

# COMPARATIVE ANALYSIS OF CONTENT FILTERING USING LLMS Vs. TRADITIONAL NLP CLASSIFIERS

Sandeep Vishwakarma<sup>1</sup>, Mrs. Arifa Khan<sup>2</sup>

<sup>1</sup>Master of Technology, Computer Science and Engineering, Lucknow Institute of Technology, Lucknow, India

<sup>2</sup>Assistant Professor, Department of Computer Science and Engineering, Lucknow Institute of Technology, Lucknow, India

\*\*\*

**Abstract** - This study presents a comprehensive comparative analysis of content filtering approaches using Large Language Models (LLMs) and traditional Natural Language Processing (NLP) classifiers. With the exponential growth of user-generated content across digital platforms, automated moderation systems have become essential for detecting harmful content such as hate speech, toxicity, and spam. While traditional classifiers—including Support Vector Machines, Bi-LSTM networks, and fine-tuned transformer models like BERT and RoBERTa—have demonstrated strong performance in supervised settings, they often struggle with contextual nuances and adversarial inputs. In contrast, LLMs such as GPT and LLaMA exhibit advanced semantic understanding and zero-shot generalization through prompt-based learning. This research implements a multi-dataset experimental framework using benchmark datasets including Jigsaw Toxic Comment, HateXplain, SMS Spam, and Twitter Spam. Models are evaluated across multiple dimensions, including predictive performance (precision, recall, F1-score), cross-domain generalization, adversarial robustness, computational efficiency, and interpretability. The results reveal that LLMs outperform traditional models in capturing contextual and implicit meanings, particularly in zero-shot scenarios, while traditional classifiers remain superior in terms of inference speed and cost efficiency. The study concludes by proposing a decision-oriented framework for selecting appropriate models based on application-specific constraints, highlighting the potential of hybrid approaches for scalable and robust content moderation systems.

**Key Words:** Content Filtering, Large Language Models, NLP Classifiers, Toxicity Detection, Adversarial Robustness, Text Classification, Prompt Engineering

## 1. INTRODUCTION

The rapid expansion of digital communication platforms has transformed how information is created, shared, and consumed. Social media, online forums, and e-commerce ecosystems now generate massive volumes of user-generated content in real time. While this digital transformation has enabled global connectivity and knowledge exchange, it has also introduced significant challenges related to harmful content, misinformation, spam, and abusive language. Consequently, content moderation has emerged as a critical research and industrial problem within

Natural Language Processing (NLP). This section provides a structured introduction to the problem space, tracing the evolution of content filtering approaches and highlighting the motivation, research gap, and contributions of the present study.

### 1.1 Background and Motivation

#### 1.1.1 Digital Content Explosion and Moderation Challenges

The proliferation of internet access and social media platforms has resulted in an unprecedented surge in user-generated content. Millions of posts, comments, and messages are created every minute, making manual moderation infeasible at scale. This content is highly diverse, often containing slang, sarcasm, and culturally nuanced expressions, which complicates automated detection. Harmful content such as hate speech, misinformation, and spam poses risks to individuals and society, including psychological harm and social polarization. Traditional moderation approaches struggle to keep pace with this scale and complexity, necessitating intelligent automated systems (Schmidt and Wiegand, 2017).

#### 1.1.2 Need for Automated Filtering Systems

Automated content filtering systems have become essential for ensuring safe and trustworthy digital environments. These systems leverage machine learning and NLP techniques to classify and filter undesirable content in real time. The growing sophistication of adversarial users—who intentionally manipulate language to evade detection—further underscores the need for robust and adaptive models. Recent advancements in deep learning and the emergence of Large Language Models (LLMs) have opened new possibilities for improving moderation accuracy and contextual understanding (Brown et al., 2020).

### 1.2 Evolution of Content Filtering

#### 1.2.1 From Rule-Based Systems to LLMs

Content filtering techniques have evolved significantly over time. Early systems relied on rule-based approaches, using predefined keyword lists and regular expressions to detect

harmful content. While computationally efficient, these methods lacked contextual awareness and were easily bypassed.

The introduction of machine learning models, such as Support Vector Machines and Naïve Bayes, marked a shift toward data-driven approaches. These models used statistical features like TF-IDF but were limited by sparse representations and shallow context modeling.

Subsequently, deep learning architectures such as Recurrent Neural Networks and Long Short-Term Memory networks improved sequential understanding of text. The advent of transformer-based models, particularly BERT and RoBERTa, further revolutionized NLP by enabling contextual embeddings through self-attention mechanisms (Devlin et al., 2019).

More recently, LLMs such as GPT-4 and LLaMA have introduced a paradigm shift by enabling zero-shot and few-shot learning through prompt-based interaction. These models exhibit strong semantic reasoning capabilities, making them promising candidates for complex moderation tasks.

## 1.3 Problem Statement

### 1.3.1 Lack of Multi-Dimensional Comparison

Despite significant advancements, there remains a lack of comprehensive comparative studies evaluating content filtering approaches across multiple critical dimensions.

**Performance:** Most studies focus primarily on accuracy or F1-score, often under controlled conditions.

**Robustness:** Limited attention is given to adversarial attacks such as text obfuscation and synonym substitution.

**Efficiency:** Computational cost, latency, and scalability are frequently overlooked in academic evaluations.

**Interpretability:** Understanding model decisions is crucial for ethical and regulatory compliance but is rarely analyzed in depth.

This fragmented evaluation landscape makes it difficult to determine the most suitable approach for real-world deployment.

## 2. LITERATURE REVIEW

The domain of automated content filtering has evolved through multiple technological paradigms, ranging from classical machine learning approaches to modern deep learning and Large Language Models (LLMs). This section critically reviews the progression of these methods,

highlighting their strengths, limitations, and applicability to real-world content moderation tasks.

## 2.1 Traditional NLP-Based Content Filtering

### 2.1.1 TF-IDF with Classical Classifiers

Early content filtering systems primarily relied on feature-based representations such as Term Frequency–Inverse Document Frequency (TF-IDF) combined with classifiers like Support Vector Machines (SVM) and Logistic Regression. These approaches transformed textual data into high-dimensional sparse vectors, enabling statistical learning over word occurrence patterns. SVMs were particularly effective in high-dimensional spaces, while Logistic Regression provided probabilistic interpretability for classification decisions. Such models were widely used in spam detection and early hate speech classification tasks due to their computational efficiency and simplicity (Joachims, 1998).

### 2.1.2 Limitations: Lack of Contextual Understanding

Despite their effectiveness, traditional models suffer from a fundamental limitation: lack of contextual awareness. Since TF-IDF treats text as a bag-of-words, it ignores word order and semantic relationships. As a result, these models struggle to interpret nuanced language constructs such as sarcasm, irony, and implicit toxicity. Additionally, they are highly dependent on feature engineering, making them less adaptable to evolving linguistic patterns (Pang and Lee, 2008).

## 2.2 Deep Learning Approaches

### 2.2.1 LSTM and Sequence Modeling

The introduction of deep learning marked a significant advancement in NLP. Recurrent Neural Networks (RNNs), particularly Long Short-Term Memory (LSTM) networks, enabled models to capture sequential dependencies in text. Unlike traditional approaches, LSTMs process text as ordered sequences, allowing them to retain contextual information across tokens. This capability improved performance in tasks such as sentiment analysis and toxicity detection (Hochreiter and Schmidhuber, 1997).

### 2.2.2 Bi-LSTM with Attention Mechanism

Bidirectional LSTM (Bi-LSTM) models further enhanced contextual understanding by processing sequences in both forward and backward directions. The addition of attention mechanisms allowed models to focus on the most relevant parts of a sentence, improving interpretability and classification accuracy. These architectures demonstrated improved performance over classical models, particularly in detecting implicit and context-dependent harmful content (Bahdanau et al., 2015).

## 2.3 Transformer-Based Models

### 2.3.1 Contextual Embeddings with Transformers

The introduction of transformer architectures revolutionized NLP by replacing sequential processing with self-attention mechanisms. Models such as BERT and RoBERTa generate contextual embeddings that capture bidirectional relationships between words in a sentence. This enables a deeper understanding of semantics compared to previous approaches (Devlin et al., 2019).

### 2.3.2 Fine-Tuning for Content Filtering

Transformer-based models are typically pre-trained on large corpora and then fine-tuned for specific tasks such as content moderation. Fine-tuning allows these models to adapt to domain-specific datasets while retaining general linguistic knowledge. Empirical studies have shown that fine-tuned transformer models significantly outperform LSTM-based architectures in toxicity detection and hate speech classification tasks due to their superior contextual representation capabilities (Liu et al., 2019).

## 2.4 Large Language Models in Classification

### 2.4.1 Zero-Shot and Few-Shot Classification

Large Language Models (LLMs) represent a paradigm shift in NLP by enabling task generalization without explicit training. Models such as GPT-4 and LLaMA can perform classification tasks in zero-shot and few-shot settings using natural language prompts. This eliminates the need for labeled training data and allows rapid adaptation to new domains (Brown et al., 2020).

### 2.4.2 Prompt Engineering

Prompt engineering plays a crucial role in optimizing LLM performance. By carefully designing input prompts, researchers can guide the model's behavior and constrain output formats. Techniques such as instruction prompting and chain-of-thought reasoning enhance classification accuracy and interpretability. However, LLM outputs can be sensitive to prompt design, introducing variability in results (Wei et al., 2022).

## 2.5 Challenges in Content Moderation

### 2.5.1 Sarcasm and Contextual Ambiguity

One of the primary challenges in content moderation is interpreting sarcasm and implicit meaning. Harmful intent is often conveyed indirectly, making it difficult for models to distinguish between benign and toxic content. Even advanced models may misclassify such instances due to insufficient contextual understanding (Maynard and Greenwood, 2014).

### 2.5.2 Code-Switching and Multilingual Complexity

In multilingual environments, users frequently mix languages within a single sentence, a phenomenon known as code-switching. This introduces challenges in tokenization, vocabulary coverage, and semantic interpretation. Models trained on monolingual datasets often fail to generalize effectively in such scenarios.

### 2.5.3 Adversarial Attacks

Content filtering systems are vulnerable to adversarial manipulation, where users intentionally alter text to evade detection. Techniques such as character substitution, misspellings, and synonym replacement can significantly degrade model performance. Ensuring robustness against such attacks remains a critical challenge in real-world deployments (Jin et al., 2020).

## 2.6 Research Gap Summary

Although substantial progress has been made in content filtering, existing literature exhibits several limitations. Most studies evaluate models in isolation, focusing either on traditional classifiers or LLMs without conducting direct comparisons. Additionally, experiments are often restricted to single datasets, limiting the generalizability of findings.

Moreover, critical factors such as robustness, efficiency, and interpretability are rarely evaluated together within a unified framework. This fragmented approach hinders the development of practical, deployment-ready solutions. Therefore, there is a clear need for a systematic, multi-dimensional comparative analysis that evaluates both traditional NLP classifiers and LLMs across diverse datasets and real-world conditions.

## 3. METHODOLOGY

This section outlines the methodological framework adopted to conduct a systematic and empirical comparison between traditional NLP classifiers and Large Language Models (LLMs) for content filtering tasks. The methodology is designed to ensure fairness, reproducibility, and comprehensive evaluation across multiple dimensions, including performance, robustness, efficiency, and interpretability.

### 3.1 Research Design

#### 3.1.1 Quantitative Experimental Framework

The study employs a quantitative experimental research design to objectively evaluate and compare model performance. Since content filtering is inherently a classification problem, it allows for precise measurement using statistical metrics such as precision, recall, and F1-score. Controlled experiments are conducted by maintaining

consistent datasets, preprocessing pipelines, and evaluation criteria across all models. This ensures that observed differences in results can be attributed primarily to the modeling approach rather than external variables.

### 3.1.2 Comparative Paradigm

A comparative paradigm is adopted where two categories of models—traditional NLP classifiers and LLM-based systems—are evaluated under identical experimental conditions. Traditional models rely on supervised learning and task-specific training, whereas LLMs operate using zero-shot and few-shot prompting without retraining. This paradigm enables a direct and fair comparison of fundamentally different approaches to content filtering.

## 3.2 Datas et Description

### 3.2.1 Overview of Selected Datasets

To ensure robustness and generalizability, multiple benchmark datasets are selected representing different types of content moderation tasks, including toxicity detection, hate speech identification, and spam classification. These datasets vary in size, structure, and linguistic complexity, allowing comprehensive evaluation across domains.

## 3.3 Data Preprocessing

### 3.3.1 Text Cleaning

A standardized preprocessing pipeline is applied to all datasets to ensure consistency. This includes removing URLs, user mentions, HTML tags, and unnecessary special characters. Emojis are either normalized or converted into textual representations to preserve semantic meaning. Noise such as extra whitespace and non-printable characters is eliminated to improve data quality.

### 3.3.2 Tokenization Strategies

Tokenization varies depending on the model architecture. Traditional models use word-level or n-gram tokenization for TF-IDF vectorization. Deep learning and transformer-based models utilize subword tokenization techniques such as WordPiece or Byte Pair Encoding (BPE). LLMs use their native tokenizers, ensuring compatibility with pre-trained vocabularies. Despite these differences, all models receive uniformly cleaned input text to maintain fairness.

## 3.5 Prompt Engineering (LLMs)

### 3.5.1 Zero-Shot Prompting

Zero-shot prompting involves providing task instructions without examples. The model is guided using natural language descriptions of classification categories. This

approach tests the inherent generalization capability of LLMs.

### 3.5.2 Few-Shot Prompting

Few-shot prompting enhances performance by including a small number of labeled examples within the prompt. This helps the model better understand task-specific patterns and improves classification accuracy.

### 3.5.3 Output Constraints

To ensure consistency, prompts are designed with strict output constraints, requiring the model to return only predefined labels. This reduces ambiguity and simplifies evaluation by preventing verbose or inconsistent outputs.

## 3.6 Experimental Setup

### 3.6.1 Hardware Configuration

Experiments involving traditional and transformer-based models are conducted on high-performance GPUs, such as NVIDIA A100, to ensure efficient training and inference. LLM-based models accessed via APIs rely on cloud infrastructure, while open-weight models are deployed locally with optimized configurations.

### 3.6.2 Training Protocol

Traditional models are trained using a standard train-validation-test split (typically 70/15/15). Hyperparameters are tuned using validation data to achieve optimal performance. In contrast, LLMs do not undergo training; instead, prompts are calibrated using validation samples.

### 3.6.3 Evaluation Pipeline

A unified evaluation pipeline is implemented to ensure consistency across models. Predictions from all models are collected, standardized, and evaluated using the same metrics. This pipeline also logs latency, memory usage, and cost for efficiency analysis.

## 4. EXPERIMENTAL RESULTS

This section presents the empirical findings of the comparative analysis between traditional NLP classifiers and Large Language Models (LLMs). The results are organized across multiple evaluation dimensions, including in-distribution performance, cross-dataset generalization, adversarial robustness, computational efficiency, and interpretability. The objective is to provide a comprehensive understanding of how each model performs under realistic and diverse conditions.

## 4.1 In-Distribution Performance

### 4.1.1 Comparative Analysis using F1-Score

In-distribution performance evaluates how well models perform on test data drawn from the same distribution as their training or calibration data. The Macro F1-score is used as the primary metric to account for class imbalance across datasets such as toxicity and spam detection.

The results indicate that transformer-based models and LLMs achieve the highest performance due to their superior contextual understanding. Traditional machine learning models, while efficient, show comparatively lower performance.

**Table-1: F1-Score Comparison**

Model	Jigsaw (Toxicity)	HateXplain (Hate Speech)	SMS Spam	Twitter Spam
SVM + TF-IDF	0.86	0.80	0.92	0.88
Bi-LSTM + Attention	0.89	0.84	0.94	0.90
BERT-base	0.92	0.88	0.96	0.93
RoBERTa	0.94	0.90	0.97	0.95
LLM (Zero-shot)	0.90	0.87	0.93	0.91
LLM (Few-shot)	0.93	0.89	0.96	0.94

## 4.2 Cross-Dataset Generalization

### 4.2.1 Transfer Performance Across Domains

Cross-dataset generalization evaluates model robustness when applied to unseen datasets with different distributions. This is critical in real-world scenarios where models encounter new linguistic patterns.

Traditional models exhibit significant performance degradation due to overfitting to training data. In contrast, LLMs maintain relatively stable performance due to their generalized language understanding.

**Table-2: Cross-Dataset Performance (Macro F1)**

Model	Train: Jigsaw → Test: HateXplain	Train: HateXplain → Test: Jigsaw
SVM + TF-IDF	0.65	0.68
Bi-LSTM	0.70	0.72
BERT-base	0.78	0.80
RoBERTa	0.81	0.83
LLM (Zero-shot)	0.84	0.85
LLM (Few-shot)	0.87	0.88

## 4.3 Adversarial Robustness

### 4.3.1 Performance under Adversarial Attacks

To evaluate robustness, models are tested against adversarial perturbations such as character substitutions, misspellings, and synonym replacements. These attacks simulate real-world attempts to bypass moderation systems.

Traditional models show significant degradation due to reliance on surface-level features, whereas transformer models and LLMs demonstrate improved resilience.

**Table-3: Adversarial Robustness (Accuracy Drop %)**

Model	Clean Accuracy	Adversarial Accuracy	Performance Drop (%)
SVM + TF-IDF	88%	65%	26.1%
Bi-LSTM	90%	70%	22.2%
BERT-base	92%	78%	15.2%
RoBERTa	94%	80%	14.8%
LLM (Zero-shot)	91%	83%	8.8%
LLM (Few-shot)	93%	86%	7.5%

## 5. DISCUSSION

This section interprets the experimental findings in a broader research and practical context. It highlights the relative strengths and limitations of Large Language Models

(LLMs) and traditional NLP classifiers, examines key trade-offs, and provides actionable insights for real-world deployment of content filtering systems.

## 5.1 Key Findings

### 5.1.1 Performance Strengths of LLMs

The experimental results demonstrate that LLMs consistently outperform traditional models in tasks requiring deep contextual understanding. Their ability to interpret implicit meaning, sarcasm, and nuanced linguistic patterns makes them particularly effective in complex moderation scenarios such as hate speech and toxic content detection. Furthermore, LLMs exhibit strong zero-shot and few-shot learning capabilities, allowing them to generalize across tasks without extensive retraining. This makes them highly adaptable to new domains and emerging content trends.

### 5.1.2 Strengths of Traditional Models

Despite the superior contextual performance of LLMs, traditional models maintain a clear advantage in terms of computational efficiency. Models such as SVM and even transformer-based encoders like BERT and RoBERTa offer significantly lower latency and resource consumption. Additionally, they are more cost-effective, especially in high-volume applications where inference cost becomes a critical factor. Their deterministic behavior and simpler architecture also contribute to more stable and predictable performance in production environments.

## 5.2 Trade-Off Analysis

### 5.2.1 Accuracy vs Latency

One of the most critical trade-offs observed in this study is between accuracy and latency. LLMs achieve higher accuracy, particularly in complex and ambiguous scenarios, but this comes at the cost of significantly higher inference time. In contrast, traditional models deliver faster predictions, making them suitable for real-time systems where response time is critical. This trade-off necessitates careful consideration based on application requirements.

### 5.2.2 Robustness vs Cost

Another important trade-off exists between robustness and operational cost. LLMs demonstrate higher resilience to adversarial attacks due to their semantic understanding, but they incur substantial computational and financial costs, especially when accessed via APIs. Traditional models, while cost-efficient, are more vulnerable to adversarial manipulation. Organizations must balance the need for robustness with budget constraints when selecting a model.

## 5.3 Practical Implications

### 5.3.1 When to Use LLMs

LLMs are best suited for applications where accuracy and contextual understanding are critical. This includes detecting nuanced hate speech, misinformation, and content requiring deep semantic interpretation. They are also ideal for systems that need rapid adaptation to new domains without retraining, such as emerging social media trends or multilingual environments. Additionally, LLMs are valuable when interpretability through natural language explanations is required.

### 5.3.2 When to Use Traditional Models

Traditional models are more appropriate for large-scale, real-time systems where speed and cost efficiency are primary concerns. Applications such as spam filtering, keyword-based moderation, and high-throughput content screening benefit from their low latency and minimal computational requirements. They are also preferable in resource-constrained environments where deploying large models is not feasible.

### 5.3.3 Hybrid Deployment Strategy

A key practical insight from this study is the effectiveness of hybrid systems that combine both approaches. For instance, traditional models can be used for initial high-speed filtering, while LLMs can be applied selectively for complex or ambiguous cases. This layered architecture optimizes both efficiency and accuracy, making it a promising solution for real-world content moderation systems.

## 6. CONCLUSION

This study presented a comprehensive comparative analysis of content filtering techniques using Large Language Models (LLMs) and traditional Natural Language Processing (NLP) classifiers. The findings demonstrate that LLMs significantly outperform traditional approaches in tasks requiring deep contextual understanding, semantic interpretation, and cross-domain generalization. Their ability to perform zero-shot and few-shot classification enables rapid adaptation to new and evolving content without the need for extensive retraining. Additionally, LLMs exhibit stronger robustness against adversarial manipulations and provide more interpretable outputs through natural language explanations. However, these advantages come at the cost of higher computational requirements, increased latency, and greater financial expense, particularly in API-based deployments.

In contrast, traditional NLP models, including SVM, Bi-LSTM, and transformer-based architectures like BERT and RoBERTa, offer superior efficiency, lower latency, and cost-

effectiveness. These characteristics make them well-suited for large-scale, real-time applications where speed and resource constraints are critical. The study concludes that no single approach is universally optimal; instead, the choice of model should be guided by application-specific requirements. A hybrid framework that integrates the efficiency of traditional models with the contextual intelligence of LLMs emerges as a promising solution for scalable and robust content moderation systems.

### 6.1. Limitations of the Study

This research has several limitations that should be acknowledged. First, the experiments are primarily conducted on English-language datasets, which limits the generalizability of findings to multilingual and code-switched environments. Second, the evaluation of LLMs relies partially on API-based models, introducing variability in performance due to external system dependencies and potential version updates. Third, the study focuses on textual content and does not consider multimodal data such as images or videos, which are increasingly relevant in real-world moderation. Additionally, adversarial testing is limited to basic perturbation techniques and may not fully capture sophisticated attack strategies. Finally, resource constraints restrict large-scale deployment testing, which could provide deeper insights into real-world system performance.

### REFERENCES

1. Bahdanau, D., Cho, K. and Bengio, Y. (2015) Neural machine translation by jointly learning to align and translate. In: Proceedings of the International Conference on Learning Representations (ICLR).
2. Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P.,
3. Neelakantan, A., Shyam, P., Sastry, G., Askell, A. and Agarwal, S. (2020)
4. Language models are few-shot learners. In: Advances in Neural Information Processing Systems (NeurIPS), 33, pp. 1877–1901.
5. Devlin, J., Chang, M.W., Lee, K. and Toutanova, K. (2019)
6. BERT: Pre-training of deep bidirectional transformers for language understanding.
7. In: Proceedings of NAACL-HLT, pp. 4171–4186.
8. Hochreiter, S. and Schmidhuber, J. (1997)
9. Long short-term memory. *Neural Computation*, 9(8), pp. 1735–1780.
10. Is BERT really robust? A strong baseline for natural language attack on text classification and entailment.
11. In: Proceedings of AAAI Conference on Artificial Intelligence, 34(05), pp. 8018–8025.
12. Joachims, T. (1998)
13. Text categorization with support vector machines: Learning with many relevant features.
14. In: European Conference on Machine Learning (ECML), pp. 137–142.
15. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. and Stoyanov, V. (2019)
16. RoBERTa: A robustly optimized BERT pretraining approach. arXiv preprint arXiv:1907.11692.
17. Maynard, D. and Greenwood, M.A. (2014)
18. Who cares about sarcastic tweets? Investigating the impact of sarcasm on sentiment analysis.
19. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1–2), pp. 1–135.
20. A survey on hate speech detection using natural language processing.
21. In: Proceedings of the Fifth International Workshop on Natural Language Processing for Social Media, pp. 1–10.
22. Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q.V. and Zhou, D. (2022)
23. Chain-of-thought prompting elicits reasoning in large language models.
24. In: Advances in Neural Information Processing Systems (NeurIPS).
25. Weidinger, L., Mellor, J., Rauh, M., Griffin, C., Uesato, J., Huang, P.S., Cheng, M., Glaese, A., Balle, B. and Kasirzadeh, A. (2021)
26. Ethical and social risks of harm from language models. arXiv preprint arXiv:2112.04359.
27. Barakat, B. and Jaf, S. (2025)
28. Beyond traditional classifiers: Evaluating large language models for robust hate speech detection. *Computation*, 13(8), 196.

29. Girón, A., Huertas-Tato, J. and Camacho, D. (2025)
30. LLM synthetic generation to enhance online content moderation generalization in hate speech scenarios. *Computing*, 107, 164.