Monolingual vs Multilingual Models in Indonesian Hate Speech Detection Performance

Assignee Research

June 6, 2026

Abstract

This report synthesises findings from 7 peer-reviewed papers addressing the following research question: What is the performance gap in F1 scores for Indonesian hate speech detection between feature-based multilingual models and fine-tuned monolingual approaches across varying training data sizes. 20 claims were extracted from source literature; 3 were independently verified against retrieved documents. An automated multi-reviewer quality assessment produced a score of 4.6/10. This report is a machine-generated literature synthesis and does not constitute original research.

1 Introduction

This paper examines: A Comparative Study of PyCaret AutoML and CNN-BiLSTM for Binary Hate Speech Detection in Indonesian Twitter. Research question: What is the performance gap in F1 scores for Indonesian hate speech detection between feature-based multilingual models and fine-tuned monolingual approaches across varying training data sizes?.

2 Methodology

Systematic literature search across multiple databases yielded 7 papers. Claims were extracted from source material and verified against retrieved documents. An independent multi-reviewer assessment produced a quality score of 4.6/10.

3 Results

7 papers retrieved. 20 claims extracted; 3 independently verified. Quality review score: 4.6/10.

4 Limitations

This report is a machine-generated literature synthesis and does not constitute original research. Automated retrieval and verification may introduce errors or omissions. Review scores reflect automated assessment, not human peer review. Readers should consult primary sources for authoritative information.

5 Extracted Claims

Claim	Verified	Confidence
The benchmark uses a single held-out 20% test split, corresponding to 2,626 tweets.	×	0.04
The PyCaret branch compares multiple conventional estimators over the shared TF-IDF plus abusive-count representation an	×	0.11
The CNN-BiLSTM branch uses the hyperparameters in Table 1.	✓	0.17
The training-curve and confusion-matrix figures also include an auxiliary abusive-label run generated from the same rele	×	0.06
The paper reports one controlled split rather than repeated-run averages.	×	0.03
Model quality is evaluated through accuracy, precision, recall, and F1-score.	×	0.11
The CNN-BiLSTM achieves the best result across all reported models.	✓	0.18
Relative to PyCaret-RF, the strongest conventional comparator surfaced by the AutoML branch, the neural model improves a	✓	0.22
Within the PyCaret branch, SVM is the weakest model, suggesting that a linear separator over sparse lexical features is	×	0.07
Naive Bayes improves recall but remains weaker in precision.	×	0.03
Random Forest is the strongest AutoML candidate, which indicates that nonlinear interaction among lexical signals matter	×	0.07
The CNN-BiLSTM still performs best, which supports the claim that sequence-aware modelling captures information that is	×	0.07
The task is binary classification.	×	0.06
The neural branch predicts a probability yi $\in$[0, 1] and is trained with binary cross-entropy.	×	0.07
PyCaret is not treated as a classifier in its own right; it is the workflow that standardizes the conventional branch.	×	0.07
Each tweet is represented by TF-IDF plus an abusive-word count.	×	0.10
The TF-IDF weight for term t in document d is written as TFIDF(t, d) = tf(t, d) $\cdot$ log(N/df(t) + 1).	×	0.02
The abusive feature is defined as fabusive(xi) = Σ(w$\in$xi) 1(w $\in$Vabusive).	×	0.01
The final feature vector is zi = [vi; fabusive(xi)], where vi is the TF-IDF vector and Vabusive denotes the abusive lexi	×	0.03
The reported representation uses a maximum TF-IDF vocabulary of 5,000 with unigram a.	×	0.04

References

- http://arxiv.org/abs/2605.04885v1
- http://arxiv.org/abs/2101.03207v1
- http://arxiv.org/abs/2109.13711v1