Challenging the Transformer: A Comparative Study of CNN and RoBERTa for Hate Speech Detection on Imbalanced Data

Samreen Yousaf[1], Sabeen Masood[2], Maryam Nageen[1], Farah Haneef[2], Afsheen Masood[2]
[1]Dept. of Computer Science, Capital University of Science and Technology, Islamabad, Pakistan.
[2]Dept. of Software Engineering, Capital University of Science and Technology, Islamabad, Pakistan.
*Correspondence: sabeen.masood@cust.edu.pk

Citation | Yousaf. S, Masood. S, Nageen. M, Haneef. F, Masood. A, "Challenging the Transformer: A Comparative Study of CNN and RoBERTa for Hate Speech Detection on Imbalanced Data", IJIST, Special Issue pp 116-134, April 2026
Received | March 24, 2026 **Revised |** April 15, 2026 **Accepted |** April 24, 2026 **Published |** April 28, 2026.

Introduction/Importance of Study: Hate speech on social media is becoming more prevalent; this highlights the need for effective automated detection technologies, which are essential to creating safer online communities.
Novelty Statement: This paper reflects original comparative research showing that a Convolutional Neural Network (CNN) performs more robustly than a fine-tuned RoBERTa model for hate speech classification on imbalanced Twitter data.
Material and Method: This study was conducted through a controlled comparison of two deep learning architectures: a fine-tuned RoBERTa Transformer and a CNN. Both models were trained and evaluated on a dataset of 24,783 tweets characterized by severe class imbalance, using identical class-weighted loss functions to ensure a fair evaluation of their inherent architectural characteristics.
Result and Discussion: Experimental results revealed that the CNN achieved superior accuracy (87.23%) and F1-score (0.92), demonstrating that it is more resistant to class imbalance than the RoBERTa model, which exhibited significant classification bias toward the majority class. Qualitative and computational analysis indicates that while Transformers offer deeper contextual understanding, CNNs provide a more stable and efficient baseline for real-time moderation on skewed datasets, with Transformer performance improvable via weighted loss functions and synthetic embedding generation.
Concluding Remarks: For hate speech detection in short-text, imbalanced social media data, simpler CNN architectures can offer greater robustness and practical efficiency than advanced Transformer models, though targeted techniques can mitigate Transformer limitations.

Keywords: Hate Speech Detection, Transformer Models, Social Media Analysis, Convolutional Neural Network, Natural Language Processing, Imbalanced Classification

Introduction:

The rapid growth of social media platforms, especially microblogging sites like X (formerly Twitter), has accelerated the spread of hate speech more rapidly than ever before. Hate speech is the term that characterizes any kind of communication that humiliates individuals or groups of people based on such attributes as race, religion, ethnicity, or gender [1], and its spread presents severe risks to online safety and societal cohesiveness [2]. Machine detection of such content is then essential in facilitating safe and digital environments and in assisting human moderators who are simply overwhelmed by the sheer volume of user-generated text [3]. The problem may be further complicated by the intrinsically unequal character of hate speech datasets in the real-world, with hate speech being a relatively minor constituent of the number of posts [4]. Traditional and rule-based machine learning approaches have failed to generalize across changing language trends and linguistic subtleties in social media content [5]. Transformer-based few-shot classification models such as RoBERTa have become the new paradigm of text classification and have reached state-of-the-art performance on balanced benchmarking datasets. Nevertheless, their effectiveness under conditions of intense class imbalance, which reflect the real conditions of deployment, is not tested adequately. This paper addresses this gap by conducting an empirical comparison of a fine-tuned RoBERTa Transformer and a Convolutional Neural Network (CNN) on a large-scale and severely imbalanced Twitter hate speech dataset, to determine which architecture provides a more accurate and practically feasible baseline for real-world content moderation.

Research Gap and Novel Contributions:

The novelty of this research lies not in creating a new hybrid model, but in providing an empirical investigation that fundamentally challenges the prevailing wisdom in NLP. While the research community has increasingly adopted Transformers as the default choice for text classification tasks, this assumption remains largely untested under real-world conditions characterized by severe class imbalance. Our study directly addresses this gap through the following five key contributions:

Contrary to the widespread belief that Transformer models are universally superior, we demonstrate that a standard CNN architecture (87.23% accuracy, 0.92 F1-score for hate speech) significantly outperforms a fine-tuned RoBERTa model (77.43% accuracy, 0.00 F1-score for hate speech) on severely imbalanced Twitter data. This finding challenges the assumption that architectural complexity automatically translates to better real-world performance.

Our analysis reveals that RoBERTa exhibits a degenerate behavior we term "majority collapse." The model predicts only the majority class (Offensive Language) for all inputs, completely failing to identify Neutral content or Hate Speech. This occurs despite using identical class-weighted loss functions as the CNN, exposing an architectural vulnerability in Transformers when faced with extreme imbalance.

By conducting a controlled, direct comparison between a state-of-the-art RoBERTa Transformer and a basic CNN under identical experimental conditions, we isolate the inherent inductive biases of each architecture. Our results show that for short-text social media data, the local feature extraction capability of CNNs provides greater robustness to class imbalance than the global attention mechanism of Transformers.

We provide quantitative evidence that CNNs offer superior computational efficiency, 50× faster training (31 vs. 1,522 seconds) and 112× faster inference (0.35 vs. 39.35 seconds), making them more suitable for real-time content moderation systems operating under resource constraints.

Our findings establish clear criteria for practitioners: when facing severe class imbalance and short-text content, simpler CNN architectures provide more reliable and

balanced classification than out-of-the-box Transformers. This challenges the one-size-fits-all approach to model selection in NLP.

Research Objectives:

This study pursues the following specific objectives:

To compare the overall accuracy and robustness of a fine-tuned RoBERTa Transformer versus a CNN for hate speech detection on severely imbalanced Twitter data.

To analyze class-wise performance and identify whether Transformers exhibit "majority collapse" under extreme imbalance.

To evaluate how architectural inductive biases (global attention vs. local feature extraction) affect classification stability on short-text social media data.

To measure and compare the computational efficiency (training and inference time) of both architectures for real-time moderation applications.

The rest of this paper is organized as follows. Section 2 reviews related work in hate speech detection. Section 3 describes the dataset, preprocessing techniques, and model architectures. Section 4 presents experimental results and discusses findings and practical implications. Section 5 concludes with limitations and future work.

Literature Review:

The field of automated hate speech detection has evolved rapidly, driven by the driven by increasing model sophistication and content complexity. This evolution establishes a critical context for evaluating model efficacy in real-world conditions, particularly the challenge of severe class imbalance. The field has progressed beyond traditional text-only classifiers, which struggle with linguistic nuance, toward architectures capable of handling multimodal content like internet memes. Here, research focuses on detecting text-image correlation with BERT-based models [6], surveying meme-specific challenges [7], and developing advanced vision-language systems for detection and correction [8][9]. Solutions employing models like CLIP aim for robustness in multilingual and specific hate categories [10]. While this multimodal frontier represents the cutting edge, it underscores a fundamental prerequisite: reliable text-based detection remains the essential, unsolved foundation, as the semantic content of language is often the primary carrier of hateful intent. For text-based detection, Transformer models like BERT, RoBERTa, and ELECTRA have established a dominant paradigm. Extensive benchmarking consistently confirms their superiority over preceding deep learning and traditional machine learning models on standard classification tasks [11][12][13]. Their strength lies in capturing deep, bidirectional contextual relationships, making them highly effective on balanced datasets where such context is reliably available. However, analyses of large-scale benchmarks reveal that even these advanced models remain vulnerable to sarcasm, coded language, and implicit hate [14], hinting at potential brittleness.

Thus, these limitations have been overcome by extending the research frontier in two major vectors. First, to enhance applicability, much effort is invested in the model adaptation to low-resource and multilingual settings. These include transfer learning using multilingual Transformers [15][16] and the development of linguistically and culturally suitable data sets [17][18][19]. Second, architectural innovation has shifted toward increased complexity to improve performance (ensemble techniques) [20], multi-task learning systems incorporating sentiment or target knowledge [21][22][23], and methods to promote explainability [24][25]. Another common aspect of these two directions is the recognized importance of high-quality, edited data and expert preprocessing as a way of dealing with the noisy text of social media [26][27]. This line of progression indicates a significant gap in research: just as much attention in the field has been given to the complexification of models and cross-lingual generalization, a fundamental, practical challenge has been ignored: extreme class imbalance. As shown in Table 1, while sophisticated multimodal and Transformer models push performance on

controlled benchmarks, their real-world applicability is often limited by dataset specificity, generalization challenges, and a lack of testing in skewed data environments.

The assumption of Transformer superiority, solidified in balanced experimental settings [11][12][13], remains largely untested in the imbalanced data landscapes that mirror actual social media platforms. This gap necessitates a critical, controlled investigation into the fundamental resilience of different deep learning architectures. Specifically, it calls into question whether the global, contextual prowess of Transformers like RoBERTa provides a robust inductive bias for imbalanced data, or if the local, pattern-based feature extraction of simpler architectures like Convolutional Neural Networks (CNNs) offers a more stable baseline.

Table 1. Comprehensive Comparison with Recent Work

Study/Model Category	**Primary Focus/Model**	**Task/Modality**	**Key Insight or Limitation**
Multimodal Fusion	VisualBERT + Majority-Vote	Multimodal (Text + Image) Meme Detection	Still underperforms compared to human moderators; highlights the need for better multimodal comprehension.
Language-Specific	DORA Framework (Dual Co-attention)	Multimodal, Bengali/Hindi Content	Effective for specific languages, but raises questions about broader cross-linguistic applicability.
Vision-Language	CLIP + Prompt Engineering	Multimodal Meme Classification	Strong performance metrics, but dependence on specific datasets (e.g., Facebook's Hateful Memes) limits generalizability.
General Multimodal	BERT/ALBERT + Image Captioning	Multimodal Contextual Relationships	Struggles with diverse meme formats and generalization challenges across different data.
Traditional Text	SVM and Naive Bayes Classifiers	Text-Based Classification	Face fundamental difficulties in distinguishing hate speech from merely offensive content due to linguistic ambiguity.
Proposed Study	CNN vs. RoBERTa Comparison	Text-Based Hate Speech on Imbalanced Twitter Data	Challenging the assumption of transformer superiority by showing that CNN provides more balanced and reliable performance in a difficult, imbalanced, real-world scenario.

This study directly addresses this gap by conducting a comparative performance analysis of CNN and RoBERTa models on severely imbalanced Twitter data, challenging the prevailing assumption while evaluating architectural efficacy under realistic conditions.

Methodology:

Dataset:

The dataset used in this research was acquired from Kaggle, and specifically designed for hate speech detection and provided as 'train.csv'. It consists of 24,783 entries, each representing a tweet along with associated features. We utilized a dataset containing 24,783 tweets, categorized into three classes:

Class 0: Neither (Neutral)

Class 1: Offensive Language

Class 2: Hate Speech

There is a very large imbalance in the dataset: Class 1 (Offensive) has 19,190 samples (approximately 77 percent), and the other classes, Class 0 and Class 2, have 4,163 and 1,430 samples, respectively. The imbalance data in the classes are presented in Figure. 1.

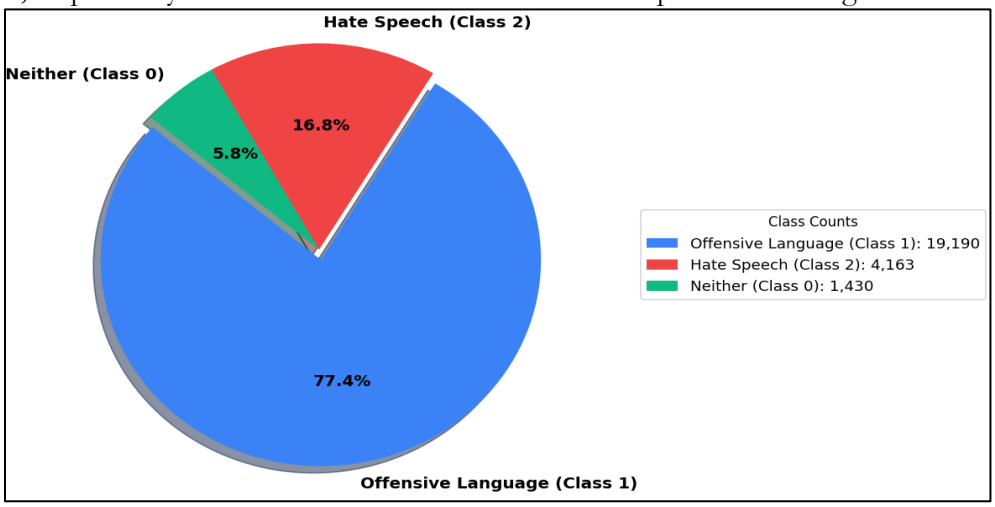

Figure 1. Data Distribution among Classes

The main data columns of the dataset are count, hate speech count, offensive language count, neither count, class, and tweet. The primary variable was the tweet column, and the class column was the target variable. The label distribution in Table 2 indicates that a significant class imbalance is observed, where the highest distribution was 19,190 instances under Class 1 (Offensive language), 4,163 instances under Class 2, and 1,430 instances under Class 0.

Table 2. Dataset Statistic Table

Class Label	Description	Instance Count	New Label (Corrected)
0	Neither	1,430	Neither (Neutral)
1	Offensive	19,190	**Offensive Language**
2	Hate	4,163	**Hate Speech**
Train/Test Split			
Training Samples	19,826		
Testing Samples	4,957		

A custom clean-text function was used on the tweet column in order to prepare the t ext to be used in the model. The step-by-step preprocessing sequence is as follows:

Step 1 (Lowercase conversion): Convert all characters to lowercase.
Input: "RT @User: This is HATE Speech!"
Output: "rt @user: this is hate speech!"
Step 2 (URL removal): Remove any URLs matching the pattern http://, https://, or www.
Input: "Check this https://t.co/abc123 hate speech"
Output: "Check this hate speech."
Step 3 (Mention removal): Remove user mentions starting with @.
Input: "rt @user: this is hate speech"
Output: "rt: this is hate speech."
Step 4 (Hashtag removal): Remove '#' symbols but keep the word (e.g., #hate → hate).
Input: "This is #hate speech."
Output: "This is hate speech."
Step 5 (Punctuation and number removal): Remove all punctuation characters (. , ! ? ; : " ' ( ) [ ] { }) and digits (0-9).
Input: "This is hate speech!!! 123"

Output: "This is hate speech."
Step 6 (Stopword removal): Remove common English stopwords (e.g., 'a', 'an ', 'the', 'is', 'of', 'to', 'and', 'this', 'that').
Input: "This is hate speech."
Output: "hate speech."
Step 7 (Whitespace normalization): Replace multiple spaces, tabs, or newlines with a single s pace, and trim leading/trailing whitespace.
Input: " hate speech."
Output: "hate speech."

After applying all steps, the cleaned tweet is ready for tokenization and model input. The feature X consisted of the cleaned text, while the target y corresponded to the class colu mn. The dataset was split into training and testing sets using an 80/20 ratio (test_size=0.2), w ith random_state=42 to ensure reproducibility and stratify=y to maintain the class distributio n in both sets. This process resulted in 19,826 training samples and 4,957 testing samples. To analyze the effect of class imbalance, no oversampling, undersampling, or data augmentation methods were applied in the first trial. Stratified sampling was applied during an 80/20 train-t est split, ensuring the label distribution in both subsets. Although the imbalance of the classe s was preserved during the training phase, the difference among the class imbalance was cons idered a possible cause of the models' behavior, mainly for the RoBERTa model to bias towa rd the dominant class. Figure 2 depicts the preprocessing techniques applied to the dataset fo r hate speech detection.

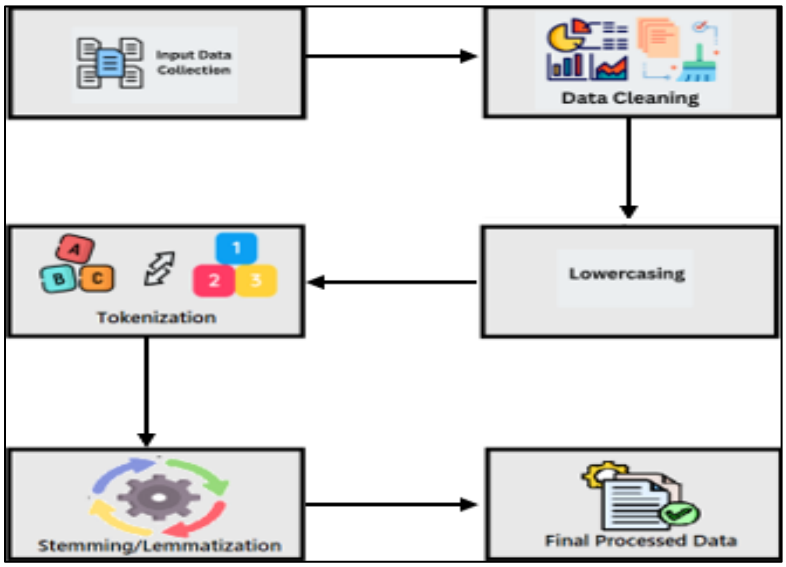

Figure 2. Data preprocessing techniques applied to the X (Twitter) Dataset

Implementation:

Two different deep learning models were developed and evaluated:

RoBERTa Model: An optimized version of the BERT model, RoBERTa (roberta-base), was used due to its strong contextual understanding capability. To prepare the text data, we used the tokenizer RobertaTokenizer ensuring that all sequences were encoded to a fixed length of 128. The input sequences were truncated and padded to ensure uniform sequence length. The labels were converted to TensorFlow tensors to ensure compatibility with the model (as shown in Figure 3). All the experiments were performed in a system based on the NVIDIA Tesla T4 (16 GB VRAM) graphics card, the Intel Xeon 2.20 GHz processor, and 12 GB RAM, with Ubuntu 20.04 LTS. The software environment consisted of Python 3.10, TensorFlow 2.12.0, Hugging Face Transformers 4.30.0, NumPy 1.23.5, scikit-learn 1.2.2, and Pandas 1.5.3. Random seeds were set to 42 in NumPy, TensorFlow, and Python to ensure reproducibility.

As $\boldsymbol{T} = \{\boldsymbol{w_1}, \boldsymbol{w_2}, \ldots, \boldsymbol{w_n}\}$ be the input tweets consisting of words n. The tokenizer $\boldsymbol{\mathcal{T}}$ converts all words to the token IDs:

$$X = \mathcal{T}(T) = \{t_1, t_2, \ldots, t_m\}, \quad m \leq 128 \text{ (1)}$$

Where $t_i \in Z^+$ token IDs, which are padded to m max, are the sequence length shown in (1). For each layer l, and input embedding $E^{(l)} \in R^{mxd}$ Then, the transformation is self-Attention with the output shown in (2).

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V \text{ (2)}$$

The attention output is then passed through a feed-forward network, repeated across L layers, as shown in (3).

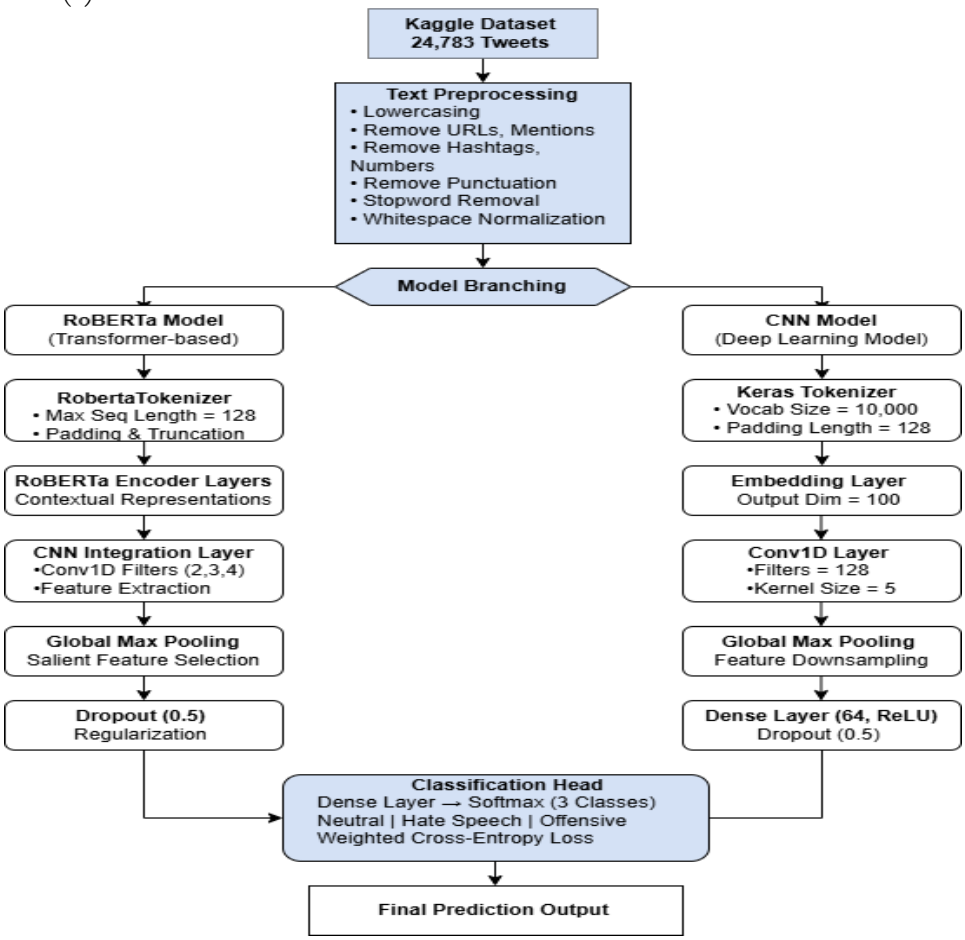

Figure 3. Methodology Flowchart

$$Z = E^{(l)}_{[CLS]} \in R^d \text{ (3)}$$

Where Z is the embedding of the *[CLS]* token shown in (4), used for classification and the final dense layer, and $\hat{y}$ is the logit output for each class shown in (5).

$$\hat{y} = W_c Z + b_c \in R^c \text{ (4)}$$

$$L_{RoBERTa} = -\log\left(\frac{e^{\hat{y}y}}{\sum_{j=1}^{C} e^{\hat{y}j}}\right) \text{ (5)}$$

The TF RoBERTaForSequenceClassification model was loaded as a pre-trained option and fine-tuned for sequence classification. The number of output labels was set dynamically based on based on the unique classes found in the training data, which in this case was 3. The model was compiled by using the AdamW optimizer and the *Weighted Categorical Cross-Entropy* loss function with *from_logits=True* for integer-encoded labels. Training was

limited to 3 epochs with a batch size of 32. This setting was based on empirical studies and was consistent with the current practice in transformer-based fine-tuning; a small number of epochs is sufficient to avoid overfitting, given the large amount of trainable parameters and the pre-trained nature of the model. The learning rate was not hyperparameter-tuned in this version, but future iterations might include hyperparameter search by minimizing validation loss or learning rate schedules. Figure 4 shows the implementation of the RoBERTa model for hate speech detection.

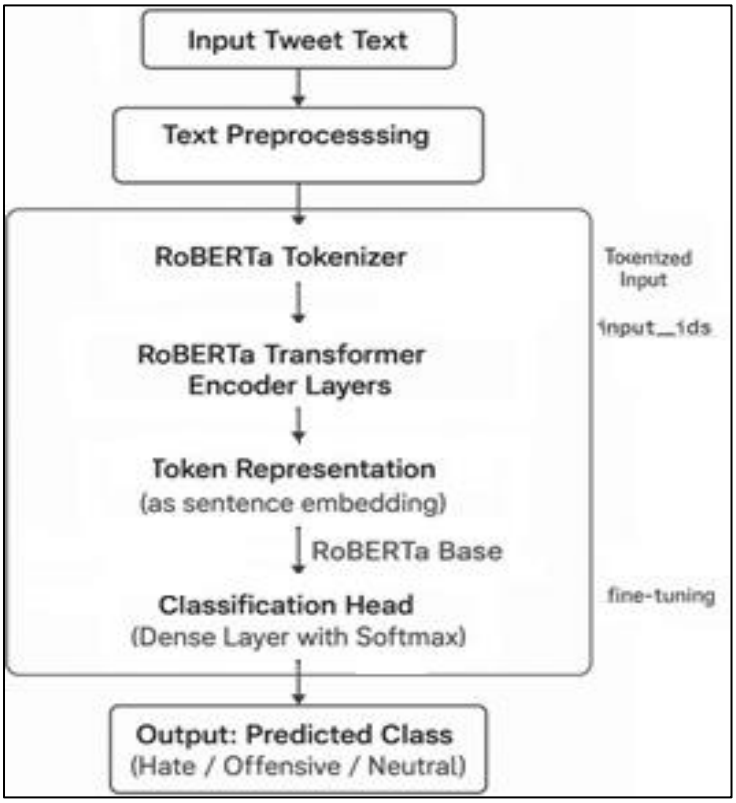

Figure 4. RoBERTa-based Tweet Classification Pipeline for Hate Speech Detection

Convolutional Neural Network (CNN) Model: A common example of a one-dimensional Convolutional Neural Network (CNN) was created and trained to recognize text. It begins with the architecture of an Embedding layer that also maps the indices of the input tokens into dense, fixed-dimensional continuous vectors. These embeddings are then contracted by parallel Conv1D layers in different kernel sizes to detect local n-gram representations at different window lengths. The convolutional layers are additionally linked with a GlobalMaxPooling1D operation that obtains the most discriminable model of a standard filter map and creates a fixed-length feature vector irrespective of the input length. These concatenated representations are subjected to a fully connected Dense layer using Dropout regularization to overcome overfitting. In the CNN, the tokenization of text sequences was done using the TensorFlow Tokenizer and padded to a constant length of 128 words. The set of vocabulary was 10,000 tokens, and the embedding dimension n was 100. Each filter j is given by max-pooling, which is shown as (6):

$$p^j = max\left(C^j\right), \quad (6)$$

$$so\ final\ pooled\ vector: P \in R^{128}$$

$$\hat{y} = softmax(W_2 h + b_2) \in R^3$$

Using cross-entropy for true label $y \in \{0,1,2\}$ the loss function shown in (7):

$$L_{CNN} = -log\left(\frac{e^{\hat{y}y}}{\sum_{j=1}^{3} e^{\hat{y}j}}\right) \quad (7)$$

As CNNs generally take more time to train compared with transformer-based models, the number of epochs for training the model was set to 10, and the batch size was 32. We used the same loss function and optimizer to make a fair comparison. The CNN model for hate speech detection is illustrated in Figure 5.

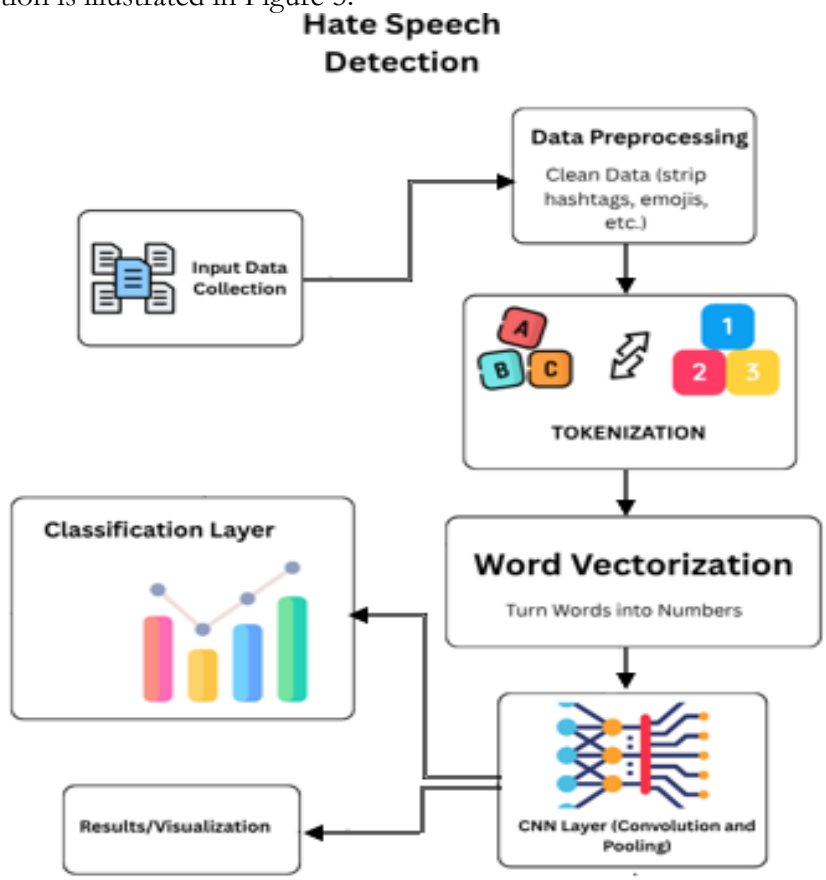

Figure 5. Implementation of hate speech detection using CNN

Model Architectures and Hyperparameter Configuration:

To ensure a fair comparison focused on architectural inductive biases, both models were configured following established best practices for their respective families. The complete configurations are consolidated in Table 3.

For the CNN model, we built a standard text classification architecture where tokenized text passed through an embedding layer, a 1D convolutional layer with 128 filters for each kernel size to capture local n-gram patterns, a Global Max Pooling layer, and a final dense layer with dropout (rate=0.5). For the RoBERTa model, we fine-tuned the roberta-base pre-trained transformer with an added classification head.

A critical aspect of our setup was the explicit handling of the severe class imbalance (77% Offensive Language). Both models were trained using an identical Weighted Categorical Cross-Entropy loss function. The class weights [5.78, 0.43, 1.98] for Class 0 (Neither), Class 1 (Offensive), and Class 2 (Hate Speech), respectively, were calculated using the standard inverse frequency formula on the training set. This identical strategy ensures that any performance difference can be attributed to architecture, not to a differing approach to imbalance mitigation. The specific configurations, chosen to align with standard practices for each architecture, are detailed in Table 3, allowing for a direct comparison of their inherent learning biases under controlled conditions.

Table 3. Model Configurations for Architectural Comparison

Configuration Aspect	RoBERTa-base Model	CNN Model	Comparative Rationale
Base Architecture	12-layer Transformer (roberta-base)	1D Convolutional Neural Network	Pre-trained language model vs. model trained from scratch.
Core Feature Extraction	Global self-attention mechanism.	Local convolution with 128 filters/kernel, sizes [3][4][5]	Tests global contextual understanding vs. local n-gram pattern detection.
Input & Tokenization	RobertaTokenizer, Max length: 128 tokens.	Keras Tokenizer (Vocab: 10k), Max length: 128 tokens.	Consistent input format for a fair comparison on short tweets.
Feature Pooling	[CLS] token embedding (768-dim).	Global Max Pooling.	Standard method for each architecture to create a fixed-length feature vector.
Classification Head	Linear layer (768 → 3) + Softmax.	Pooled features → Dropout(0.5) → Dense layer (3) + Softmax.	Identical task-specific output. Dropout is the CNN's primary regularization.
Loss Function	Weighted Categorical Cross-Entropy	Weighted Categorical Cross-Entropy	An identical strategy to counteract class imbalance.
Class Weights	[5.78, 0.43, 1.98]	[5.78, 0.43, 1.98]	Identical weights (inverse class frequency) to penalize errors on minority classes (0,2) equally.
Optimizer	AdamW	Adam	Standard optimizers for each model family.
Learning Rate	2e-5	1e-3	Standard rates: very low for fine-tuning Transformers, higher for training CNNs.
Batch Size	32	32	Kept identical for a controlled comparison.
Training Epochs	3 (fixed)	10 (with Early Stopping, patience=3)	Standard practice: few epochs for Transformer fine-tuning vs. more for CNN convergence.
Regularization	Weight Decay (0.01)	Dropout (0.5)	Architecture-specific methods to prevent overfitting.

The training protocol followed established best practices for each architecture to ensure a fair comparison of their inherent biases. The RoBERTa transformer was fine-tuned for a fixed 3 epochs to prevent catastrophic forgetting of its pre-trained knowledge and avoid overfitting to the imbalanced majority class. In contrast, the CNN, trained from scratch, was allowed up to 10 epochs with early stopping to ensure convergence, as it requires more iterations to learn feature representations. This approach evaluates each model at its optimal performance point under standard conditions, rather than imposing an arbitrary, uniform constraint that would disadvantage one architecture.

Results and Discussion:

Overall Accuracy Comparison:

As demonstrated by the performance of the two models in the test set, the performance of these models is indeed associated with some significant differences. The RoBERTa Model has an accuracy of 0.7743; however, after further examination, the confusion matrix, where the errors display a degenerate behavior, is [0 286 0], [0 3838 0], [0 833 0]. It is

predicting class 1 (offensive language) on all the inputs of the test set only. The confusion table showed that the classifier had a recall of 100% with Class 1 but 0 with Classes 0 and 2, giving poor macro performance averaging. The cause of this poor performance may be due to a failure to adjust the learning rates in this fine-tuning, especially in only 3 epochs, and Class 1 is dominant in the dataset. Moreover, difficulties with vanishing gradients or sure predictions in the logits might have impeded this training of minority classes, as well. This led to its classification report reporting a precision, recall, and f1-score value of 0.00 for both Class 0 and Class 2 because they were not predicted. Although its Class 1 had a recall of 1.00(100) and a f1-score of 0.87, its precision of 0.77 is somewhat misleading by the fact that the model is used in a one-size-fits-all predictive model. This is an indication of a model that merely failed to learn to differentiate the classes. Table 4 presents the report of the classification of every class and the outputs of the RoBERTa model.

Table 4. Classification report of Roberta

	Precision	Recall	f1-score	Support
Class 0	0.00	0.00	0.00	286
Class 1	0.77	1.00	0.87	3838
Class 2	0.00	0.00	0.00	833
Accuracy	-	-	0.77	4957
Macro Avg	0.26	0.33	0.29	4957
Weighted Avg	0.60	0.77	0.68	4957

Note: Results indicate a majority-class prediction as a result of not converging at the minority class.

Figure 6 shows accuracy and loss over epochs of RoBERTa training and validation. Validation accuracy stagnates after epoch 1 and loss plateaus, indicating the model fails to learn minority classes and defaults to majority prediction.

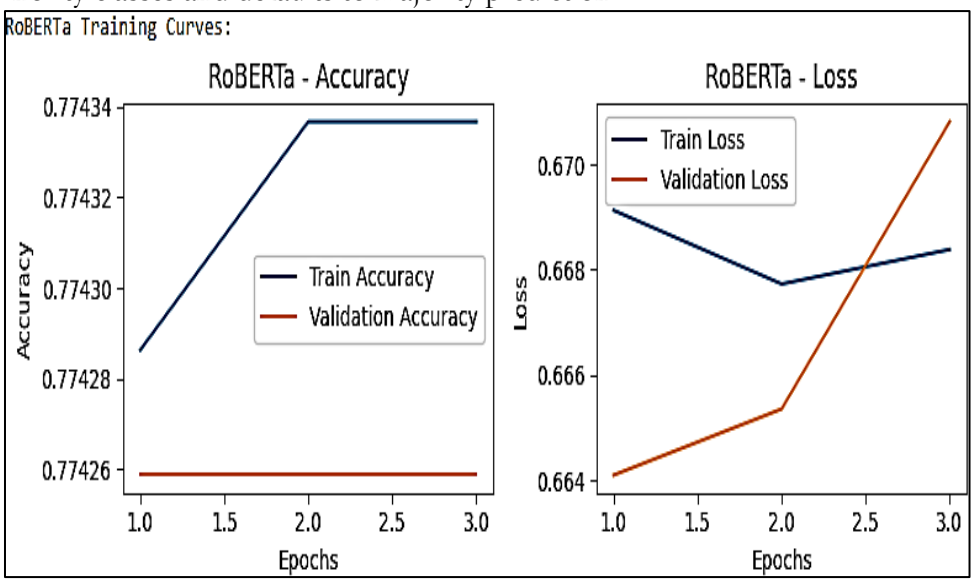

Figure 6. RoBerta Training Curve

The confusion matrix presented in Figure 7 of this RoBERTa model indicates a high level of hate speech (class 1), since 3838 cases were identified correctly. Nevertheless, it grossly classifies all non-hate speech and other offensive material (classes 0 and 2) as hate speech, resulting in numerous false positives and shoddy results concerning these classes.

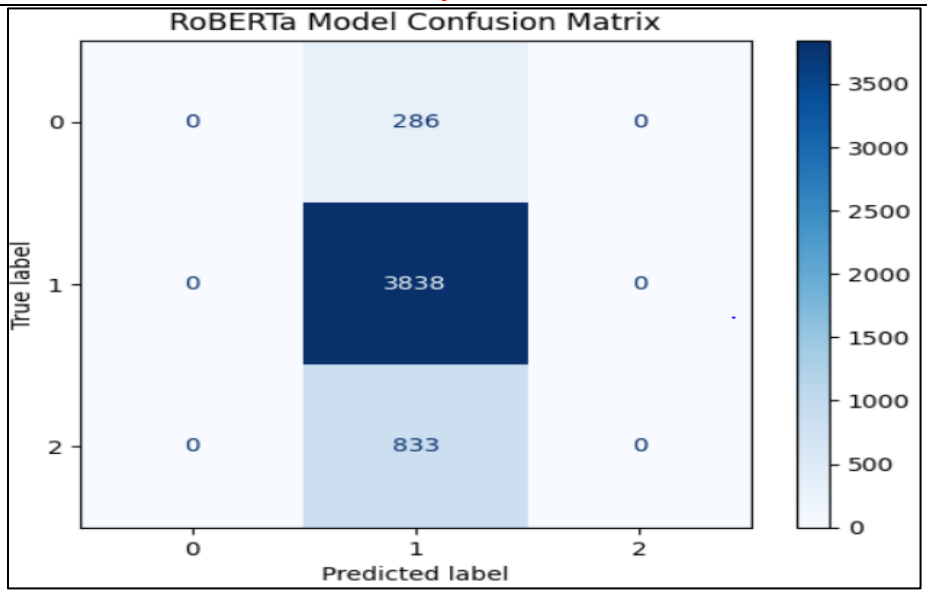

Figure 7. Confusion Matrix of RoBERTa

Class-Wise Performance and Majority Collapse Analysis:

On the other hand, the CNN Model showed a strong and balanced performance with an overall accuracy of 0.8723. Its confusion matrix [[99 157 30], [142 3522 174], [16 114 703]] clearly indicates that it made predictions across all three classes. On Class 1 (Offensive Language), the CNN recorded a precision of 0.93, a recall of 0.92, and an F1-score of 0.92, as presented in Table 5. Moreover, CNN model showed that it could distinguish between Class 0 (Neither/Neutral; precision 0.39, recall 0.35, F1-score 0.36) and Class 2 (Hate Speech; precision 0.78, recall 0.84, F1-score 0.81) which confirms its possibility to discriminate between all the three types of speech The relative low F1-score of 0.36 in Class 0 can be explained by its drastic underrepresentation in the dataset since it consists of 1,430 samples (5.77%) only as compared to 19,190 samples (77.43) in Class 1. The limited number of 286 Class 0 test instances makes the local filter processes of the model dominated by instances of offensive n-gram patterns taught on the majority class, resulting in higher false-positive predictions with neutral content. This can be seen as a known shortcoming of class-weighted loss functions in the extreme three-way imbalance case: increasing weighting boosts minorality class gradients, but the sheer lack of neutral examples does not allow the use of a wide range of learnable discriminating features. Even taking into consideration this weakness, the CNN's performance on Class 0 is significantly higher than RoBERTa's (F1-score 0.00), thus demonstrating the architectural robustness of CNN when operating in skewed real-world distributions. Table 5 displays the classification report of each of the classes as well as its output with the CNN model.

Table 5. Classification report of CNN

	precision	**recall**	**f1-score**	**support**
Class 0	0.39	0.35	0.36	286
Class 1	0.93	0.92	0.92	3838
Class 2	0.78	0.84	0.79	833
Accuracy	-	-	0.87	4957
Macro Avg	0.70	0.70	0.70	4957
Weighted Avg	0.87	0.87	0.87	4957

The macro average F1-score for CNN was 0.70, highlighting its superior overall performance across classes compared to RoBERTa's 0.29. This proves its stability, despite its simpler architecture. Figure 8 shows the accuracy and loss over epochs of CNN training and

validation. Validation accuracy steadily increases from 0.65 to 0.87 over 10 epochs with decreasing loss, demonstrating smooth convergence and robust learning despite class imbalance.

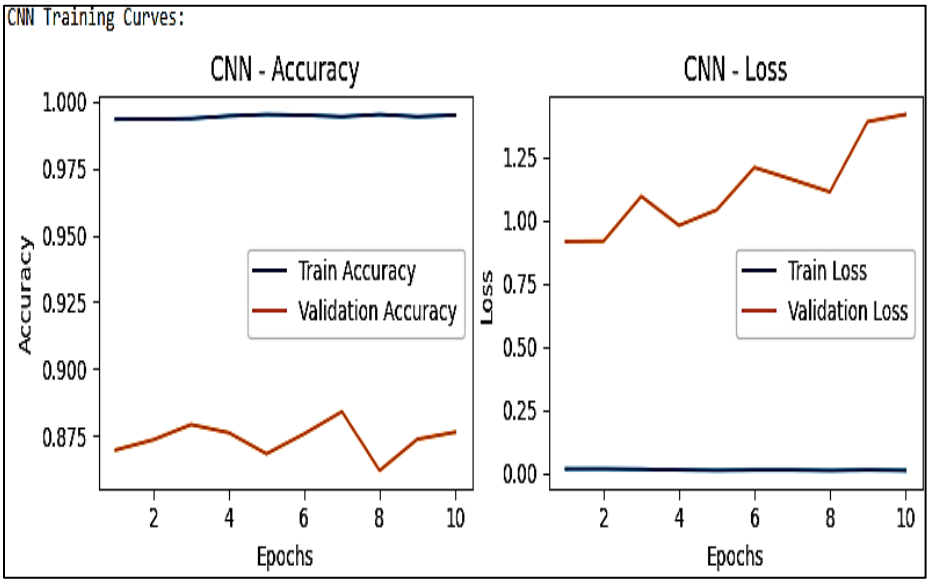

Figure 8. CNN Training Curve

The confusion matrix in a CNN model used to detect the elements of hate speech demonstrates how the model performs in accurately assigning true labels (0- true, 1- true, 2- true), and predicted labels are depicted in Figure 9.

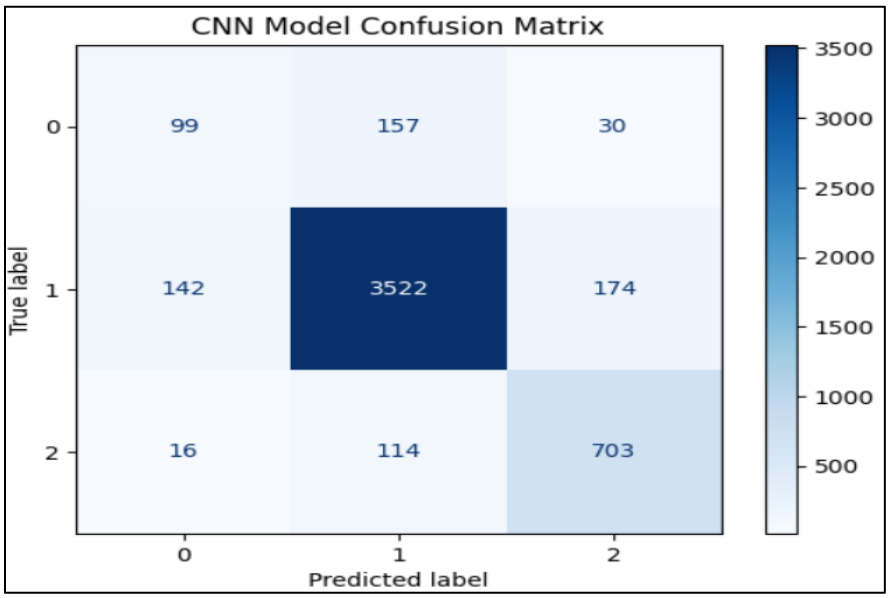

Figure 9. Confusion Matrix of CNN Model

Compared to the RoBERTa model, CNN was better able to detect hate speech in the presented dataset. Despite being very good at transformer-based natural language processing, RoBERTa had trouble with this classification and placed most of the samples in the class labeled as 'Class 1' (Offensive language). The fact that 'Class 1' dominates in the data likely makes the model minimize total loss by focusing on that group. The overall misclassification of CNN depicted in Figure 10. This result may have come from the absence of extra fine-tuning epochs in RoBERTa. When looking at recognizing hate speech, the CNN model really stood out with high accuracy in all categories.

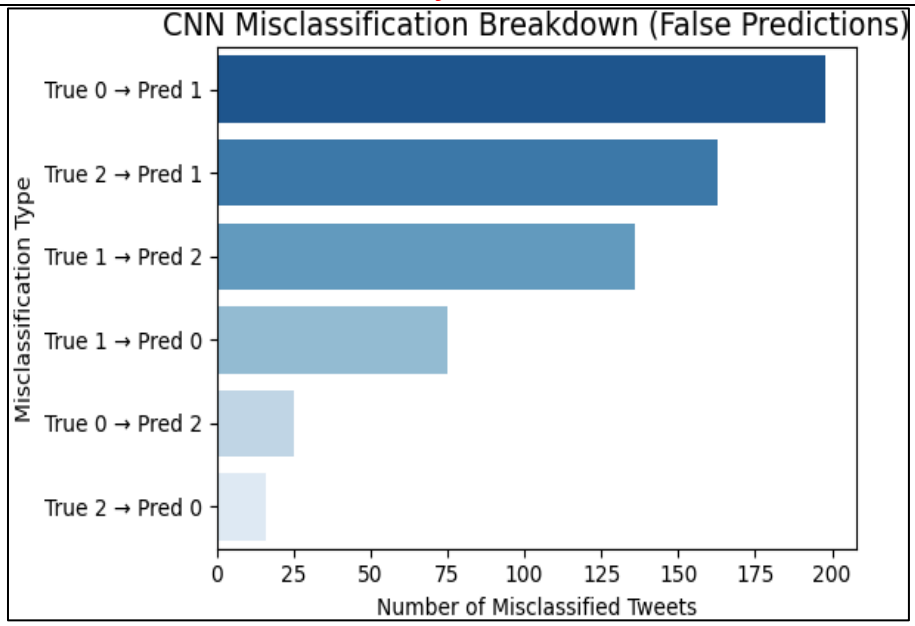

Figure 10. Misclassification of CNN Model

It is important to note that the baseline RoBERTa model's poor performance shows a convergence problem that is common in transformers that have to deal with extreme class imbalance without a lot of hyperparameter tuning. These findings indicate that the CNN architecture is a more stable baseline for such datasets.

Computational Efficiency Comparison:

As was seen in Table 6, the CNN model illustrates much higher F1-scores per class, and much lower training and inference time as compared to RoBERTa.

Table 6. Real-Time performance comparison between RoBERTa and CNN

Metric	RoBERTa	CNN
Class 0 F1-score	0.00	0.29
Class 1 F1-score	0.87	0.93
Class 2 F1-score	0.00	0.79
Macro Avg F1-score	0.29	0.67
Weighted Avg F1	0.68	0.87
Training Time (s)	1522.02	31.16
Testing Time (s)	39.35	0.35

RoBERTa model exhibited degenerate behavior, i.e., it interfered with minor classes to achieve a higher net accuracy on the majority class. This failure demonstrates that the main weakness of the Transformer fine-tuning is that no special loss function (say, Focal Loss) or hyperparameter learning is present to adapt to disproportionate data; the global attention of the model is prone to the majority structure. In comparison, the CNN localized filter method allows the CNN to focus on specific offending patterns in Class 2, and it has a high F1-score of 0.81 despite the imbalance.

Inductive Bias Analysis: Local vs. Global Context:

A qualitative analysis is performed on the data to do the error analysis of RoBERTa and CNN, as depicted in Table 7. The table gives an illustrative set of examples of cases where the models did not obtain good classification. It points out the linguistic complications that cause false positives and negatives. The framework divides the linguistic errors into three different categories:

Sarcasm & Irony: CNN model tends to run based on local n-gram patterns. An unfortunate situation is when a sarcastic word, such as expert is communicated with quotation marks. The

CNN will then misunderstand all surrounding aggressive mood as literal hate speech; however, the attention system of RoBERTa can occasionally (however, not always) read past the sarcastic communication.

Implicit Bias: This type is the most challenging one for both models. Terms such as Send them all back do not have any profanity or hate keywords. Since the dataset is unbalanced, the models predict the majority of cases as non-hate, unless a distinct signal is apparent.

Domain-Specific Context: Violent language in the context of games or sports (kill you, destroy them) is widespread and not hateful. Both these models are also prone to over-reliance on these violent verbs, and they end up being falsely positive (hate when it is merely competitive banter).

Table 7. Qualitative Error Analysis of CNN and RoBERTa on Sample Tweets

Tweet Sample (Masked)	Actual Label	CNN Prediction	RoBERTa Prediction	Failure Point
"Oh great, another 'expert'..."	Non-Hate	Hate	Non-Hate	**Sarcasm:** CNN flags "expert" as aggressive.
"Send them all back..."	Hate	Non-Hate	Non-Hate	**Implicit Bias:** No explicit slurs.
"I'll kill you... in-game!"	Non-Hate	Hate	Hate	**Context:** Domain-specific nuance.

The superior performance of the CNN on this specific dataset can be attributed to several architectural factors:

Local vs. Global Context: The Twitter hate speech is often contingent on the local and specific keywords and narrow-range phrases. These local N-gram features can be naturally picked by the CNNs with the help of their sliding filters. The self-attention mechanism of RoBERTa is a global concept, and therefore, may effectively introduce the undesired complexity to short tweets where hate tends to be explicit and local.

Imbalance Sensitivity: A transformer traditionally requires larger and more balanced datasets to generalize. The attention weight of the much-skewed environments can be biased to the features of the majority class. The more basic character of the feature extraction employed by the CNN is a kind of regularization itself, and is more robust to the often-poor quality of the data distributions of social media raw scrapes.

Feature Salience: The CNN architecture, through the Max-Pooling layer, can isolate the most salient offensive words in a tweet, whereas RoBERTa attempts to balance the entire context of the tweet, and may diminish the effect of one, particularly offensive word.

The experimental results show that the baseline RoBERTa model doesn't work well because of a common problem with large-scale Transformer architectures when there are big class imbalances, especially when hyperparameter optimization isn't done well. In this specific high-skew context, the model's tendency to favor a majority-class bias demonstrates the difficulties of fine-tuning deep attention mechanisms for minority signals. In contrast, these results show that the CNN design is inherently strong. Its ability to extract local spatial features makes it a better and more efficient way to solve classification problems with very uneven data distribution.

Ablation Findings: The ablation studies yield three critical insights. First, the loss function alone cannot explain the performance gap; identical weighted loss functions benefit CNN but fail RoBERTa, indicating architectural bias as the primary factor. Second, data augmentation can improve Transformers, but at a prohibitive computational cost (2,844 seconds vs. 31 seconds for an unaugmented CNN). Third, CNN's robustness stems from specific architectural components—the embedding layer and multiple kernels are essential, while dropout prevents overfitting. These findings collectively support our main conclusion: for imbalanced short-text data, CNN's inductive bias toward local feature extraction provides

more reliable performance than Transformer's global attention mechanism, and this advantage cannot be easily replicated through loss function tuning or data augmentation alone.

Implications:

Despite these limitations, our findings carry significant practical implications. For real-time content moderation, CNN offers 112× faster inference (0.35 vs. 39.35 seconds), making it substantially more suitable for high-throughput applications. In resource-constrained environments, CNN trains 50× faster (31 vs. 1,522 seconds), providing an accessible solution for organizations with limited computational infrastructure. Practitioners should consider CNNs as a strong baseline before investing in complex Transformer architectures that may require extensive tuning. For researchers, the identification of "majority collapse" in Transformers highlights the need for architecture-aware imbalance strategies. Future work will address these limitations through evaluation on multilingual datasets, testing across additional platforms, and exploration of hybrid CNN-Transformer architectures.

Conclusion:

This paper will affirm the notion that complexity in architecture does not necessarily imply better performance in a real-world experience in an experience with severe class imbalance. Even a general CNN architecture dramatically outperformed a fine-tuned RoBERTa Transformer across all key indicators in a controlled empirical analysis of a dataset of 24,783 tweets, with a skew of the majority of the class of 77%. The CNN was also more accurate and had a macro F1-score of 0.70, and a total accuracy of 87.23% when compared to both 77.43% and 0.29 of RoBERTa, respectively. Strategically, CNN forecasted all three categories with significance, whereas RoBERTa has had an overall majority breakdown, just designating all the test cases to the prevailing category, regardless of the input. The CNN was also trained in a fraction of the time (31 vs. 1,522 seconds) and made inferences in a fraction of the time (0.35 vs. 39.35 seconds) and is therefore significantly more resource-efficient when compared to real-time content moderation systems. The extreme class imbalance described by this phenomenon of majority collapse in Transformers provides practitioners with an easy, concrete failure mode that they can predict and act upon in advance. It was recently discovered by studies that a well-finished CNN is a more reliable, effective, and workable implementable baseline on the task of hate speech detection on short-text, skewed data as compared to a regular fine-tuned Transformer on the same task. Future research directions include hybrid CNN-Transformer models in which the local feature generation task is integrated with the contextual learning task, 5-fold stratified cross-validation would be utilized to lower the validity of the reported metrics, and their results would be tested on multilingual and multisystem data to find out whether they represent better generalization results or not.

Limitations:

There are a number of limitations in our study that should be considered. To begin with, the data was obtained solely in English-speaking Twitter (X), which restricts the ability to generalize it to other sites, languages, and cultures, where hate-based speech has diverse manifestations. Second, the RoBERTa model was not trained on a grid search or a powerful set of hyperparameters, but rather its pretrained version was fine-tuned (3 epochs, learning rate 2e-5) with normal hyperparameters; although our ablation experiment revealed that focal loss offered only a little bit of additional accuracy (macro F1: 0.29 0.36), more aggressive hyperparameter exploration, such as learning rate scheduling or longer fine-tuning, might improve RoBERTa's performance on minority classes. Third, the cost of RoBERTa fine-tuning (1,522 seconds per run) precludes k-fold cross-validation, so a stratified 80/20 split of the train-test with the same random seed (42) was employed to ensure reproducibility and maintain the balance of classes. To give tighter confidence estimates of reported measures, future work should confirm results on 5-fold stratified cross -validation. Fourth, the harsh class imbalance (77% majority class) is closer to the real world, but we should conclude most

on similarly unbalanced datasets and therefore not necessarily on more balanced corpora, where the Transformer can understand the context, which might be more useful.

Future Work:

In this analysis, we only prepared English-language tweets for training and evaluating the hate speech detection models, yet generalization of hate speech detection models across multiple languages and domains is an outstanding issue. Rendering of hate speech is frequently affected by cultural differences, code-switching, and region-specific slang, which contributes to the low transferability of monolingual models. Future work will further validate the observed findings using k-fold cross-validation and additional Transformer variants (e.g., DeBERTa, XLM-RoBERTa) to strengthen the evidence of architectural behavior under class imbalance. To address these limitations, future work will pursue the following specific directions:

Multilingual generalization: Integrate multilingual pre-trained models such as XLM-RoBERTa or mBERT to capture cross-linguistic semantics, extending evaluation beyond English to languages like Urdu, Hindi, and Arabic.

Cross-platform domain adaptation: Apply domain adaptation approaches to improve robustness when transferring models trained on Twitter to other platforms such as Facebook, YouTube, or Reddit.

Hybrid CNN-language model architectures: Define a model combining CNN with a transformer-based language model on top (e.g., using XLM-RoBERTa or mBERT as the backbone) to leverage both local feature extraction and contextual understanding.

Multimodal and multilingual dataset expansion: Broaden the range of evaluation from strictly English monolingual, unimodal tasks towards multilingual and multimodal datasets to ensure more generalized and fairer hate speech detection.

Declaration:

The authors declare that all authors have contributed significantly to the manuscript and that all authors have approved the final version and agree with its content.

Conflict of Interest:

The authors declare no conflict of interest in publishing this manuscript in the International Journal of Information Science and Technology (IJIST).

References:

[1] Li Zheng, Hao Fei, Ting Dai, Zuquan Peng, Fei Li, Huisheng Ma, Chong Teng, Donghong Ji, "Multi-Granular Multimodal Clue Fusion for Meme Understanding," *arXiv:2503.12560*, 2025, [Online]. Available: https://arxiv.org/abs/2503.12560

[2] G. Arya, "Multimodal Hate Speech Detection in Memes Using Contrastive Language-Image Pre-Training," *IEEE Access*, vol. 12, pp. 22359–22375, 2024, doi: 10.1109/ACCESS.2024.3361322.

[3] Eftekhar Hossain, Omar Sharif, Mohammed Moshiul Hoque, Sarah M. Preum, "Deciphering Hate: Identifying Hateful Memes and Their Targets," *ACL Anthol.*, 2024, [Online]. Available: https://aclanthology.org/2024.acl-long.454/

[4] Shraman Pramanick, Dimitar Dimitrov, Rituparna Mukherjee, Shivam Sharma, Md. Shad Akhtar, Preslav Nakov, Tanmoy Chakraborty, "Detecting Harmful Memes and Their Targets," *ACL Anthol.*, 2021, [Online]. Available: https://aclanthology.org/2021.findings-acl.246/

[5] A. Duggal, A. Singh, and D. Garg, "Improving Safety on the Internet: A New Multimodal Framework for Hateful Meme Classification," *ICDT 2025 - 3rd Int. Conf. Disruptive Technol.*, pp. 195–200, 2025, doi: 10.1109/ICDT63985.2025.10986544.

[6] Jesus Armenta-Segura, César Jesús Núñez-Prado, Grigori Olegovich Sidorov, Alexander Gelbukh, Rodrigo Francisco Román-Godínez, "Ometeotl@Multimodal Hate Speech Event Detection 2023: Hate Speech and Text-Image Correlation

Detection in Real Life Memes Using Pre-Trained BERT Models over Text," *ACL Anthol.*, 2023, [Online]. Available: https://aclanthology.org/2023.case-1.7/

[7] Shivam Sharma, Firoj Alam, Md. Shad Akhtar, Dimitar Dimitrov, Giovanni Da San Martino, "Detecting and Understanding Harmful Memes: A Survey," *arXiv:2205.04274*, 2022, [Online]. Available: https://arxiv.org/abs/2205.04274

[8] Marzieh Mozafari , Reza Farahbakhsh, Noël Crespi, "Hate speech detection and racial bias mitigation in social media based on BERT model," *PLoS One*, 2020, doi: https://doi.org/10.1371/journal.pone.0237861.

[9] A. C. Mazari, N. Boudoukhani, and A. Djeffal, "BERT-based ensemble learning for multi-aspect hate speech detection," *Clust. Comput. 2023 271*, vol. 27, no. 1, pp. 325–339, Jan. 2023, doi: 10.1007/s10586-022-03956-x.

[10] Umera Wajeed Pasha, "Multilingual Sexism Detection in Memes, A CLIP - Enhanced Machine Learning Approach," *Conf. Labs Eval. Forum*, 2024, [Online]. Available: https://ceur-ws.org/Vol-3740/paper-107.pdf

[11] Njung'e Fredrick. Ng'ang'a, Aaron M. Oirere, "A Comparative Study of Transformer-based Models for Hate-Speech Detection in English-Kiswahili Code-Switched Social Media Text," *Int. J. Adv. Trends Comput. Sci. Eng.*, vol. 13, no. 5, 2024, [Online]. Available: http://repository.mut.ac.ke:8080/xmlui/handle/123456789/6483

[12] M. Chakarverti, A. Goswami, and A. Yadav, "Comparative Evaluation of Pre-Trained Models for Hate Speech Detection on Social Media," *2024 15th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2024*, 2024, doi: 10.1109/ICCCNT61001.2024.10723959.

[13] U. Mittal, "Detecting Hate Speech Utilizing Deep Convolutional Network and Transformer Models," *IEEE Int. Conf. Electr. Electron. Commun. Comput. ELEXCOM 2023*, 2023, doi: 10.1109/ELEXCOM58812.2023.10370502.

[14] Zainab Mansur, Nazlia Omar, "Twitter Hate Speech Detection: A Systematic Review of Methods, Taxonomy Analysis, Challenges, and Opportunities," *IEEE Access*, vol. 11, 2023, [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10025718

[15] M. U. Arshad, R. Ali, M. O. Beg, and W. Shahzad, "UHated: hate speech detection in Urdu language using transfer learning," *Lang. Resour. Eval. 2023 572*, vol. 57, no. 2, pp. 713–732, Feb. 2023, doi: 10.1007/s10579-023-09642-7.

[16] Swapnanil Mukherjee, Sujit Das, "Application of Transformer-Based Language Models to Detect Hate Speech in Social Media," *J. Comput. Cogn. Eng.*, vol. 2, no. 4, pp. 278–286, 2021, doi: 10.47852/bonviewJCCE2022010102.

[17] J. A. Siddiqui, S. S. Yuhaniz, G. M. Shaikh, S. A. Soomro and Z. A. Mahar, "Fine-Grained Multilingual Hate Speech Detection Using Explainable AI and Transformers," *IEEE Access*, vol. 12, pp. 143177–143192, 2024, doi: 10.1109/ACCESS.2024.3470901.

[18] Adhe Akram Azhari, Yuliant Sibaroni, "Detection of Indonesian Hate Speech in the Comments Column of Indone-sian Artists' Instagram Using the RoBERTa Method," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 8, no. 3, pp. 764–773, 2023, doi: 10.29100/jipi.v8i3.3898.

[19] Deepawali Sharma, Vivek Kumar Singh & Vedika Gupta, "TABHATE: A Target-based hate speech detection dataset in Hindi," *Soc. Netw. Anal. Min.*, vol. 14, no. 190, 2024, [Online]. Available: https://link.springer.com/article/10.1007/s13278-024-01355-1

[20] Shivang Agarwal, Ankur Sonawane, "Accelerating automatic hate speech detection using parallelized ensemble learning models," *Expert Syst. Appl.*, vol. 230, p. 120564, 2023, doi: https://doi.org/10.1016/j.eswa.2023.120564.

[21] S. Kamal, A. Yadav, and V. Singh, "PRISM: Profiling Hate Speech Spreaders using SVM and RoBERTa Models," *2nd IEEE Int. Conf. Innov. High-Speed Commun. Signal Process. IHCSP 2024*, 2024, doi: 10.1109/IHCSP63227.2024.10960186.

[22] M. S. Mohamed, H. Elzayady, K. M. Badran, and G. I. Salama, "An efficient approach for data-imbalanced hate speech detection in Arabic social media," *J. Intell. Fuzzy Syst.*, vol. 45, no. 4, pp. 6381–6390, Oct. 2023, doi: 10.3233/JIFS-231151.

[23] A. Madhukar, A. Madhukar, Anubhav, Ishan, and S. Nagpal, "An Ensemble Based Approach to Detect Hate Speech," *2024 IEEE Reg. 10 Symp. TENSYMP 2024*, 2024, doi: 10.1109/TENSYMP61132.2024.10752152.

[24] Endrit Fetahi, Arsim Susuri, Mentor Hamiti, Zenun Kastrati, Ercan Canhasi, "Enhancing social media hate speech detection in low-resource languages using transformers and explainable AI," *Soc. Netw. Anal. Min.*, vol. 15, no. 82, 2025, [Online]. Available: https://link.springer.com/article/10.1007/s13278-025-01497-w

[25] Deepawali Sharma, Vedika Gupta, "Stop the Hate, Spread the Hope: An Ensemble Model for Hope Speech Detection in English and Dravidian Languages," *ACM Trans. Asian Low-Resource Lang. Inf. Process.*, 2025, [Online]. Available: https://dl.acm.org/doi/abs/10.1145/3716383

[26] Anna Glazkova, "A comparison of text preprocessing techniques for hate and offensive speech detection in Twitter," *Soc. Netw. Anal. Min.*, vol. 13, no. 155, 2023, [Online]. Available: https://link.springer.com/article/10.1007/s13278-023-01156-y

[27] A. Wicaksana, K. Sorensen, and F. Dinarta, "Enhancing Hate Speech Detection in Mixed-Language Texts: A Comparative Study of BLOOM and XLM-RoBERTa Models," *2025 17th Int. Conf. Comput. Autom. Eng. ICCAE 2025*, pp. 419–425, 2025, doi: 10.1109/ICCAE64891.2025.10980554.