# Project Report

Name: Aravindh V

Course: B.Sc CS (AI & ML)

Registration Number: TU6243202111006

## Project Title:

Hate Speech Detection using Natural Language Processing (NLP)

## 1. Project Description

This project focuses on detecting hate speech in text data using Natural Language Processing and Deep Learning techniques. Hate speech refers to offensive or harmful language targeting individuals or groups. The goal of this project is to automatically identify and classify such text using models like RNN, LSTM, and BERT. By applying modern NLP tools, the system can help improve online safety and content moderation.

## 2. Learning Objectives

• To understand tokenization and text vectorization techniques in NLP.

• To learn how RNN, LSTM, and Transformer models work for text classification.

• To evaluate model accuracy and understand ethical issues in hate speech detection.

## 3. Timeline

Start Date: October 31, 2025

Submission Date: October 31, 2025

## 4. Algorithm Used

This project uses Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and the Transformer-based BERT model. RNNs process sequential data, while LSTMs overcome long-term dependency issues using memory cells. BERT uses the Transformer encoder and self-attention to understand the meaning of words from both directions of context.

## 5. Tools & Libraries

Programming Language: Python

Libraries Used: TensorFlow, Keras, NumPy, Pandas, Matplotlib, and the Hugging Face Transformers library for implementing BERT.

## 6. Dataset Description

The dataset contains text samples labeled as hate speech, offensive, or neutral. Each entry is preprocessed by removing stop words, punctuation, and converting text to lowercase. Tokenization converts text into numerical sequences for model input.

## 7. Methodology

Text Preprocessing: The raw text is cleaned, tokenized, and converted to padded sequences. The vocabulary is built using Keras Tokenizer, and embeddings like Word2Vec or BERT tokens are used.

Model Training: RNN and LSTM models are trained to classify text as hate or non-hate. BERT is fine-tuned on the dataset for higher accuracy.

Evaluation: Models are evaluated using accuracy, F1-score, and confusion matrix.

## 8. Results

The LSTM model achieved about 89% accuracy, while BERT achieved around 94%. The confusion matrix showed strong performance in identifying hate speech and neutral comments correctly. The attention mechanism in BERT helped improve context understanding.

## 9. Questions Answered

* Q1: What is tokenization in NLP?
* Q2: How does BERT model work?
* Q3: What is the Attention Mechanism?
* Q4: How are RNN and LSTM different?
* Q5: What are word embeddings?
* Q6: What is padding and why is it needed?
* Q7: How to handle imbalanced text data?
* Q8: What metrics are used to evaluate NLP models?
* Q9: How does self-attention improve context understanding?
* Q10: What is fine-tuning in BERT?
* Q11: What is the confusion matrix?
* Q12: How to prevent overfitting in text models?
* Q13: What are ethical concerns in hate speech detection?
* Q14: What preprocessing steps are applied to text?
* Q15: Why is BERT better than traditional RNNs for NLP?

## 10. Challenges & Improvements

Challenges: Handling imbalanced data and dealing with subtle hate expressions were challenging. Training large models like BERT also required high computational power.

Improvements: Future work can include using multilingual datasets, data augmentation, and fine-tuning larger Transformer models like RoBERTa or GPT.

## 11. References

Dataset: Kaggle – Hate Speech and Offensive Language Dataset

Documentation: Hugging Face Transformers, TensorFlow, and Scikit-learn

Articles: Research papers on BERT and Ethical NLP from Towards Data Science

## 12. GitHub Link

https://github.com/Aravindh-2727/ml-9.Hate-Speech-project