ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

Evaluation Artefacts in Deep Hate Speech Detection: A Reproduction Study and a Character-Aware Model Robust to Adversarial Obfuscation

Hafsat Mahe Omar[1]; Abdulhakeem Ibrahim[2]; Karatu Musa Tanimu[2]; Sirajo Abdullahi Bakura[3]

[1,2,3]Department of Computer Science, Federal University Birnin Kebbi, 860222, Nigeria

Publication Date: 2026/09/15

Abstract: Automated hate speech detection is routinely reported at accuracies above 98%. We show that for two widely cited systems such figures are substantially artefacts of how their evaluation corpora were built, and we propose a model and protocol designed to avoid the same traps. First, we reproduce Ali et al. (2023) and Kibriya et al. (2024). Re-collecting the hashtag space of the former yields a corpus that is 91.3% duplicated; under the paper's stated protocol a lookup table that performs no learning scores 0.9804 accuracy, and deduplicating before splitting reduces the best model from 0.9915 to 0.5842. For the latter we rebuild the detection corpus from the paper's own recipe, matching its reported hate-class count exactly (22,862 tweets), and show that because the corpus merges two sources along the label axis, corpus identity nearly determines the label: the two sources are 98.8% separable from text alone, a provenance-only baseline reaches 0.8837 against a reported 0.933, and cross-corpus transfer falls below the target corpus's majority-class baseline in both directions. Second, we present a compact character-aware model a character CNN and corpus-trained WordPiece embedding feeding a BiLSTM with additive attention and a per-document graph convolutional branch evaluated over three seeds on a single human-annotated corpus with deduplication and an annotator-column guard. It attains 0.7354 macro-F1 (mean of three seeds) with 713k parameters, exceeding faithful re-implementations of both published architectures by 0.047 and 0.056 macro-F1 at a fraction of their size. Under five character-level evasion attacks it loses 0.252 macro-F1 against 0.359-0.379 for the neural baselines. We further report a quantitative evaluation of SHAP, LIME and attention against a random-deletion control, finding all three faithful but attention clearly the weakest. We report two negative results: the graph branch's contribution lies inside the seed noise, and a TF-IDF linear baseline matches the proposed model on clean data and degrades comparably under attack. All code, splits and artefacts are released.

Keywords: *Hate Speech Detection, Reproducibility, Dataset Bias, Adversarial Robustness, Explainable AI, Graph Convolutional Networks.*

How to Cite: Hafsat Mahe Omar; Abdulhakeem Ibrahim; Karatu Musa Tanimu; Sirajo Abdullahi Bakura (2026) Evaluation Artefacts in Deep Hate Speech Detection: A Reproduction Study and a Character-Aware Model Robust to Adversarial Obfuscation. *International Journal of Innovative Science and Research Technology*, 11(9), 208-215. https://doi.org/10.38124/ijisrt/26sep325

I. INTRODUCTION

Online hate speech causes measurable harm, and automated detection is now a standard component of platform moderation. The published literature reports the problem as close to solved: accuracies of 98% and above are common for multi-class hate categorisation [1,2], and such figures are used to justify deployment recommendations. Practitioner experience does not match those numbers, and the gap is rarely investigated.

This paper argues that a substantial part of the gap is an evaluation artefact rather than a modelling result, and that the artefacts are diagnosable with cheap, standard measurements the original papers omit. We make the argument constructively: we first reproduce two influential systems and identify precisely why their headline numbers cannot be read as detection ability, then design a model and protocol around the failures we found.

➢ *Our Contributions are:*

- A reproduction of Ali et al. [2] establishing that its reported accuracy is reachable by memorisation alone on a corpus re-collected from its own stated hashtag list, and that cross-validation cannot detect the responsible leakage.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

- A reproduction of Kibriya et al. [1] in which the detection corpus is rebuilt from the paper's own construction recipe — matching its reported hate-class count exactly — and shown to merge two sources along the label axis, so that a provenance-only baseline recovers most of the reported accuracy and cross-corpus transfer collapses.
- A compact character-aware BiLSTM-attention-GCN model evaluated under a corrected protocol (deduplication before splitting, annotator-column guard, three seeds, macro-F1 primary), which outperforms faithful re-implementations of both published architectures.
- A systematic quantitative evaluation of SHAP, LIME and attention weights against a random-deletion control, in place of the qualitative examples the literature typically offers.
- Two negative results reported in full: the graph branch's contribution is inside the seed noise, and a linear TF-IDF baseline is competitive with the proposed model on both clean and attacked data.

II. RELATED WORK

Davidson et al. [3] released the corpus that underpins much of this literature and made the central methodological point that hate speech and merely offensive language overlap heavily and are easily conflated. Subsequent deep-learning systems have reported steadily higher scores: BiLSTM and CNN hybrids [17], BERT-based transfer learning [18,19], ensemble approaches [20], and the two systems reproduced here [1,2].

A parallel line of work questions whether these gains are real. Arango et al. [26] found that several reported hate-speech results did not survive corrected evaluation. Wiegand et al. [14] showed that abusive-language datasets carry sampling biases that models exploit. Gorman and Bedrick [15] demonstrated that standard splits inflate apparent differences between systems, and Reimers and Gurevych [16] showed that single-run comparisons of neural architectures are frequently indistinguishable from seed noise. Our reproduction findings are consistent with this line and extend it with a specific mechanism that, to our knowledge, has not previously been quantified: label-axis corpus merging, in which the class label is nearly recoverable from the identity of the source corpus alone.

On explainability, SHAP [4] and LIME [5] are the standard post-hoc methods and are widely applied to hate-speech classifiers, generally through a handful of illustrative figures. Jain and Wallace [12] showed that attention weights are not reliable explanations, and DeYoung et al. [13] formalised comprehensiveness as a faithfulness measure. We adopt that measure here.

III. REPRODUCTION STUDY

Both systems were re-implemented from their published descriptions in PyTorch. Where a paper specifies Keras layers we used the direct equivalents, including the front-padding default of Keras' pad_sequences, which matters because every architecture involved classifies from the final timestep.

➢ *Ali et al. (2023): Duplicate Leakage*

The annotated six-class corpus of [2] is not obtainable; its distribution link returns HTTP 404. We therefore re-collected its stated hashtag space, obtaining a corpus whose twelve hashtags all appear in the paper's own collection list. That corpus is 91.3% duplicated after preprocessing.

All four architectures of the paper's Table 3 were evaluated under three protocols: the paper's stated pipeline (random split, no deduplication), the same pipeline with duplicates removed before splitting, and the Davidson corpus [3] with human labels and deduplication. Table 1 reports the outcome.

Table 1 Reproduction of Ali et al. [2] Under Three Protocols. P1 Follows the Paper's Stated Pipeline; P2 Differs Only in Removing Duplicate Texts before the Split; P3 Uses a Human-Labelled Corpus. Accuracy Unless Stated.

Model	P1: as stated	P2: deduplicated	P3: Davidson acc.	P3: Davidson macro-F1
LSTM-RNN	0.9897	0.5316	0.8579	0.6657
Stacked LSTM	0.9911	0.5632	0.8719	0.6541
LSTM-CNN	0.9915	0.5158	0.8771	0.6722
LSTM-GRU (proposed)	0.9884	0.4895	0.8776	0.6670
TF-IDF + LinearSVC (reference)	0.9906	0.5842	0.8920	0.7298
Lookup table (no learning)	**0.9804**	**0.3684**	—	—

Three observations follow. First, a lookup table that memorises training texts and otherwise predicts the majority class reaches 0.9804 under the paper's protocol, because 96.8% of test items appear verbatim in training. Any reported accuracy on such a corpus must clear that floor before it can be read as evidence of learning. Second, all four architectures fall within 0.0031 accuracy of one another, so the paper's central claim that one architecture is superior cannot be evaluated at all: leakage saturates every model. Third, the cross-validation the paper reports offers no protection: mean leakage within each validation fold is 96.9% (measured over three folds), because copies of a text are distributed across folds and each validation fold is therefore largely memorised from the others.

A second, independent leakage route exists in the Davidson release itself, which ships raw annotator vote tallies beside the label. We verify that the label is a deterministic function of those columns for 100.00% of rows: a decision tree on the vote columns alone, with no text whatsoever, attains 0.9999 accuracy, and text plus votes attains 1.0000. Text alone — the honest task — reaches 0.8970 accuracy and 0.7343 macro-F1.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

➢ *Kibriya et al. (2024): Label-Axis Corpus Merging*

The detection corpus of [1] is assembled by an explicit arithmetic recipe: two classes of the Davidson corpus are merged into a single hate class, that corpus is merged with a Kaggle sentiment corpus, and the normal class of the (unobtainable) third corpus is added. Two of the three inputs are available, so the recipe can be executed and checked against the paper's own reported totals.

Table 2 Reconstruction of the Detection Corpus of [1] from its Stated Recipe. The Hate Class Matches the Reported Count Exactly, which Validates the Reconstruction.

Component	Normal	Hate speech
DB1 hate + offensive (merged per the recipe)	—	20,620
DB1 neither	4,163	—
DB2 normal / racist-sexist	29,720	2,242
DB3 normal (unavailable; inferred)	4,979	—
Reconstructed total	**38,862**	**22,862**
Reported in [1]	**38,862**	**22,862**

The hate class reconstructs to the tweet (22,862), validating the reconstruction; 92.0% of the stated corpus is rebuildable without the missing third source. We note in passing that the paper's stated class counts sum to 61,724 against its stated total of 61,702, and that its description of the Davidson class distribution transposes the hate and offensive counts.

The consequential finding is structural. Roughly 90% of hate examples originate in one source corpus and 88% of normal examples in the other, so the label is nearly a function of provenance. We quantify this three ways (Table 3, Figure 1). A classifier trained to predict the source corpus from text alone attains 0.9883 accuracy. A model that predicts the source and answers with that source's majority label — performing no hate modelling of its own — attains 0.8837 against the paper's reported 0.933; our faithful re-implementation attains 0.9399, reproducing the paper.

That baseline is not by itself decisive, and we say so: because one source is 83% hate speech, the source classifier's strongest features are themselves slurs, so predicting provenance is partly hate detection by proxy. Two further measurements separate the hypotheses. Evaluated within the majority-normal source alone, the model exceeds that source's majority class by only 0.0147. And under cross-corpus transfer — train on one source, test on the other, the same task throughout — macro-F1 falls from 0.890 to 0.479 in one direction and from 0.794 to 0.182 in the other, in both cases to an accuracy below the target corpus's majority-class baseline. A model that had learned hate speech would not behave this way.

Table 3 Diagnosing the Label-Axis Merge in the Corpus of [1].

Measurement	Value
Source corpora separable from text alone	0.9883
Provenance-only baseline (no hate modelling)	**0.8837**
Accuracy reported by [1]	0.9330
Accuracy of our re-implementation	0.9399
Gain over majority class, within majority-normal source only	+0.0147
Cross-corpus transfer, macro-F1 (train DB1 → test DB2)	**0.890 → 0.479**
Cross-corpus transfer, macro-F1 (train DB2 → test DB1)	**0.794 → 0.182**

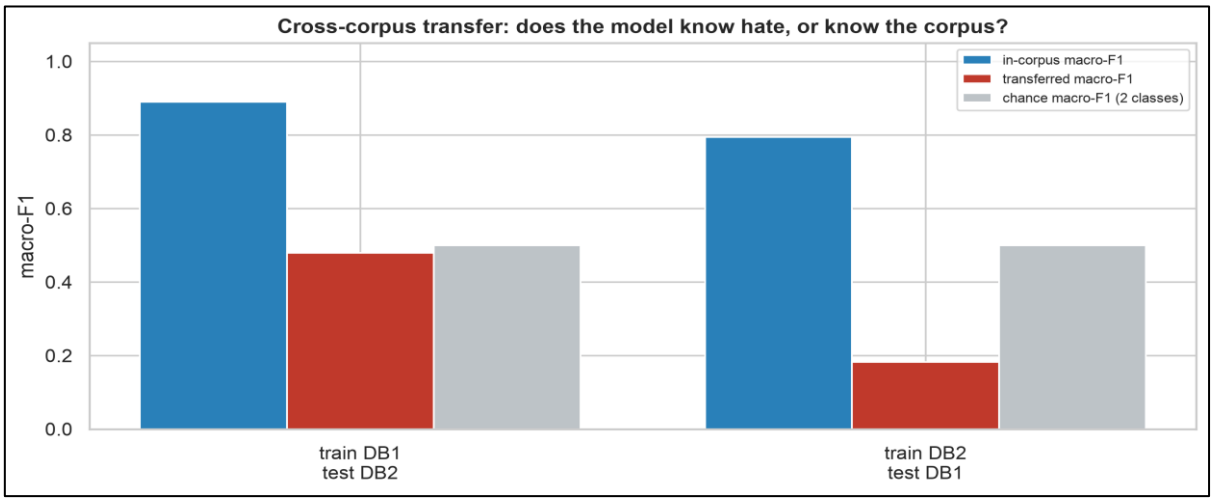

Fig 1 Cross-Corpus Transfer for the Detection Corpus of [1]. Training on One Source and Testing on the Other — the Same Task, the Same Language, Only the Collector Changes — Reduces Macro-F1 to at or below Chance for a Two-Class Problem.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

We also note that the two stages of the framework in [1] are evaluated only in isolation. Composing them under our re-implementation gives an end-to-end accuracy of 0.8381, and because the second stage has no 'not hate' class, the 3.3% of normal traffic that the first stage passes through is necessarily assigned a hate category. Finally, the architecture as specified in the paper's hyper-parameter table contains 3,112,997 parameters rather than the 2M claimed in its comparison table; the latter corresponds to a 200-dimensional embedding, and 96% of the count is the embedding table in either case.

IV. PROPOSED METHOD

The reproduction motivates three design commitments: represent text below the word level so that obfuscation degrades gracefully; train on a single human-annotated corpus rather than a merged one; and evaluate with deduplication, seed repetition and explicit floors.

➢ *Preprocessing and Representation*

Text is lowercased and stripped of URLs and user mentions. Three normalisations target evasion specifically: character substitutions are reversed (@→a, 1→i, 0→o, 3→e, $→s), elongations are collapsed, and contractions expanded. De-obfuscation necessarily precedes the removal of non-alphabetic characters, or the signal is destroyed rather than recovered. Lemmatisation, part-of-speech tags and dependency parses are produced in a single spaCy pass.

Each token receives two representations, concatenated to form its node feature. A character CNN (a convolution over the token's character embeddings, max-pooled) captures morphology and survives character-level corruption. A WordPiece [10] embedding, with the vocabulary trained on the training split only, decomposes rare or misspelt words into known units rather than mapping them to a single unknown token.

➢ *Architecture*

The combined node features feed two branches. A BiLSTM [7,8] models sequence, and additive attention [9] pools its hidden states: $u_t = \tanh(W h_t + b)$, $\alpha = \mathrm{softmax}(u^T v)$, and the context vector is the α-weighted sum of hidden states. In parallel, each document is treated as a graph whose nodes are tokens and whose edges come from dependency relations and a co-occurrence window of ±2; self-loops are added and the adjacency symmetrically normalised, and a two-layer GCN [6] operates on it. The attention vector and the mean-pooled GCN output are concatenated, batch-normalised, dropped out at 0.3, and passed to a softmax layer. Per-document graphs keep the adjacency dense and small and, unlike a corpus-level graph, cannot leak information across the train/test split.

V. EXPERIMENTAL SETUP

All experiments use the Davidson corpus [3], deduplicated after preprocessing to 23,721 tweets in three classes, split 70:15:15 stratified (16,604 / 3,558 / 3,559). Two guards are enforced. The annotator vote columns are a forbidden input, asserted programmatically with a test that the assertion fires. Deduplication occurs after preprocessing and before splitting, since cleaning converts near-duplicates into exact ones.

Training uses Adam [22] at 10^{-3}, batch size 64, up to 30 epochs with early stopping on validation macro-F1, inverse-frequency class weighting, dropout and batch normalisation. Model selection never touches the test split. Because architecture comparisons at this scale are easily confused with noise [16], every configuration is run over three seeds and reported as mean ± standard deviation. Macro-F1 is the primary metric, given the 13:1 class imbalance. The robustness and explanation studies of Sections 6.2 and 6.3 use the first-seed model throughout, so their comparisons are internally consistent though not seed-averaged.

Baselines are a TF-IDF linear SVM, faithful re-implementations of the architectures of [1] and [2] trained on identical splits, and an ablation ladder over the proposed model's components. We compare against re-implementations rather than published numbers because, as Section 3 establishes, figures measured on different corpora under different protocols are not commensurable.

VI. RESULTS

➢ *Classification and Ablation*

Table 4 Classification Results on the DE Duplicated Davidson Corpus, 70:15:15. Deterministic Baselines have No Seed Variance.

Model	Accuracy	Macro-F1 (mean ± sd, 3 seeds)	Parameters
TF-IDF + LinearSVC (reference)	0.8980	0.7356 —	60,000
Proposed: char/sub-word + BiLSTM + Attn + GCN	**0.8826**	**0.7354 ± 0.0081**	**712,819**
Ablation: BiLSTM + Attention	0.8672	0.7194 ± 0.0039	692,179
Ablation: BiLSTM only	0.8693	0.7008 ± 0.0045	679,763
Ablation: BiLSTM + GCN	0.8556	0.6985 ± 0.0076	700,403
Ablation: word embeddings (no char/sub-word)	0.8586	0.6977 ± 0.0168	1,836,547
Kibriya et al. [1] architecture, re-implemented	0.8626	0.6881 ± 0.0172	4,509,131
Ali et al. [2] architecture, re-implemented	0.8415	0.6793 ± 0.0109	1,767,775

Typical seed-to-seed standard deviation is 0.0099 macro-F1, so differences below roughly 0.02 should not be interpreted. Three claims survive that threshold. The proposed model exceeds the re-implemented architecture of [1] by +0.0474 and that of [2] by +0.0561 macro-F1, using 713k parameters against 4.51M and 1.77M. Attention contributes

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

+0.0186 over the plain BiLSTM. The character and sub-word representation contributes +0.0378 over the identical architecture with word embeddings substituted in.

Two results do not survive it, and we report them as negative findings. The graph branch is -0.0023 macro-F1 relative to the plain BiLSTM when used without attention, and adding it on top of attention moves the score by less than the seed noise. We attribute this to document length: tweets average about a dozen tokens, so a per-document graph offers little structure a BiLSTM has not already captured. And the TF-IDF linear baseline attains 0.7356 macro-F1 on 60k parameters, statistically level with the proposed model. On clean data alone, the deep architecture does not earn its cost.

Per class (first seed, the model also used in Sections 6.2 and 6.3), the minority hate-speech category remains hardest at F1 0.408 against 0.920 for offensive language and 0.854 for neither, on 203 test items. This is the overlap Davidson et al. [3] identified and is a property of the annotation rather than of the architecture. Majority-class and memorisation baselines both sit at 0.2912 macro-F1 — deduplication drives verbatim test-train overlap to zero — so the model clears its floors by 0.44.

➢ *Adversarial Robustness*

The test set was attacked with five evasion tactics — leetspeak substitution, character elongation, vowel deletion, adjacent-character swaps and intra-word space insertion — at increasing rates. Attacks are applied to the raw text and the full pipeline is re-run, so de-obfuscation gets its opportunity; no model is retrained, so this measures robustness rather than adaptation.

Table 5 Macro-F1 Under Character-Level Attack, Every Word Attacked. 'Mean Drop' Averages the Loss Across all Five Attacks. Lower Drop is Better.

Model	Clean	Leet	Vowel drop	Swap	Space	Mean drop
TF-IDF + LinearSVC (reference)	0.7356	0.7343	0.4386	0.4374	0.4200	0.2352
Proposed: char/sub-word + BiLSTM + Attn + GCN	**0.7271**	**0.7275**	**0.3473**	**0.3237**	**0.4560**	**0.2515**
Ablation: word embeddings (no char/sub-word)	0.7128	0.7153	0.2463	0.2441	0.2828	0.3590
Ali et al. [2] architecture, re-implemented	0.6736	0.6736	0.1927	0.1777	0.2358	0.3732
Kibriya et al. [1] architecture, re-implemented	0.7079	0.7130	0.2339	0.1829	0.2743	0.3794

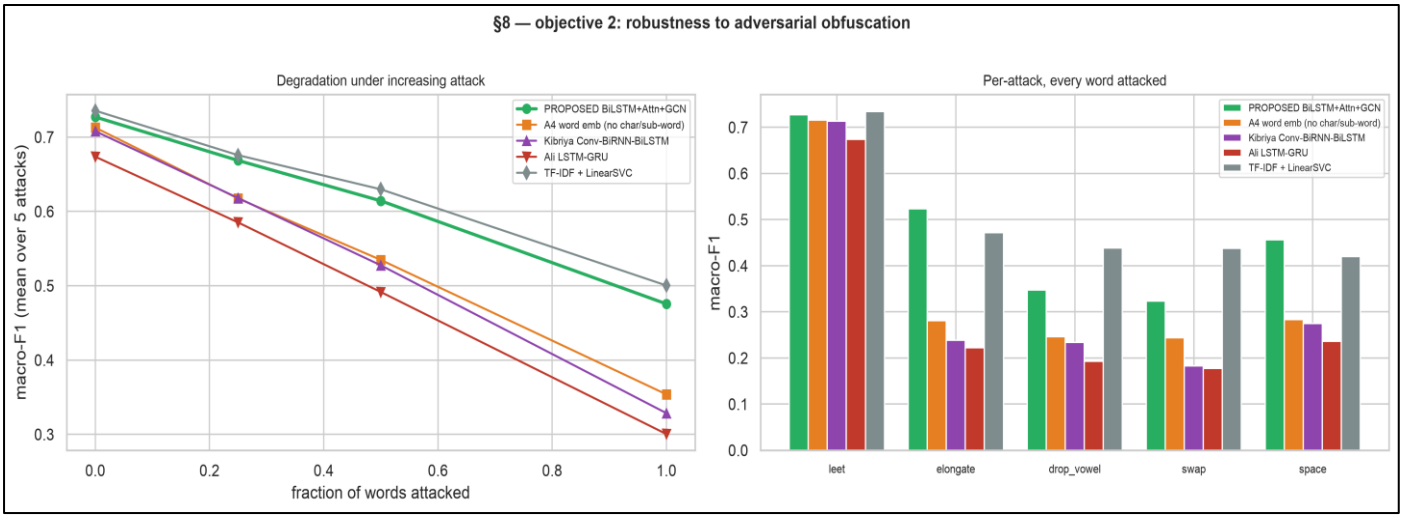

Fig 2 Degradation Under Increasing Attack Rate (Left) and Per-Attack Macro-F1 at Full Attack (Right).

Among neural models the proposed design is decisively the most robust, losing 0.252 macro-F1 against 0.359, 0.373 and 0.379 for the word-level variant and the two published architectures. Because the word-level variant is the identical architecture with the representation swapped, that margin is attributable to the character and sub-word views alone.

Two qualifications are necessary. Leetspeak barely affects any model sharing our preprocessing, because de-obfuscation reverses it before the model sees it; that is preprocessing rather than architecture, and the per-attack columns separate the two. And the TF-IDF baseline does not collapse, losing 0.235 from a higher starting point. The expected mechanism — character edits destroying exact n-grams — is offset by the several intact cues a tweet usually retains. The defensible claim is therefore that the proposed model is the most robust neural architecture tested and the only one that remains competitive with a linear baseline under attack.

➢ *Explanation Quality*

SHAP [4], LIME [5] and the model's own attention weights were evaluated on 24 correctly classified test instances against three measurable properties: mutual agreement, faithfulness by comprehensiveness [13] against a random-deletion control, and stability across repeated runs. Explanations are computed over raw strings, so the wrapper re-runs the complete pipeline; explaining a model over pre-tokenised input would attribute importance to tokens the deployed system never sees.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

Table 6 Faithfulness of Three Explanation Methods: Mean Drop in the Predicted Class Probability when the Three Highest-Ranked Tokens are Deleted, against Deleting Three Random Tokens.

Method	Comprehensiveness	Random-deletion control	Lift
SHAP	**0.7406**	**0.1497**	**+0.5908**
LIME	0.6920	0.1497	+0.5423
Attention	0.5131	0.1497	+0.3634

All three methods beat the random control comfortably, so all three are faithful in this sense. SHAP is strongest (lift +0.5908), LIME close behind (+0.5423), and attention clearly weakest (+0.3634). Top-5 agreement is 0.658 between SHAP and LIME but only 0.533 and 0.533 between attention and either. LIME's own stability across repeated runs on identical input is 0.867. These results support the caution of Jain and Wallace [12]: attention weights are available for free and are the least reliable of the three, yet are the ones most often presented as explanations without validation.

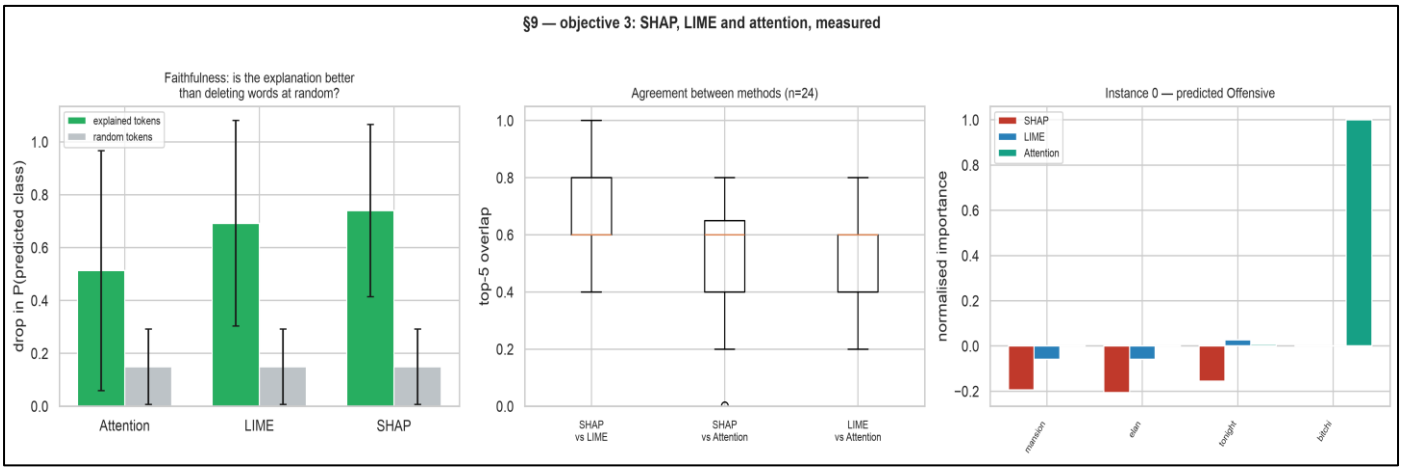

Fig 3 Explanation Faithfulness against a Random-Deletion Control (Left), Inter-Method Agreement (Centre), and a Worked Instance (Right).

VII. DISCUSSION AND THREATS TO VALIDITY

The reproduction and the model point at the same conclusion: in this literature, protocol decides more than architecture. Duplicate leakage and label-axis corpus merging each produce headline figures well above what the same models achieve under corrected evaluation, and neither is visible in the metrics these papers report. Both are cheap to detect. We recommend three measurements as routine: a memorisation floor, a provenance baseline with cross-corpus transfer whenever corpora are merged, and seed repetition with an explicit noise threshold.

Several threats bound our own claims. The corpus of [2] is unobtainable, so Section 3.1 establishes that a figure like the reported one is reachable by memorisation on a corpus re-collected from its stated hashtags — not that the specific published number is an artefact. Our own model is trained on a single corpus, so its generalisation beyond that sampling is untested; this is a deliberate trade against the merged-corpus confound documented in Section 3.2, and it means our absolute figures should not be compared with numbers obtained elsewhere. Three seeds bound the noise but do not support confidence intervals. The explanation study uses 24 instances, which is adequate for the large faithfulness lifts but not for fine distinctions in agreement. Finally, our attacks are synthetic and lexical; they do not model an adaptive adversary with access to the model.

We also record that a linear TF-IDF model matches our architecture on clean data and degrades comparably under attack. Two published deep architectures, faithfully re-implemented, lose to that baseline on the same corpus. Read together with the reproduction results, this suggests the field's reported progress is smaller than the published numbers imply, and that strong simple baselines belong in every comparison.

VIII. CONCLUSION

We reproduced two widely cited hate-speech detection systems and identified concrete mechanisms duplicate leakage and label-axis corpus merging by which their reported accuracies exceed what the models achieve on the underlying task. We then built a compact character-aware BiLSTM-attention-GCN model under a corrected protocol. It outperforms faithful re-implementations of both published architectures at a fraction of their parameter count, is markedly more robust to character-level evasion than any neural baseline tested, and comes with a quantitative rather than illustrative evaluation of its explanations. We report in full two results that do not favour our design: the graph branch's contribution lies within seed noise, and a linear baseline remains competitive throughout. [FUTURE WORK — e.g. multi-corpus evaluation with provenance controls, adaptive attacks, transformer comparison]

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

DECLARATIONS

- Data availability. The Davidson corpus [3] and the Kaggle sentiment corpus are publicly available; the corpus of [2] was not obtainable at the time of writing (HTTP 404). All code, splits, trained-model artefacts, figures and result tables supporting this paper are available at [REPOSITORY URL / DOI].
- Funding: This research received no specific grant from funding agencies in the public, commercial, or not-for-profit sectors.
- Declaration of competing interest: the authors declare that they have no competing financial interests or personal relationships that could have influence the work reported in this paper.
- Ethics. This study uses previously published, publicly available and de-identified social media corpora; no new data were collected from human participants. As the research involves only secondary analysis of pre-existing publicly available and anonymized datasets, formal institutional ethics committee approval was not required.
- CRediT author statement. Hafsat Mahe Omar: conceptualisation, methodology, software, investigation, formal analysis, writing – original draft. Abdulhakeem Ibrahim: supervision, methodology, writing – review and editing. Karatu Musa Tanimu: supervision, writing – review and editing.
- Content warning. Tables and figures derived from the corpora contain slurs and offensive language, reproduced only where necessary to report model behaviour.

REFERENCES

[1]. Kibriya H, Siddiqa A, Khan WZ, Khan MK. Towards safer online communities: Deep learning and explainable AI for hate speech detection and classification. Computers and Electrical Engineering. 2024;116:109153. doi:10.1016/j.compeleceng.2024.109153

[2]. Ali M, Hassan M, Kifayat K, Kim JY, Hakak S, Khan MK. Social media content classification and community detection using deep learning and graph analytics. Technological Forecasting and Social Change. 2023;188:122252.

[3]. Davidson T, Warmsley D, Macy M, Weber I. Automated hate speech detection and the problem of offensive language. In: Proceedings of the International AAAI Conference on Web and Social Media (ICWSM). 2017;11(1):512-515.

[4]. Lundberg SM, Lee S-I. A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems (NeurIPS). 2017:4768-4777.

[5]. Ribeiro MT, Singh S, Guestrin C. "Why should I trust you?": Explaining the predictions of any classifier. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2016:1135-1144.

[6]. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. In: International Conference on Learning Representations (ICLR). 2017.

[7]. Hochreiter S, Schmidhuber J. Long short-term memory. Neural Computation. 1997;9(8):1735-1780.

[8]. Schuster M, Paliwal KK. Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing. 1997;45(11):2673-2681.

[9]. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. In: International Conference on Learning Representations (ICLR). 2015.

[10]. Wu Y, Schuster M, Chen Z, et al. Google's neural machine translation system: Bridging the gap between human and machine translation. arXiv:1609.08144. 2016.

[11]. Devlin J, Chang M-W, Lee K, Toutanova K. BERT: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of NAACL-HLT. 2019:4171-4186.

[12]. Jain S, Wallace BC. Attention is not Explanation. In: Proceedings of NAACL-HLT. 2019:3543-3556.

[13]. DeYoung J, Jain S, Rajani NF, Lehman E, Xiong C, Socher R, Wallace BC. ERASER: A benchmark to evaluate rationalized NLP models. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL). 2020:4443-4458.

[14]. Wiegand M, Ruppenhofer J, Kleinbauer T. Detection of Abusive Language: the Problem of Biased Datasets. In: Proceedings of NAACL-HLT. 2019:602-608.

[15]. Gorman K, Bedrick S. We need to talk about standard splits. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL). 2019:2786-2791.

[16]. Reimers N, Gurevych I. Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging. In: Proceedings of EMNLP. 2017:338-348.

[17]. Fazil M, Khan S, Albahlal BM, Alotaibi RM, Siddiqui T, Shah MA. Attentional multi-channel convolution with bidirectional LSTM cell toward hate speech prediction. IEEE Access. 2023;11:16801-16811.

[18]. Mozafari M, Farahbakhsh R, Crespi N. A BERT-based transfer learning approach for hate speech detection in online social media. In: Complex Networks and Their Applications VIII. Springer; 2019:928-940.

[19]. Saleh H, Alhothali A, Moria K. Detection of hate speech using BERT and hate speech word embedding with deep model. Applied Artificial Intelligence. 2023;37(1):2166719.

[20]. Mazari AC, Boudoukhani N, Djeffal A. BERT-based ensemble learning for multi-aspect hate speech detection. Cluster Computing. 2024;27(1):325-339.

[21]. Newman MEJ, Girvan M. Finding and evaluating community structure in networks. Physical Review E. 2004;69(2):026113.

[22]. Kingma DP, Ba J. Adam: A method for stochastic optimization. In: International Conference on Learning Representations (ICLR). 2015.

[23]. Basile V, Bosco C, Fersini E, et al. SemEval-2019 Task 5: Multilingual detection of hate speech against immigrants and women in Twitter. In: Proceedings of the 13th International Workshop on Semantic Evaluation. 2019:54-63.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26sep325

[24]. Awal MR, Lee RK-W, Tanwar E, Garg T, Chakraborty T. Model-agnostic meta-learning for multilingual hate speech detection. IEEE Transactions on Computational Social Systems. 2023.
[25]. Zhou X, Yong Y, Fan X, et al. Hate speech detection based on sentiment knowledge sharing. In: Proceedings of ACL-IJCNLP. 2021:7158-7166.
[26]. Arango A, Pérez J, Poblete B. Hate speech detection is not as easy as you may think: A closer look at model validation. In: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval. 2019:45-54.