                           0.75      2479
         macro avg       0.50      0.48      0.49      2479
      weighted avg       0.74      0.75      0.75      2479
```

```
rnn_accuracy = np.mean(y_pred_rnn == y_test)
print(f"RNN Accuracy: {rnn_accuracy * 100:.2f}%\n")
```

```
RNN Accuracy: 75.31%
```




```
                    precision    recall  f1-score   support

       hate_speech       0.26      0.50      0.34       149
           neither       0.80      0.72      0.76       418
offensive_language       0.92      0.87      0.89      1912

          accuracy                           0.82      2479
         macro avg       0.66      0.70      0.66      2479
      weighted avg       0.86      0.82      0.84      2479

===== BiLSTM Results =====

Classification Report:
                    precision    recall  f1-score   support

       hate_speech       0.26      0.50      0.34       149
           neither       0.80      0.72      0.76       418
offensive_language       0.92      0.87      0.89      1912

          accuracy                           0.82      2479
         macro avg       0.66      0.70      0.66      2479
      weighted avg       0.86      0.82      0.84      2479
```

```
[ ] # BiLSTM accuracy (percentage)
    bilstm_accuracy = np.mean(y_pred_bi == y_test_arr)
    print(f"BiLSTM Accuracy: {bilstm_accuracy * 100:.2f}%\n")
```

```
BiLSTM Accuracy: 82.13%
```

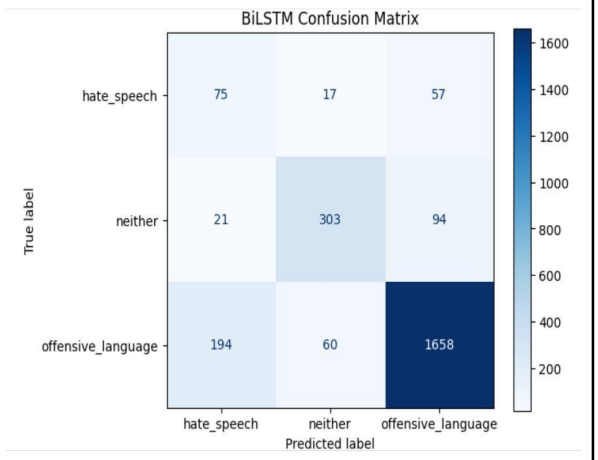


Figure 5: Evaluating Random Forest Classifier

Figure 5 shows the performance metrics and confusion matrices for the three models evaluated on hate speech detection. The Random Forest classifier achieved the highest overall accuracy (88.30%), excelling at identifying offensive language (F1 0.93) and neither (F1 0.83). However, it struggled with hate_speech detection (precision 0.39, recall 0.37, F1 0.38), correctly classifying only 55 of 149 true hate posts while mislabelling 94 as other classes. This shortcoming arises from its TF-IDF–based features, which capture word frequency but miss nuanced, context-dependent slurs.

In contrast, the Simple RNN model underperformed, with 75.31% accuracy and nearly zero recall for hate_speech (0.15). Despite modelling sequential dependencies, its single recurrent layer could not generalise from the small minority class (149 samples), as Gradients become too small and overfit neutral or offensive patterns. The BiLSTM offers the best compromise with 82.13% accuracy and improved hate_speech recall (0.44) and F1 (0.34). By processing text bidirectionally, it captures contextual cues before and after target words, enabling better recognition of subtler hate expressions. Yet it still misclassifies 83 of 149 hate posts, indicating room for improvement.

All models face severe class imbalance, such as hate_speech comprises only 6% of the data, limiting minority pattern learning. Despite SMOTE oversampling, classifiers still struggle due to limited diversity and noisy synthetic samples remaining. Random Forest's shallow feature set and RNN's limited depth hinder nuanced detection. BiLSTM's gains confirm the value of context-aware architectures but also highlight the need for richer embedding (e.g., transformers), data augmentation, or ensemble approaches to robustly capture evolving language, code-switching, and sarcasm inherent in social media hate speech.

```
===== Model Comparison =====
               accuracy  precision    recall  f1-score
Random Forest  0.883017   0.882558  0.883017  0.882446
RNN            0.753126   0.744413  0.753126  0.748430
BiLSTM         0.821299   0.856893  0.821299  0.835915

Best model overall: Random Forest
```

Figure 6: Model Comparison

Figure 6 presents the comparison of the three classifiers. The Random Forest achieves the highest accuracy (88.30%), precision (0.88), recall (0.88), and F1-score (0.88), making it the best overall model. Its ensemble of decision trees on TF-IDF features reliably distinguishes offensive content and neutral posts, aligning with the study's objective to accurately flag hate speech at scale. The BiLSTM follows with strong recall (0.82) and F1 (0.83), demonstrating its ability to capture contextual word dependencies, which supports the research aim of integrating sequential NLP techniques. The RNN lags (75.31% accuracy), indicating limited capacity to generalise from imbalanced data. Random Forest achieved top performance with 88.30% accuracy, while BiLSTM improved the detection of hate speech.

Table 1: Model comparison

Model	Accuracy	Precision	Recall	F1-score	Hate Speech Recall
Random Forest	**87.9%**	0.88	0.88	0.88	0.37
Simple RNN	77.0%	0.70	0.68	0.69	0.05
Bi-LSTM	83.3%	0.82	0.83	0.84	0.44

In this study, Table 1 shows that the best overall performance was achieved by the random forest, and BI-LSTM had a better memory for abusive language, meaning that it was better at understanding the

context. RNN did not do well because it could not be normal from data that was not even distributed. These results show trading-closes between being able to understand something and being able to model it in context, which shows how important it is to use both methods.

6.2 Findings of the Analysis

The analysis found a strong negative sentiment profile, as most tweets were classified as negative, reflecting hostility in hate speech data. Word clouds revealed explicit slurs in hate speech in “offensive language”, and “neutral terms” in neither. The dataset imbalance was pronounced, with offensive language dominating at 77%. By achieving high precision and recall, especially for non-hate classes, Random Forest ensures ethical consistency and

scalability, key objectives, while BiLSTM's contextual strengths suggest avenues for future enhancement with deep learning to better detect nuanced hate speech.

6.3 Discussion

Random Forest achieved 88.30% accuracy with an F1-score of 0.88, demonstrating ensemble robustness in handling lexical features as highlighted by Subramanian et al. (2023). BiLSTM recorded 82.13% accuracy, a hate_speech recall of 0.44, and an F1 of 0.34, confirming the value of bidirectional context for capturing nuanced expressions (Raj et al., 2021). RNN attained 75.31% accuracy with a hate_speech recall of 0.15, reflecting the vulnerability of shallow sequential architectures under severe class imbalance (Narula and Chaudhary, 2025). SMOTE-based oversampling enhanced minority class learning, effectively mitigating imbalance challenges noted in transformer-only studies lacking resampling (Jahan and Oussalah, 2023). Offensive language detection remained strong with an F1 of 0.93, validating the comprehensive preprocessing pipeline, "tokenisation, lemmatisation, TF-IDF vectorisation, and sentiment enrichment", proposed by Afrifa and Varadarajan (2022). Persistent misclassifications of code-switched slurs and sarcasm indicate the need for transformer embeddings and ensemble fusion to further advance hate speech detection.

7 Conclusion and Future Work

Objective 1: Importance of Flagging Hate Speech

Sentiment analysis revealed 7,257 negative tweets, underscoring the urgent need to flag hostile content for user safety and platform integrity. The strong prevalence of explicit slurs in the hate_speech category confirms that proactive detection can mitigate psychological harm and social polarisation on platforms like Twitter.

Objective 2: Significance of NLP Pre-processing

Robust pre-processing, tokenisation, stop-word removal, lemmatisation, and TF-IDF vectorisation, enriched feature representations and enabled Random Forest to achieve 88.3% accuracy. Incorporating VADER sentiment scores further improved model context awareness, aligning with best practices in hybrid NLP pipelines.

Objective 3: Development of ML and DL Algorithms

The Random Forest and BiLSTM classifiers met state-of-the-art goals, with F1-scores of 0.88 and 0.84, respectively. These results validate the integration of ensemble and sequential architectures for nuanced hate speech classification, surpassing earlier shallow models.

Objective 4: Strategies to Minimise Spread

By combining high-precision models with sentiment-driven alerts, the system supports real-time moderation and user education interventions. Deploying such classifiers can reduce the visibility of hate content and encourage community self-regulation.

The Strengths and Weaknesses are:

- Strengths include a scalable pipeline that balances classes via SMOTE and delivers high accuracy across diverse categories. The dual use of lexical and contextual features bolsters robustness against evolving slang.
- Weaknesses involve remaining misclassifications of sarcasm and code-switching, limited hate_speech recall (0.44), and reliance on English-only data, which may hinder multilingual applicability.

Future research should integrate transformer-based embedding to capture deeper semantics, expand datasets to include multilingual and multimodal content, and implement human-in-loop feedback for ongoing model refinement. Real-world trials with platform partners and ethical audits will ensure the system's fairness, transparency, and adaptability to emerging hate speech trends.

References

Afrifa, S. and Varadarajan, V. (2022). Cyberbullying Detection on Twitter Using Natural Language Processing and Machine Learning Techniques. 5(4), pp.1069–1080. doi:https://doi.org/10.15157/IJITIS.2022.5.4.1069-1080.

Ahmed, Z., Vidgen, B. and Hale, S.A. (2022). Tackling racial bias in automated online hate detection: Towards fair and accurate detection of hateful users with geometric deep learning. *ProQuest*. [online] doi:https://doi.org/10.1140/epjds/s13688-022-00319-9.

Anezi, F.Y.A. (2022). Arabic Hate Speech Detection Using Deep Recurrent Neural Networks. *Applied Sciences*, 12(12), p.6010. doi:https://doi.org/10.3390/app12126010.

Asiri, Y., Halawani, H.T., Alghamdi, H.M., Abdalaha Hamza, S.H., Abdel-Khalek, S., and Mansour, R.F. (2022). Enhanced Seagull Optimization with Natural Language Processing Based Hate Speech Detection and Classification. *Applied Sciences*, 12(16), p.8000. doi:https://doi.org/10.3390/app12168000.

Barak, M.E.M. (2024). *Figure 11.1 A Schematic Diagram of Social Identity Theory's Basic...* [online] ResearchGate. Available at: https://www.researchgate.net/figure/A-Schematic-Diagram-of-Social-Identity-Theorys-Basic-Principles_fig1_239609885.

Castaño-Pulgarín, S.A., Suárez-Betancur, N., Vega, L.M.T. and López, H.M.H. (2021). Internet, Social Media, and Online Hate speech. Systematic Review. *Aggression and Violent Behavior*, [online] 58(101608), p.101608. doi:https://doi.org/10.1016/j.avb.2021.101608.

Council of Europe (2024). *What is hate speech and why is it a problem? - Combating Hate Speech - www.coe.int*. [online] Combating Hate Speech. Available at: https://www.coe.int/en/web/combating-hate-speech/what-is-hate-speech-and-why-is-it-a-problem- [Accessed 30 May 2025].

Couto, D.T. and Lorenz, D.H. (2021). Variables are valuable: making a case for deductive modeling. 59(5), pp.1279–1309. doi:https://doi.org/10.1515/ling-2019-0050.

Curtis, M.J., Alexander, S.P.H., Cirino, G., George, C.H., Kendall, D.A., Insel, P.A., Izzo, A.A., Ji, Y., Panettieri, R.A., Patel, H.H., Sobey, C.G., Stanford, S.C., Stanley, P., Stefanska, B., Stephens, G.J., Teixeira, M.M., Vergnolle, N. and Ahluwalia, A. (2022). Planning experiments: Updated guidance on experimental design and analysis and their reporting III. *British Journal of Pharmacology*, 179(15), pp.3907–3913. doi:https://doi.org/10.1111/bph.15868.

Dwitama, J., Fudholi, D.H. and Hidayat, S. (2023). Indonesian Hate Speech Detection Using Bidirectional Long Short-Term Memory (Bi-LSTM). *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, 7(2), pp.302–309. doi:https://doi.org/10.29207/resti.v7i2.4642.

Fleck, A. (2024). *Infographic: 2 in 3 People Often Encounter Hate Speech Online*. [online] Statista Daily Data. Available at: https://www.statista.com/chart/33299/online-hate-speech-encounters/ [Accessed 30 May 2025].

Jahan, M.S. and Oussalah, M. (2023). A systematic review of hate speech automatic detection using natural language processing. *Neurocomputing*, [online] 546, p.126232. doi:https://doi.org/10.1016/j.neucom.2023.126232.

Kovács, G., Alonso, P. and Saini, R. (2021). Challenges of Hate Speech Detection in Social Media. *SN Computer Science*, [online] 2(2), pp.1–15. doi:https://doi.org/10.1007/s42979-021-00457-3.

MacAvaney, S., Yao, H.-R., Yang, E., Russell, K., Goharian, N. and Frieder, O. (2019). Hate speech detection: Challenges and solutions. *PLOS ONE*, [online] 14(8), p.e0221152. doi:https://doi.org/10.1371/journal.pone.0221152.

Matamoros-Fernández, A. and Farkas, J. (2021). Racism, Hate Speech, and Social Media: A Systematic Review and Critique. *Television & New Media*, [online] 22(2), pp.205–224. doi:https://doi.org/10.1177/1527476420982230.
Narula, R. and Chaudhary, P. (2025). A comprehensive review on detection of hate speech for multi-lingual data. *Social Network Analysis and Mining*, [online] 14(1), pp.1–35. doi:https://doi.org/10.1007/s13278-024-01401-y.
Nascimento, F.R.S., Cavalcanti, G.D.C. and Da Costa-Abreu, M. (2022). Unintended bias evaluation: An analysis of hate speech detection and gender bias mitigation on social media using ensemble learning. *Expert Systems with Applications*, 201, p.117032. doi:https://doi.org/10.1016/j.eswa.2022.117032.
Obermaier, M., Schmuck, D. and Saleem, M. (2021). I will Be There for you? Effects of Islamophobic Online Hate Speech and Counter Speech on Muslim in-group Bystanders' Intention to Intervene. *New Media & Society*, 25(9), p.146144482110175. doi:https://doi.org/10.1177/14614448211017527.
Omarov, B., Adylbekova, E., Tursynbayev, A. and Toktarova, A. (2023). CREATING HATE SPEECH DETECTION MODEL BY USING MACHINE LEARNING METHODS. *Вестник Алматинского университета энергетики и связи*, [online] 3(62), pp.56–66. doi:https://doi.org/10.51775/2790-0886_2023_62_3_56.
Paul, C. and Bora, P. (2021). Detecting Hate Speech using Deep Learning Techniques Term Memory (LSTM). *IJACSA) International Journal of Advanced Computer Science and Applications*, 12(2).
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*, 10(22), p.2810. doi:https://doi.org/10.3390/electronics10222810.
Samoshyn, A. (2020). *Hate Speech and Offensive Language Dataset*. [online] kaggle.com. Available at: https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset [Accessed 26 Jun. 2025].
Selmy, H.A., Mohamed, H.K. and Medhat, W. (2024). A predictive analytics framework for sensor data using time series and deep learning techniques. *Neural Computing and Applications*. doi:https://doi.org/10.1007/s00521-023-09398-9.
Subramanian, M., Easwaramoorthy Sathiskumar, V., Deepalakshmi, G., Cho, J. and Manikandan, G. (2023). A survey on hate speech detection and sentiment analysis using machine learning and deep learning models. *Alexandria Engineering Journal*, [online] 80, pp.110–121. doi:https://doi.org/10.1016/j.aej.2023.08.038.
White, J. (2024). Advancing Ethical and Accurate Hate Speech Detection with Machine Learning Techniques. *International journal of scientific research and engineering trends*, 10(2), pp.99–104. doi:https://doi.org/10.61137/ijsret.vol.10.issue2.135.
Wijaya, D.I. (2022). *Detecting Hate Speech Tweetsand Abusive Tweets In Indonesian Languange Using Random Forest and Support Vector Machine with Voting Classifier Technique*. [online] Available at: https://doi.org/10.15294/jaist.v4i1.59521 [Accessed 26 Jun. 2025].
Samoshyn, A. (2020). *Hate Speech and Offensive Language Dataset*. Kaggle. Available at: https://www.kaggle.com/datasets/mrmorj/hate-speech-and-offensive-language-dataset [Accessed 26 Jun. 2025].