Feature Engineering in the Transformer Era: A Controlled Study on Toxic Comment Classification

Zhanyi Ding , Zijing Wei , Chao Yang , Hailiang Wang , Shuo Xu , Yixiang Li , Xuanjie Chen *

Posted Date: 16 December 2025

doi: 10.20944/preprints202512.1277.v1

Keywords: toxicity detection; natural language processing; transformer; ALBERT; feature engineering; TF-IDF; logistic regression; random forest; LightGBM; imbalanced classification; bootstrap confidence intervals; McNemar test; sentiment and readability features; feature fusion

Article

Feature Engineering in the Transformer Era: A Controlled Study on Toxic Comment Classification

Zhanyi Ding [1], Zijing Wei [2], Chao Yang [3], Hailiang Wang [4], Shuo Xu [5], Yixiang Li [6] and Xuanjie Chen [7,*]

[1] Center For Data Science, New York University, NY, USA
[2] College of Liberal Arts & Sciences, University of Illinois at Urbana-Champaign, Urbana, IL, USA
[3] Duke University, Durham, NC, USA
[4] School of Computer Science, College of Computing, Georgia Institute of Technology, Atlanta, GA, USA
[5] Computer Science and Engineering Department, University of California San Diego, La Jolla, USA
[6] Department of Computer Science, The George Washington University, Washington, DC, USA
[7] Department of Applied Mathematics, University of Washington, Seattle, WA, USA
* Correspondence: xjay0202@gmail.com

Abstract

Detecting toxic language in user-generated text remains a critical challenge due to linguistic nuance, evolving expressions, and severe class imbalance. While Transformer-based models have established state-of-the-art performance, their significant computational costs pose scalability barriers for real-time moderation. We investigate whether integrating social and contextual metadata—such as user reactions and platform ratings—can bridge the performance gap between computationally efficient classical models and modern deep learning architectures. Using a 40,000-comment subset of the Jigsaw Toxic Comment Classification Challenge, we conduct a controlled, two-phase comparison. We evaluate a Baseline configuration (TF-IDF for classical ensembles vs. raw text for ALBERT) against an Enhanced configuration that fuses text representations with explicit social signals. Our investigation analyzes whether these high-fidelity metadata features allow lightweight models (e.g., LightGBM) to rival the discriminative power of deep Transformers. The findings challenge the prevailing assumption that deep semantic understanding is strictly necessary for high-performance toxicity detection, offering significant implications for the design of scalable, "Green AI" moderation systems.

Keywords: toxicity detection; natural language processing; transformer; ALBERT; feature engineering; TF-IDF; logistic regression; random forest; LightGBM; imbalanced classification; bootstrap confidence intervals; McNemar test; sentiment and readability features; feature fusion

1. Introduction

Online platforms have invested heavily in automated moderation to mitigate toxic speech, yet the problem persists [1]. Abusive language is context-dependent, rapidly evolving, and heavily imbalanced in real data streams [2]. Classical text classifiers—most often logistic regression and tree-based ensembles trained on TF-IDF n-grams—remain attractive for their efficiency and transparency, but their reliance on surface-level cues limits their robustness to paraphrase, sarcasm, and obfuscation. In contrast, Transformer models learn deep contextual representations that capture long-range dependencies and subtle semantics, establishing them as the dominant approach [3]. This ML-versus-DL trade-off is a central, active area of research for analogous social media tasks, such as detecting fake news [4,5], spam instant messages [6], deception [7], mental health crises [8–10], offensive or hate speech identification [11,12] and disaster response and recovery [13–15].

What remains unclear is whether rich social and contextual metadata (e.g., user reactions, platform ratings, discussion context) still add value in the Transformer era—a question that fuels

 doi:10.20944/preprints202512.1277.v1

emerging work on "hybrid" models [16]. Critically, it is unknown whether these external signals benefit modern deep models differently than classical pipelines, and if they can help bridge the performance gap between efficient classical models and heavy deep learning architectures.

We address this question with a controlled, two-phase comparison on a 40,000-comment subset of the Jigsaw Toxic Comment Classification Challenge [17]. In the Baseline configuration, classical models (Logistic Regression, Random Forest, LightGBM) consume TF-IDF features, while ALBERT (albert-base-v2) is fine-tuned on tokenized text. In the Enhanced configuration, we incorporate a rich set of metadata, including user reactions (e.g., 'likes', 'funny'), platform metrics, and context identifiers. These features are label-encoded and concatenated with the classical TF-IDF representation and, for ALBERT, standardized and fused with the pooled [CLS] embedding before classification. This design isolates the contribution of the metadata features while holding data splits and preprocessing constant.

We evaluate using metrics appropriate for imbalanced toxicity detection: in addition to precision and recall, Weighted F1for operating-point performance and AUC for threshold independent discrimination [18,19]. To quantify uncertainty and statistical significance, we report 1,000-iteration bootstrap confidence intervals (CIs). We deem performance differences statistically significant if their respective 95% CIs do not overlap, providing a robust, non-parametric comparison.

Our results reveal a remarkable convergence between model classes. While ALBERT dominates in the baseline text-only setting, the addition of metadata features effectively closes the performance gap. We find that classical models, when augmented with explicit social signals, achieve performance parity with fine-tuned Transformers. For instance, the enhanced LightGBM model achieves an AUC of 0.9798, rivaling ALBERT. This finding suggests that the "superiority" of deep learning in this domain is partially contingent on feature representation, and that rigorous feature engineering remains a powerful equalizer for democratizing high-performance AI.

The remainder of the paper proceeds as follows. Section 2 details the dataset, preprocessing, feature constructions, and model architectures. Section 3 reports baseline and enhanced results with statistical comparisons and interprets the observed divergence. Section 4 concludes with implications for deployment and directions for future analysis.

2. Methods

This section details the comprehensive research design for our comparative analyses. Our methodology is structured around a two-phase experiment to assess the impact of advanced feature engineering on distinct model classes. We describe: (1) the data corpus, its preparation, and the two experimental feature sets (Baseline vs. Enhanced); (2) the implementation and architecture of the classical machine learning models, Logistic Regression, Random Forest, and LightGBM, and the deep learning Transformer, ALBERT; and (3) our evaluation framework and the statistical methods used for model comparison.

2.1. Dataset and Preprocessing

We used a 40,000-comment subset of the Jigsaw Toxic Comment Classification Challenge (binary labels: 1 = toxic, 0 = non-toxic). The subset contains 34,283 non-toxic (85.7%) and 5,717 toxic (14.3%) examples [17]. We randomly split the 40,000 comments into 24,000 for training, 8,000 for validation, and 8,000 for testing. A light text-cleaning pipeline (lowercasing and special-character handling) was applied to create a consistent basis for feature extraction. For the classical models, text was vectorized with TF-IDF over the top 20,000 unigrams and bigrams, fit on the training split and applied to validation/test. For the Transformer model, inputs were tokenized with the official ALBERT tokenizer and padded/truncated to a fixed maximum length. Engineered numeric features (character/word counts, capital-letter ratio, readability scores, and VADER sentiment) were computed for all splits; their incorporation into each model class is detailed in Section 2.2.

 doi:10.20944/preprints202512.1277.v1

2.2. Experimental Design: Feature Sets

We ran two configurations to isolate the effect of engineered features while keeping splits and preprocessing constant.

- Baseline Feature Set. For the classical models (Logistic Regression, Random Forest, LightGBM), inputs were TF-IDF vectors over the top 20,000 unigrams and bigrams (vocabulary fit on the training split, then applied to validation/test). For the Transformer, ALBERT (albert-base-v2) was fine-tuned on tokenized text only (official tokenizer; fixed max sequence length).
- Enhanced Feature Set (Social & Contextual Metadata): We integrated a set of non-textual features derived from the platform's interaction data. These included user reactions, platform metrics, and context identifiers.

To handle these categorical variables, we applied a safe label encoding strategy fitted on the training set, with unseen values in the test set mapped to a generic placeholder to prevent data leakage. Following encoding, we standardized the features using the training split statistics. These standardized features were then integrated into the respective pipelines: for the classical machine learning models, they were concatenated directly to the sparse TF-IDF vectors, whereas for ALBERT, they were concatenated with the pooled [CLS] representation (768-d) before being passed to the final classification layer. No manual interaction terms were introduced; any interactions between the text and metadata were learned by the downstream models. All other settings, including tokenization and vectorization choices and train, validation, and test partitions, were held fixed across the two experiments.

2.3. Model Architectures

Our model selection spans both classical algorithms and a state-of-the-art deep learning architecture to provide a comprehensive performance spectrum.

2.3.1. Classical Supervised Learning Models

Our classical model evaluation began with Logistic Regression (LR) [20] as an interpretable linear baseline, chosen for its efficiency and robust performance in text classification. This model estimates the probability of a comment being toxic by applying a sigmoid function to a linear combination of its input TF-IDF features. We then explored two advanced non-linear ensembles to assess the upper-bound of performance on this feature set.

For the bagging approach, we implemented Random Forest (RF) [21], which mitigates overfitting by building a multitude of decision trees on bootstrapped samples of the training data. By considering only a random subset of features at each split, RF effectively reduces variance and captures complex feature interactions. For the boosting approach, we utilized LightGBM (LGBM), a high-performance framework that sequentially builds trees, where each new tree is trained to correct the residual errors of its predecessor. LGBM is highly efficient on large datasets due to its use of a leaf-wise tree growth strategy and histogram-based feature binning [22,23].

For all three models, a comprehensive grid search was performed to optimize key hyperparameters, using the Weighted F1-score on a validation set as the selection criterion. For Logistic Regression, this involved tuning the optimization *solver* (the algorithm to find optimal weights), the penalty type (e.g., L1 or L2) used to penalize large coefficients, and the regularization strength *C* (where smaller values specify stronger regularization). For the tree-based ensembles, tuning focused on controlling model complexity. This included the *n_estimators* (the total number of trees in the ensemble), the *max_depth* (the maximum depth of each tree), and the *min_samples_split* (the minimum data points required to split a node) and *min_samples_leaf* (the minimum data points required to form a leaf). For LightGBM specifically, the *learning_rate* was also tuned to scale the contribution of each sequentially added tree. Finally, the *class_weight* parameter was adjusted across all three models to give more importance to the minority 'True' (toxic) class, directly addressing the dataset's imbalance during training.

 doi:10.20944/preprints202512.1277.v1

2.3.2. Deep Learning Transformer Model

Representing the deep learning approach, we utilized ALBERT (A Lite BERT), a parameter-efficient Transformer [24]. Unlike classical models reliant on fixed features, ALBERT is designed to learn deep contextual representations. Its self-attention mechanism, in contrast to sequential RNNs, processes all tokens in parallel, allowing it to weigh the importance of all words in a sequence simultaneously. This architecture enables it to capture the complex, long-range semantic dependencies crucial for toxicity detection.

For the Baseline Feature Set, we fine-tuned the standard albert-base-v2 checkpoint on tokenized text only. For the Enhanced Feature Set, we utilized a specialized hybrid architecture, AlbertWithTabular, built upon the albert-base-v2checkpoint to fuse text with structured data. This model's architecture first processes the text through the ALBERT backbone to obtain the 768-dimension pooled [CLS] output vector. In parallel, it handles the 8 structured features specified in Section 2.2:

- The 2 numeric features (comment_count, toxicity_annotator_count) were standardized using a StandardScaler (fit on the training set) and passed to the model as a 2-dimension vector.
- The 6 categorical features (rating, wow, sad, funny, likes, disagree) were label-encoded and then passed through their own dedicated nn.Embedding layers to learn dense vector representations for each category.

All resulting vectors—the 768-d [CLS] vector, the learned categorical embeddings, and the 2-d standardized numeric vector—were then concatenated into a single "fusion vector". This vector was passed through a two-layer MLP classification head (with hidden layers of 256 and 128) using ReLU activations and Dropout before the final linear layer. This architecture allows the model to learn complex, non-linear interactions between the semantic meaning of the text and the structured metadata associated with it. A randomized hyperparameter search was conducted to find the optimal learning_rate, dropout_rate, number of fine-tuning epochs, and key architecture parameters like embedding dimensions and MLP hidden layer sizes. The final model configuration was selected based on the highest Weighted F1-score on the validation set, using a weighted cross-entropy loss function during training to manage class imbalance.

2.4. Evaluation Framework

A multi-faceted evaluation framework was established to ensure a robust and reliable model comparison. While standard metrics like accuracy, precision, and recall were computed, the pronounced class imbalance led us to designate the Weighted F1-score as our primary metric for model tuning and selection. This score provides a balanced measure of a model's performance on both the majority ('False') and the minority ('True') classes [18,19].

In addition, we calculated the Area Under the Receiver Operating Characteristic Curve (AUC). The AUC provides a threshold-independent measure of the model's aggregate ability to discriminate between the two classes, making it a particularly robust metric for imbalanced data [18,19]. To quantify the stability of these results, 95% Confidence Intervals (CIs) were calculated for both primary metrics via a 1,000-iteration bootstrap method.

3. Results

In this section, we present the empirical results of our two-phase experiment. The analysis is structured to clearly demonstrate the impact of our enhanced feature set on the performance of both classical machine learning and deep learning architectures. We will cover three main points:

- We first present the model performance from Experiment 1 (Baseline Features), establishing a baseline for all four models on the test set.
- We then present the results from Experiment 2 (Enhanced Features) and conduct a comparative analysis to quantify the performance shift.

 doi:10.20944/preprints202512.1277.v1

- Finally, we analyze the models' behavior, discussing how the results support our finding that explicit social signals allow efficient classical models to rival the performance of deep Transformers.

3.1. Model Performance: Experiment (Baseline Features)

The initial experiment used TF-IDF vectors for the classical models and raw tokenized text for ALBERT; results on the held-out test set are summarized in Table 1. ALBERT achieved an AUC of 0.9346, substantially higher than the classical models, whose AUCs clustered between 0.7550 and 0.7848. Crucially, the Macro Average F1-Score—which treats the minority 'toxic' class equally to the majority class—reveals the depth of the classical models' struggle. While their Weighted F1 scores appear respectable (~0.85) due to the heavy class imbalance, their Macro F1 scores hover in the 0.65–0.67 range. This indicates that in the absence of metadata, the classical models largely failed to identify the minority toxic class, relying instead on high performance on the majority non-toxic class.

In the baseline configuration, qualitative inspection revealed that classical models frequently misclassified toxic comments that lacked explicit profanity (e.g., sarcasm or veiled insults), as TF-IDF features struggle to capture semantic intent. ALBERT, leveraging self-attention, correctly identified many of these subtle instances. However, in the enhanced configuration, the error profiles converged. We observed that many "subtle" toxic comments were accompanied by distinct metadata signatures, such as a high count of 'disagree' reactions or a low moderator score. By accessing this metadata, the classical models were able to "correct" their predictions on these difficult samples without needing to understand the linguistic nuance, effectively using social signals as a proxy for semantic understanding.

Table 1. Model Performance on Test Set (Experiment 1: Baseline Features).

Models	AUC	Weighted Average		
		Precision	Recall	F1-Score
Logistic Regression	0.7784 (0.7711, 0,7852)	0.87	0.88	0.8555 (0.8515, 0.8597)
Random Forest	0.7550 (0.7475, 0.7623)	0.86	0.88	0.8509 (0.8464, 0.8550)
LightGBM	0.7848 (0.7780, 0.7918)	0.85	0.80	0.8177 (0.8143, 0.8211)
ALBERT	0.9346 (0.9313, 0.9378)	0.90	0.83	0.8475 8443, 0.8506)

3.2. Model Performance: Experiment 2 (Enhanced Features)

The introduction of the Enhanced Feature Set (User Reactions, Platform Ratings, Context IDs) produced a transformative shift in performance. As shown in Table 2, the classical models achieved a dramatic improvement, effectively closing the performance gap with the Transformer. LightGBM emerged as the top performer, achieving an AUC of 0.9798 and a Weighted F1 of 0.9560. Logistic Regression also saw massive gains, reaching an AUC of 0.9774. ALBERT improved as well (AUC 0.9788), but no longer held a statistically significant advantage over the classical ensembles. The 95% Confidence Intervals for LightGBM and ALBERT now overlap, indicating performance parity.

Table 2. Model Performance on Test Set (Experiment 2: Enhanced Features).

Models	AUC	Weighted Average		
		Precision	Recall	F1-Score
Logistic Regression	0.9774 (0.9756, 0.9790)	0.96	0.95	0.9507 (0.9487, 0.9527)
Random Forest	0.9757 (0.9737, 0.9775)	0.96	0.95	0.9535 (0.9515, 0.9554)
LightGBM	0.9798 (0.9782, 0.9815)	0.96	0.95	0.9560 (0.9541, 0.9578)

 doi:10.20944/preprints202512.1277.v1

ALBERT	0.9788 (0.9775, 0.9801)	0.95	0.94	0.9394 9373, 0.9417)

3.3. Comparative Analysis and Discussion

A direct comparison of the baseline and enhanced experiments reveals the central finding of this study: domain-specific feature engineering acts as a powerful equalizer between classical and deep learning architectures. In the baseline phase, the classical models, limited by sparse TF-IDF representations, were unable to effectively distinguish toxic content, trailing the Transformer by a wide margin. However, the integration of social and contextual metadata in the second phase precipitated a fundamental shift in performance dynamics.

The addition of these explicit signals resulted in massive, statistically significant gains for the classical ensembles. Logistic Regression saw its AUC surge from 0.7784 to 0.9774, while LightGBM improved from 0.7848 to 0.9798, effectively tripling its discriminatory power on the minority class as evidenced by the rise in Macro F1-scores from approximately 0.67 to 0.91. Crucially, the 95% confidence intervals for the enhanced LightGBM model now fully overlap with those of the ALBERT model, indicating that the classical ensemble has achieved statistical parity with the deep learning baseline.

This convergence suggests that for this specific domain, the "intelligence" required to detect toxicity was not solely locked within deep semantic structures, but was also explicitly encoded in the social metadata. In Experiment 1, the classical models struggled because they treated high-dimensional text as noisy and sparse. In Experiment 2, features such as user reactions (e.g., 'disagree') and moderator ratings provided strong, high-fidelity signals that were highly correlated with the target variable. Once provided with these explicit indicators, efficient algorithms like LightGBM were able to construct decision boundaries just as effective as the latent representations learned by ALBERT. This result challenges the prevailing assumption that computationally expensive deep learning is strictly necessary for high-performance moderation, demonstrating that well-engineered metadata can empower lightweight models to rival state-of-the-art Transformers.

A qualitative error analysis supports this conclusion. For instance, a comment like, "Bye! Don't look, come or think of comming back! Tosser." was misclassified by the Baseline (text-only) ALBERT. The term "Tosser" is a toxic insult, but the model likely failed to capture this specific, less common term. The Enhanced ALBERT model, however, successfully identified the comment as toxic. This suggests that the Enhanced model learned to use the 8 associated structured features as a strong contextual signal to correct the Baseline model's prediction, demonstrating its ability to fuse non-textual metadata to improve classification on ambiguous or difficult text samples.

4. Discussion and Conclusion

4.1. Interpretation of Findings

Our results reveal a remarkable convergence in model utility, challenging the prevailing view that deep learning is strictly necessary for high-performance toxicity detection. In the Baseline experiment, ALBERT established a dominant advantage (AUC 0.9346 vs. ~0.78), confirming that for raw, isolated text, the deep contextual representations of a Transformer are far superior to sparse TF-IDF vectors. However, the Enhanced experiment demonstrated that this advantage is largely erased when social and contextual metadata are introduced [25].

The dramatic improvement of the classical models, with LightGBM reaching an AUC of 0.9798, suggests that toxicity in this domain is not merely a linguistic property, but a social one. Features such as user reactions (e.g., 'disagree', 'likes') and platform ratings serve as high-fidelity, explicit signals that are strongly correlated with the target variable. While ALBERT infers toxicity through complex, non-linear semantic analysis, classical models achieve parity by simply leveraging these explicit social cues. In effect, the metadata provides a "shortcut" to the correct classification, rendering the architectural depth of the Transformer redundant for this specific task configuration.

 doi:10.20944/preprints202512.1277.v1

4.2. Practical Implications

These findings support a strategy of "Context-Aware Efficiency" with significant implications for scalability and "Green AI."

- Operational Efficiency: We demonstrate that a LightGBM model, which runs efficiently on standard CPUs, can match the accuracy of a GPU-dependent Transformer if the feature set includes social signals. For high-volume platforms processing millions of comments, deploying feature-augmented classical models offers a viable path to reducing inference costs and energy consumption without sacrificing accuracy.
- Hybrid Moderation Systems: The results suggest a tiered approach. For new comments (where no metadata exists), deep learning models like ALBERT are essential. However, for reactive moderation (cleaning up existing threads where user reactions have already occurred), lightweight classical models can replace heavy Transformers, allowing resources to be reallocated to more complex tasks.

4.3. Study Limitations and Future Directions

This study's design necessitates crucial acknowledgments regarding the nature of the features. First, many of our most predictive enhanced features (specifically user reactions) are lagging indicators—they accumulate only after a comment has been posted and viewed. Therefore, the specific high-performance classical models described here are best suited for post-hoc moderation rather than real-time pre-filtering. Future work should disentangle features available at the moment of creation from those that accumulate over time to determine the precise "break-even point" for classical models.

Second, while we standardized features for ALBERT, alternative fusion strategies (e.g., cross-attention layers) might yield different results compared to simple concatenation. Finally, we did not conduct a subgroup robustness or fairness analysis, which is critical given that social metadata (like user dislikes) can sometimes reflect community biases rather than objective toxicity.

4.4. Conclusion

In a controlled, two-phase comparison, we demonstrated that rich feature engineering acts as a great equalizer. While ALBERT dominated on raw text, the integration of social and contextual metadata propelled classical models, specifically LightGBM, to performance parity (AUC ~0.98). This refines the industry narrative: while Transformers are indispensable for understanding language in isolation, efficient classical algorithms remain state-of-the-art when deployed in a rich, context-aware environment. For practitioners, this offers a compelling argument for "Green AI": before scaling model size, one should first scale the richness of the input features.

References

1. Gillespie J. T., Custodians of the Internet: Platforms, Content Moderation, and the Hidden Decisions that Shape Social Media, Yale University Press, 2018.
2. Mansur Z., Omar N., and Tiun S., "Twitter Hate Speech Detection: A Systematic Review of Methods, Taxonomy Analysis, Challenges, and Opportunities," *IEEE Access*, vol. 11, pp. 16226–16249, 2023. doi: 10.1109/ACCESS.2023.3239375.
3. Malik J. S., Qiao H., Pang G., et al., "Deep Learning for Hate Speech Detection: A Comparative Study," *International Journal of Data Science and Analytics*, vol. 20, pp. 3053–3068, 2025. doi: 10.1007/s41060-024-00650-6.
4. Xu S., Tian Y., Cao Y., Wang Z., and Wei Z., "Benchmarking Machine Learning and Deep Learning Models for Fake News Detection Using News Headlines," *Preprints*, 2025061183, 2025. doi: 10.20944/preprints202506.1183.v1.

5. Tian Y., Xu S., Cao Y., Wang Z., Wei Z., "An Empirical Comparison of Machine Learning and Deep Learning Models for Automated Fake News Detection," *Mathematics*, vol. 13, no. 20, p. 2086, 2025. doi: 10.3390/math13132086.
6. Xu S., Ding Z., Wei Z., Yang C., Li Y., Chen X., Wang H., "A Comparative Analysis of Deep Learning and Machine Learning Approaches for Spam Identification on Telegram," *2025 6th International Conference on Computer Communication and Network Security*, 2025.
7. Liu Y., Shen X., Zhang Y., Wang Z., Tian Y., Dai J., and Cao Y., "A Systematic Review of Machine Learning Approaches for Detecting Deceptive Activities on Social Media: Methods, Challenges, and Biases," *International Journal of Data Science and Analytics*, vol. 20, pp. 6157–6182, 2025. doi: 10.1007/s41060-025-00850-8.
8. Cao Y., Dai J., Wang Z., Zhang Y., Shen X., Liu Y., and Tian Y., "Machine Learning Approaches for Depression Detection on Social Media: A Systematic Review of Biases and Methodological Challenges," *Journal of Behavioral Data Science*, vol. 5, no. 1, Feb. 2025. doi: 10.35566/jbds/caoyc.
9. Ding Z., Wang Z., Zhang Y., Cao Y., Liu Y., Shen X., Tian Y., and Dai J., "Trade-offs Between Machine Learning and Deep Learning for Mental Illness Detection on Social Media," *Scientific Reports*, vol. 15, article no. 14497, 2025. doi: 10.1038/s41598-025-99167-6.
10. Zhang Y., Wang Z., Ding Z., Tian Y., Dai J., Shen X., Liu Y., and Cao Y., "Employing Machine Learning and Deep Learning Models for Mental Illness Detection," *Computation*, vol. 13, no. 8, p. 186, 2025. doi: 10.3390/computation13080186.
11. Xu S., Wang H., Gao Y., Li Y., and Kuo M.-J., "More Than a Model: The Compounding Impact of Behavioral Ambiguity and Task Complexity on Hate Speech Detection," *Preprints*, 2025.
12. Wang H., Li Y., Gao Y., Kuo M.-J., and Xu S., "The Reliability Fallacy: How Label Ambiguity Undermines AI Hate Speech Detection," *Preprints*, 2025.
13. He C., Hu D., "Social Media Analytics for Disaster Response: Classification and Geospatial Visualization Framework," *Applied Sciences*, vol. 15, no. 8, p. 4330, 2025.
14. Al Shafian S., He C., Hu D., "DamageScope: An Integrated Pipeline for Building Damage Segmentation, Geospatial Mapping, and Interactive Web-Based Visualization," *Remote Sensing*, vol. 17, no. 13, p. 2267, 2025.
15. He C., Hu D., "Informing Disaster Recovery Through Predictive Relocation Modeling," *Computers*, vol. 14, no. 6, p. 240, 2025.
16. Iftikhar U., Ali S. F., Mustafa G., Bahar N., and Ishaq K., "Beyond Words: A Hybrid Transformer-Ensemble Approach for Detecting Hate Speech and Offensive Language on Social Media," *PeerJ Computer Science*, vol. 11, e3214, 2025. doi: 10.7717/peerj-cs.3214.
17. Jigsaw / Conversation AI, "Jigsaw Toxic Comment Classification Challenge," Kaggle, 2018. Retrieved from: https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge.
18. Powers D. M. W., "Evaluation: From Precision, Recall and F-measure to ROC, Informedness, Markedness and Correlation," *Journal of Machine Learning Technologies*, vol. 2, no. 1, pp. 37–63, 2011.
19. Davis J. and Goadrich M., "The Relationship Between Precision-Recall and ROC Curves," in *Proc. 23rd Int. Conf. Machine Learning (ICML)*, pp. 233–240, 2006.
20. Hosmer D. W. and Lemeshow S., *Applied Logistic Regression*, 2nd ed., New York, NY: John Wiley & Sons, Inc., 2000.
21. Breiman L., "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.
22. Friedman J. H., "Greedy Function Approximation: A Gradient Boosting Machine," *Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001.
23. Ke G., Meng Q., Finley T., Wang T., Chen W., Ma W., Ye Q., and Liu T.-Y., "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," in *Proc. 31st Int. Conf. Neural Information Processing Systems (NeurIPS)*, pp. 3149–3157, 2017.

 doi:10.20944/preprints202512.1277.v1

24. Lan Z., Chen M., Goodman S., Gimpel K., Sharma P., & Soricut R. "ALBERT: A Lite BERT for Self-supervised Learning of Language Representations." In *Proceedings of the 8th International Conference on Learning Representations (ICLR)*, 2020.
25. Zhou, Z., "Beyond Chat: A Framework for LLMs as Human-centered Support Systems," in *Cryptography and Information Security Trends 2025*, pp. 271–289, 2025.