SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

A Controlled Evaluation of Hybrid TF-IDF Representations and Classical Machine Learning for Abusive-Language Detection

Namrata Gaikwad [1], Sharada Ohatkar [2]

[1] MKSSS's Cummins College of Engineering for Women, Pune, India. E-mail: namrata.m.gaikwad@cumminscollege.in
[2] MKSSS's Cummins College of Engineering for Women, Pune, India. E-mail: sharada.ohatkar@cumminscollege.in
* Correspondence: namrata.m.gaikwad@cumminscollege.in;

Abstract
Abusive-language detection in short social-media text is challenging because harmful expressions are informal, context-dependent, and substantially less frequent than non-abusive content. This study presents a controlled empirical evaluation of preprocessing, TF-IDF representations, classical classifiers, and voting ensembles for binary abusive-language detection. A corpus of 20,114 labeled English social-media texts contains 2,533 abusive and 17,581 non-abusive instances. The experiments use stratified five-fold cross-validation and evaluate six preprocessing alternatives, eight feature configurations, six classifiers, class-weight sensitivity, numerical-feature scaling, and hard-voting ensembles. Lemmatization followed by hybrid word- and character-level TF-IDF gives the strongest sequentially selected representation. Random Forest achieves macro-F1 = 0.8893 ± 0.0091, accuracy = 0.9531 ± 0.0042, and abusive-class recall = 0.7706 ± 0.0129. Logistic Regression provides higher abusive recall (0.8160) and substantially lower prediction time. The best voting ensemble, combining SVM, RF, and LR, reaches macro-F1 = 0.8762 and therefore does not improve on the selected RF pipeline. Repeated five-fold evaluation gives RF macro-F1 = 0.8883 ± 0.0064. The results show that the hybrid word- and character-level representation gives the strongest tested feature configuration, while the tested handcrafted feature additions and voting ensembles do not improve macro-F1 over the selected Random Forest pipeline.

Keywords: abusive-language detection; TF-IDF; character n-grams; preprocessing ablation; classical machine learning; ensemble learning; macro-F1; stratified cross-validation

1. Introduction

Abusive language in online communication includes insults, threats, hate speech, harassment, and other target-directed harmful expressions. Automated detection has been studied since early hostile-message recognition and has subsequently expanded to Web 2.0 harassment and cyberbullying detection [1–3]. The task remains difficult because abusive expressions are informal, context-dependent, and sensitive to linguistic and cultural variation [4–9].

Classical machine-learning approaches remain relevant when reproducibility, computational cost, and relatively lightweight deployment are important. Word n-grams capture lexical and short-range sequential patterns, whereas character n-grams can capture subword variation, misspellings, obfuscation, and informal forms. Prior work has therefore explored word and character representations with classifiers including Naive Bayes, Support Vector Machines (SVM), and Random Forest (RF) [4].

A limitation of many comparative studies is that several experimental factors change simultaneously. Preprocessing, feature construction, class weighting, model selection, and validation strategy can each affect the reported result. Consequently, an observed gain cannot always be attributed to a particular design choice. The present study addresses this issue through a controlled sequence of ablation and comparison experiments. The study asks five related questions: (i) which text preprocessing strategy provides the strongest performance; (ii) whether word and character TF-IDF are complementary; (iii) whether handcrafted linguistic, punctuation/emoji, lexicon, sentiment, and part-of-speech features improve the hybrid representation; (iv) which classical classifier provides the best balance between abusive-class detection and overall performance; and (v) whether hard-voting ensembles provide an advantage over the strongest individual classifier.

The contribution is therefore an empirical analysis of representation and model choices rather than a claim of state-of-the-art performance. The experiments use a common dataset, a common stratified five-fold protocol,

explicit class-imbalance analysis, fold-level variability, out-of-fold error analysis, repeated cross-validation, and a separate ensemble experiment.

The main contributions are:

- A controlled preprocessing ablation covering six normalization strategies.
- An eight-configuration feature study comparing word TF-IDF, character TF-IDF, hybrid TF-IDF, and progressively enriched feature representations.
- A comparative evaluation of six classical classifiers under a common evaluation protocol.
- A hard-voting ensemble comparison using five combinations of base classifiers.
- Additional sensitivity analyses covering class weighting, numerical-feature scaling, repeated five-fold evaluation, pooled out-of-fold errors, and a paired RF-versus-LR statistical comparison.
- A reproducible implementation in Python/scikit-learn with fold-wise fitting of learned transformations.

2. Materials and Methods

2.1. Dataset

The experiments use `Updated_dataset.csv`, containing 20,114 labeled short English social-media text records. The dataset contains 17,581 non-abusive records (87.41%) and 2,533 abusive records (12.59%). The available schema contains a record identifier, text, binary label, label name, word count, and character count. The repository audit reports 20,114 unique texts and no author, thread, timestamp, or language metadata. Accordingly, the principal evaluation is message-level stratified cross-validation; author- or thread-grouped evaluation cannot be performed with the available metadata. The composition of the analyzed dataset is summarized in Table 1.

Table 1. Composition of the analyzed dataset.

Class	Count	Percentage
Abusive	2,533	12.59%
Non-abusive	17,581	87.41%
Total	20,114	100.00%

The strong class imbalance makes accuracy alone inadequate as the primary performance measure. A trivial majority-class predictor obtains accuracy = 0.8741 and macro-F1 = 0.4664 because it completely fails to identify the abusive class. This baseline is therefore reported alongside the learned models.

2.2. Experimental Design and Leakage Control

The enhanced experiment is organized as sequential stages. Stage 0 reproduces the original hybrid word/character RF configuration without additional preprocessing. Stage 1 compares six preprocessing strategies and selects the highest-performing option by mean macro-F1. Stage 2 evaluates eight feature configurations under the selected preprocessing. Stage 3 tests whether scaling the numerical handcrafted-feature block changes RF and KNN behavior. Stage 4 compares six classifiers using the selected preprocessing and feature configuration. Separate diagnostics evaluate class weighting, out-of-fold errors, repeated cross-validation, probability bins, and RF-versus-LR paired predictions.

All principal experiments use StratifiedKFold with five folds, shuffling, and random_state = 42. Preprocessors, TF-IDF vectorizers, numerical scalers, and classifiers are fitted using training data within each fold and then applied to the held-out fold. This fold-wise implementation prevents the vectorizer and learned numerical transformations from being fitted on held-out observations. The preprocessing and feature configuration are selected sequentially using the same cross-validation framework used for performance reporting; therefore, the final 0.8893 result is an internal model-selection estimate rather than a fully nested estimate of post-selection generalization.

2.3. Text Preprocessing Ablation

Six preprocessing conditions are evaluated: no preprocessing; English stopword removal; Porter stemming; WordNet lemmatization; stopword removal plus stemming; and stopword removal plus lemmatization. The transformation is deterministic and implemented inside each fold. For lemmatization, WordNetLemmatizer is used; for stemming, PorterStemmer is used. The highest mean macro-F1 is used to select the preprocessing strategy for subsequent experiments. The macro-F1 performance of the six preprocessing configurations is shown in Figure 2.

2.4. TF-IDF and Feature Construction

Word-level TF-IDF uses lowercase=True, strip_accents='unicode', sublinear_tf=True, max_features=10,000, min_df=2, max_df=0.95, and word n-grams of order 1–2. Character-level TF-IDF uses the same normalization settings, max_features=15,000, min_df=2, max_df=0.95, character n-grams of order 3–5. Concatenating the two blocks produces a hybrid representation with up to 25,000 sparse TF-IDF dimensions before numerical features are appended.

The feature study contains eight configurations: a word-TF-IDF baseline and E1–E7. E1 uses character TF-IDF; E2 combines word and character TF-IDF; E3 adds six lexical statistics (word count, character count, average word length, uppercase ratio, digit ratio, and alphabetic ratio); E4 adds elongated-word count, repeated-punctuation count, emoji count, and punctuation count; E5 adds an exploratory count of terms from a small internally defined abuse-term list; E6 adds four VADER sentiment scores (negative, neutral, positive, compound); and E7 adds seven POS ratios (noun, verb, adjective, adverb, pronoun, preposition, determiner). The E5 term list is treated as an exploratory feature rather than a validated external lexicon.

2.5. Classifiers and Configurations

Six classifiers are compared: linear SVM, KNN, RF, Logistic Regression (LR), Multinomial Naïve Bayes (NB), and Gradient Boosting (GB). The primary RF, SVM, and LR models use balanced class weights. KNN uses k = 5 and parallel neighbor search. GB operates after a 200-component TruncatedSVD reduction of the sparse feature matrix. The six classifiers and their principal implementation parameters are summarized in Table 2.

Table 2. Classifier implementations and principal hyperparameters.

Classifier	Implementation	Key parameters
SVM	SVC	linear kernel; C=1.0; class_weight=balanced
KNN	KNeighborsClassifier	n_neighbors=5; n_jobs=-1
RF	RandomForestClassifier	200 trees; max_depth=None; min_samples_leaf=1; max_features=sqrt; balanced; random_state=42
LR	LogisticRegression	L2; C=1.0; max_iter=2000; balanced; random_state=42
NB	MultinomialNB	alpha=1.0
GB	SVD + GradientBoosting	SVD=200 components; 100 estimators; learning_rate=0.1; max_depth=3

2.6. Ensemble Experiment

A separate experiment evaluates hard-voting ensembles using the original word-level TF-IDF representation and the same stratified five-fold protocol. Five combinations are evaluated: A = SVM + KNN + RF; B = SVM + RF + LR; C = SVM + KNN + LR; D = KNN + RF + LR; and E = SVM + KNN + RF + LR. Each base learner is independently fitted within the training fold and the final class is determined by hard majority voting. Because this ensemble experiment uses the original word-level TF-IDF pipeline rather than the final lemmatized hybrid pipeline, it is interpreted as a complementary baseline rather than as a direct ablation of the final model.

2.7. Sensitivity and Diagnostic Analyses

Class-weight sensitivity compares balanced and unbalanced RF, LR, and SVM. Scaling sensitivity compares numerical-feature scaling and no scaling on the full E7 representation for RF and KNN. For the final RF, pooled out-of-fold predictions are used to construct a confusion matrix and separate false-positive and false-negative files. Repeated StratifiedKFold with three repeats provides a 15-fold stability check. A paired RF-versus-LR analysis uses identical out-of-fold observations; the reported macro-F1 difference is accompanied by a bootstrap confidence interval, while a two-sided paired permutation test is used for accuracy.

2.8. Evaluation Metrics

Accuracy, precision, abusive-class recall, specificity, and macro-F1 are reported. Macro-F1 is the primary metric because it gives equal weight to the abusive and non-abusive classes under the 12.59% abusive prevalence. Training, feature-fit/transform, and prediction times are also reported where available. Runtime values are fold-level measurements on the repository execution environment and are interpreted comparatively rather than as hardware-independent benchmarks.

The overall experimental workflow, including preprocessing ablation, feature representation experiments, classifier comparison, sensitivity analyses, ensemble evaluation, and out-of-fold error analysis, is summarized in Figure 1.

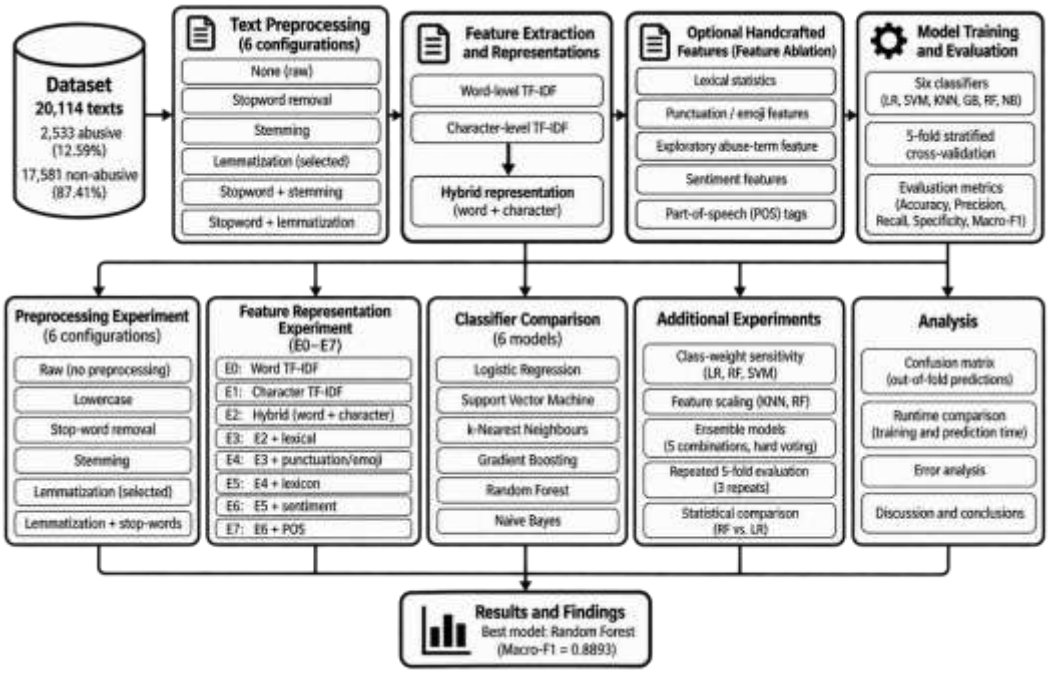

Figure 1. Overview of the controlled experimental workflow

3. Results

The experiments produce a consistent but nuanced result. Lemmatization provides the strongest preprocessing condition, hybrid word/character TF-IDF provides the strongest compact representation, and RF provides the highest macro-F1 among the six individual classifiers. However, the best hard-voting ensemble does not surpass RF. At the same time, LR detects a larger proportion of abusive instances and is substantially faster at prediction, demonstrating that the optimal model depends on the operational objective.

3.1. Majority Baseline and Overall Progression

The majority-class baseline and the sequential change from the no-preprocessing hybrid RF configuration to the selected lemmatized configuration are reported in Table 3.

Table 3. Baseline and sequential improvement of the selected RF pipeline.

System	Accuracy	Precision	Specificity	Abusive Recall	Macro-F1
Majority-class predictor	0.8741	0.0000	1.0000	0.0000	0.4664
Stage 0: RF + hybrid, no preprocessing	0.9516 ± 0.0042	0.8432 ± 0.0263	0.9796 ± 0.0039	0.7572 ± 0.0097	0.8852 ± 0.0089
Stage 1 selected: RF + hybrid + lemmatization	0.9531 ± 0.0042	0.8437 ± 0.0249	0.9794 ± 0.0038	0.7706 ± 0.0129	0.8893 ± 0.0091

3.2. Preprocessing Ablation

The complete performance results for the six preprocessing configurations are reported in Table 4.

Table 4. Preprocessing ablation using RF and hybrid word/character TF-IDF. Values are mean ± SD across five folds.

Condition	Macro-F1	Accuracy	Precision	Abusive Recall	Specificity
P0: None	0.8852 ± 0.0089	0.9516 ± 0.0042	0.8432 ± 0.0263	0.7572 ± 0.0097	0.9796 ± 0.0039
P1: Stopword removal	0.8858 ± 0.0121	0.9514 ± 0.0057	0.8340 ± 0.0331	0.7675 ± 0.0134	0.9779 ± 0.0051

P2: Stemming	0.8887 ± 0.0074	0.9527 ± 0.0036	0.8399 ± 0.0236	0.7718 ± 0.0072	0.9787 ± 0.0037
P3: Lemmatization	0.8893 ± 0.0091	0.9531 ± 0.0042	0.8437 ± 0.0249	0.7706 ± 0.0129	0.9794 ± 0.0038
P4: Stopword + stemming	0.8875 ± 0.0118	0.9518 ± 0.0056	0.8313 ± 0.0327	0.7758 ± 0.0133	0.9772 ± 0.0051
P5: Stopword + lemmatization	0.8872 ± 0.0094	0.9518 ± 0.0046	0.8320 ± 0.0288	0.7742 ± 0.0114	0.9774 ± 0.0047

Lemmatization achieves the highest macro-F1 (0.8893 ± 0.0091), narrowly exceeding stemming (0.8887 ± 0.0074) and the combined stopword/stemming condition (0.8875 ± 0.0118). Stopword removal alone improves abusive recall relative to no preprocessing but does not produce the highest balanced score. Lemmatization is therefore selected for the subsequent stages.

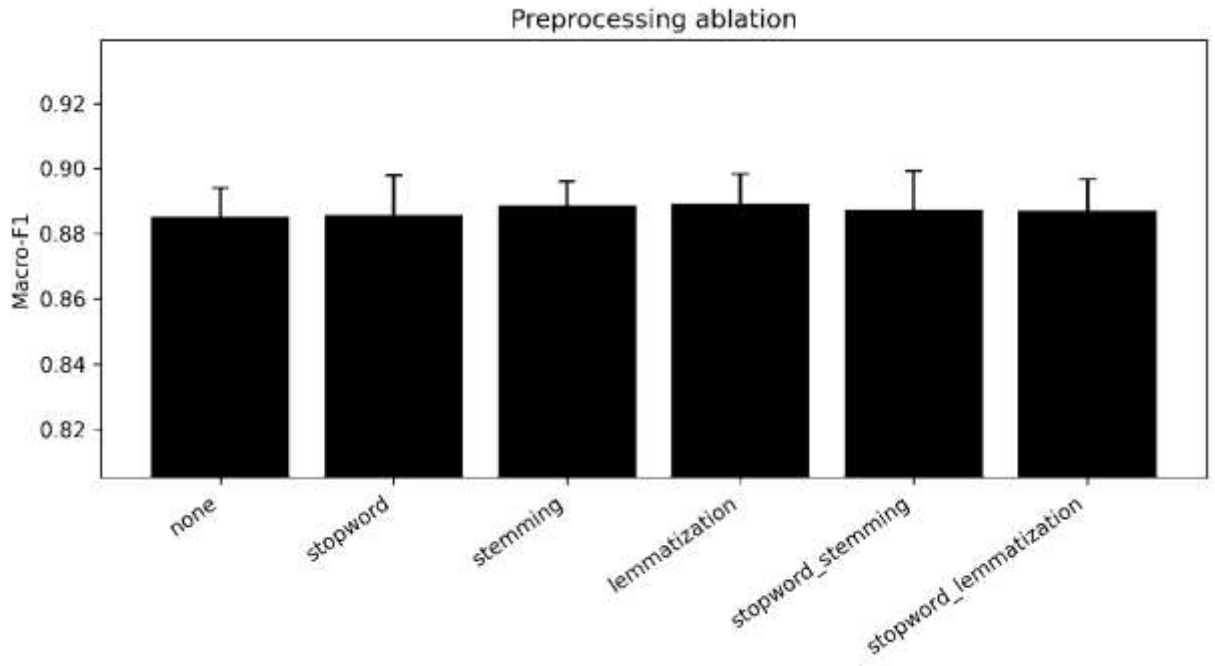

Figure 2. Macro-F1 across the six preprocessing configurations evaluated using Random Forest and hybrid word/character TF-IDF. Error bars denote one standard deviation across five folds

3.3. Feature Ablation

Table 5. Feature-ablation results under lemmatization. Values are mean ± SD across five folds.

Configuration	Macro-F1	Accuracy	Precision	Abusive Recall	Specificity
E0: Word TF-IDF	0.8665 ± 0.0061	0.9474 ± 0.0021	0.8821 ± 0.0135	0.6719 ± 0.0173	0.9870 ± 0.0018
E1: Character TF-IDF	0.8855 ± 0.0074	0.9514 ± 0.0036	0.8352 ± 0.0234	0.7655 ± 0.0074	0.9782 ± 0.0037
E2: Hybrid word + character	0.8893 ± 0.0091	0.9531 ± 0.0042	0.8437 ± 0.0249	0.7706 ± 0.0129	0.9794 ± 0.0038
E3: Hybrid + lexical	0.8881 ± 0.0079	0.9526 ± 0.0036	0.8417 ± 0.0216	0.7683 ± 0.0121	0.9791 ± 0.0033
E4: Hybrid + lexical + punctuation/emoji	0.8874 ± 0.0060	0.9523 ± 0.0030	0.8409 ± 0.0225	0.7667 ± 0.0114	0.9790 ± 0.0037
E5: Hybrid + lexical + punctuation/emoji + lexicon	0.8881 ± 0.0081	0.9526 ± 0.0038	0.8427 ± 0.0238	0.7675 ± 0.0100	0.9793 ± 0.0037

E6: Hybrid + sentiment	0.8859 ± 0.0078		0.9521 ± 0.0036		0.8474 ± 0.0238		0.7560 ± 0.0114		0.9803 ± 0.0035		
E7: Full configuration + POS	0.8861 ± 0.0067		0.9521 ± 0.0030		0.8464 ± 0.0197		0.7576 ± 0.0111		0.9801 ± 0.0030		

The eight feature configurations evaluated under lemmatization are compared in Table 5. The corresponding macro-F1 comparison is shown in Figure 3, with macro-F1 = 0.8893 ± 0.0091. Character TF-IDF alone reaches 0.8855 ± 0.0074, while word TF-IDF alone reaches 0.8665 ± 0.0061. Adding lexical statistics lowers macro-F1 to 0.8881; the subsequent additions of punctuation/emoji, the exploratory abuse-term indicator, sentiment, and POS features also remain below the E2 result. The full E7 configuration reaches 0.8861 ± 0.0067. Thus, none of the tested feature additions improves macro-F1 over the hybrid representation.

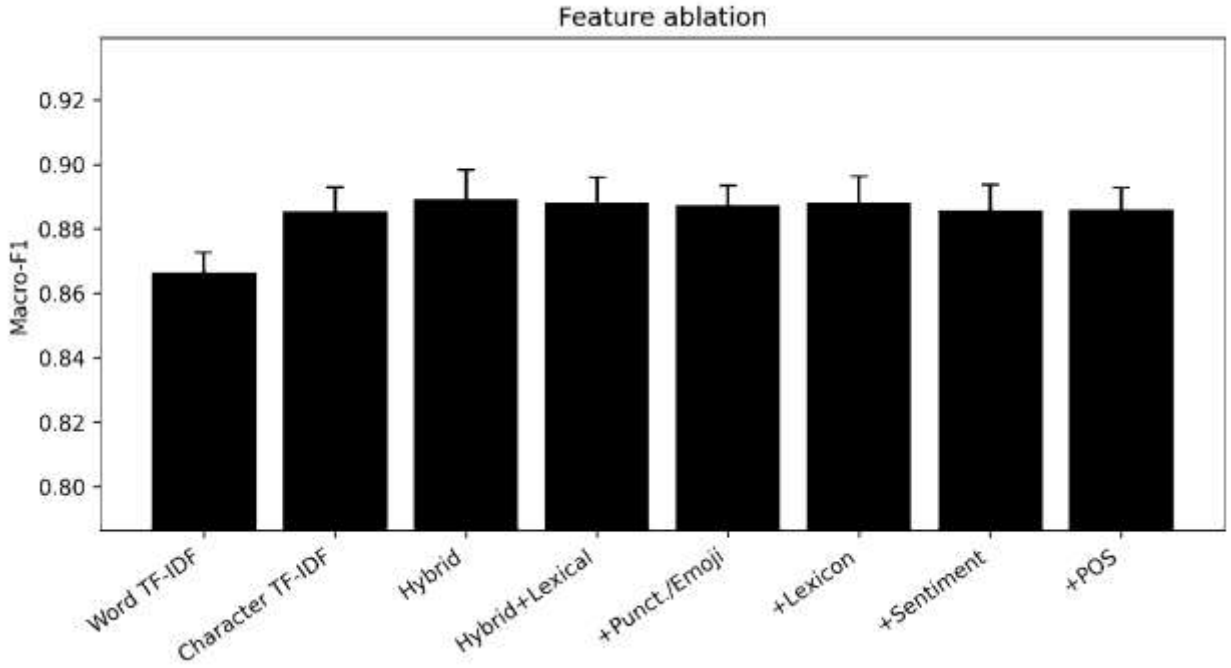

Figure 3. Macro-F1 across the eight feature configurations. Error bars denote one standard deviation across five folds.

3.4. Classifier Comparison

Table 6. Final classifier comparison on lemmatized hybrid word/character TF-IDF. Values are mean ± SD where applicable.

Model	Macro-F1	Accuracy	Precision	Abusive Recall	Specificity	Train s/fold	Predict s/fold
RF	0.8893 ± 0.0091	0.9531 ± 0.0042	0.8437 ± 0.0249	0.7706 ± 0.0129	0.9794 ± 0.0038	1.789	0.0341
LR	0.8782 ± 0.0066	0.9448 ± 0.0029	0.7623 ± 0.0125	0.8160 ± 0.0157	0.9633 ± 0.0024	0.188	0.0007
SVM	0.8697 ± 0.0068	0.9421 ± 0.0030	0.7644 ± 0.0135	0.7809 ± 0.0149	0.9653 ± 0.0024	52.661	11.1949
GB	0.8409 ± 0.0074	0.9376 ± 0.0032	0.8370 ± 0.0227	0.6269 ± 0.0091	0.9824 ± 0.0028	73.131	0.0380
NB	0.8216 ± 0.0031	0.9315 ± 0.0013	0.8218 ± 0.0101	0.5823 ± 0.0050	0.9818 ± 0.0012	0.004	0.0018
KNN	0.4826 ± 0.0066	0.8748 ± 0.0016	0.6402 ± 0.2756	0.0166 ± 0.0065	0.9984 ± 0.0014	0.001	0.9560

As shown in Table 6 and Figure 4, RF achieves the highest macro-F1 (0.8893 ± 0.0091) and accuracy (0.9531 ± 0.0042). LR has the highest abusive-class recall (0.8160 ± 0.0157) but lower macro-F1 (0.8782 ± 0.0066). SVM

achieves macro-F1 = 0.8697 ± 0.0068, while GB and NB are lower. KNN is particularly weak for the minority class: abusive recall is only 0.0166 ± 0.0065 despite specificity of 0.9984 ± 0.0014. RF prediction requires approximately 0.034 s per fold in this experiment, compared with approximately 0.0007 s for LR; SVM is much more expensive in both training and prediction. The complete performance and runtime results for the six classifiers are reported in Table 6.

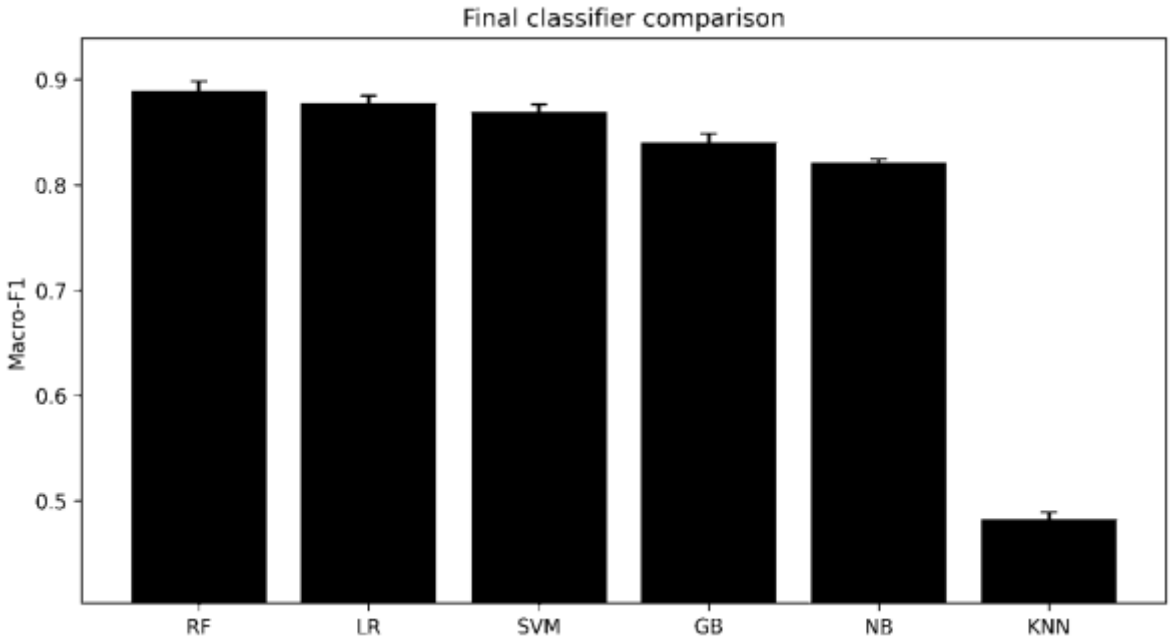

Figure 4. Macro-F1 comparison of the six classical classifiers on the selected pipeline.

3.5. Class-Weight Sensitivity

Table 7. Class-weight sensitivity for RF, LR, and SVM.

Model	Class Weight	Macro-F1	Abusive Recall	Precision	Specificity
LR	balanced	0.8782 ± 0.0066	0.8160 ± 0.0157	0.7623 ± 0.0125	0.9633 ± 0.0024
LR	unbalanced	0.8638 ± 0.0039	0.6593 ± 0.0112	0.8904 ± 0.0133	0.9883 ± 0.0017
RF	balanced	0.8893 ± 0.0091	0.7706 ± 0.0129	0.8437 ± 0.0249	0.9794 ± 0.0038
RF	unbalanced	0.8889 ± 0.0078	0.7738 ± 0.0114	0.8384 ± 0.0226	0.9784 ± 0.0036
SVM	balanced	0.8697 ± 0.0068	0.7809 ± 0.0149	0.7644 ± 0.0135	0.9653 ± 0.0024
SVM	unbalanced	0.8831 ± 0.0074	0.7378 ± 0.0165	0.8586 ± 0.0145	0.9825 ± 0.0020

Class weighting produces different changes across the tested models. For LR, balancing raises abusive recall from 0.6593 to 0.8160 and macro-F1 from 0.8638 to 0.8782. For SVM, balancing raises recall but lowers macro-F1. For RF, macro-F1 changes only slightly, from 0.8889 to 0.8893. The effects of balanced and unbalanced class weighting for RF, LR, and SVM are reported in Table 7.

3.6. Numerical-Feature Scaling Sensitivity

Table 8. Sensitivity to scaling of numerical features in the full E7 configuration.

Model	Scaling	Macro-F1	Abusive Recall	Precision	Specificity
KNN	S0_No_Scaling	0.5414 ± 0.0102	0.0940 ± 0.0121	0.4191 ± 0.0481	0.9812 ± 0.0021
KNN	S1_Scaled_Numerical	0.5990 ± 0.0177	0.1848 ± 0.0279	0.4872 ± 0.0318	0.9722 ± 0.0015
RF	S0_No_Scaling	0.8861 ± 0.0067	0.7576 ± 0.0111	0.8464 ± 0.0197	0.9801 ± 0.0030
RF	S1_Scaled_Numerical	0.8866 ± 0.0089	0.7580 ± 0.0122	0.8478 ± 0.0251	0.9803 ± 0.0037

Scaling changes the tested models differently: KNN macro-F1 rises from 0.5414 to 0.5990, whereas RF changes only from 0.8861 to 0.8866. The results of the numerical-feature scaling comparison for KNN and RF are reported in Table 8.

3.7. Ensemble Comparison

The results of the five hard-voting ensemble configurations are reported in Table 9.

Table 9. Hard-voting ensemble results using the original word-level TF-IDF pipeline.

Ensemble	Members	Macro-F1	Accuracy	Precision	Abusive Recall	Specificity
A	SVM + KNN + RF	0.8561	0.9446	0.8928	0.6368	0.9890
B	SVM + RF + LR	0.8762	0.9450	0.7752	0.7931	0.9668
C	SVM + KNN + LR	0.8737	0.9442	0.7766	0.7821	0.9676
D	KNN + RF + LR	0.8588	0.9452	0.8863	0.6478	0.9880
E	SVM + KNN + RF + LR	0.8563	0.9447	0.8942	0.6364	0.9891

Ensemble B (SVM + RF + LR) is the strongest of the five hard-voting combinations, as shown in Table 9 and Figure 5, with macro-F1 = 0.8762 and abusive recall = 0.7931. It does not exceed the final RF macro-F1 of 0.8893. Its abusive recall is higher than RF, while its macro-F1 is lower. The ensemble experiment used the original word-level TF-IDF pipeline, so it provides a complementary comparison rather than a direct extension of the final lemmatized hybrid pipeline.

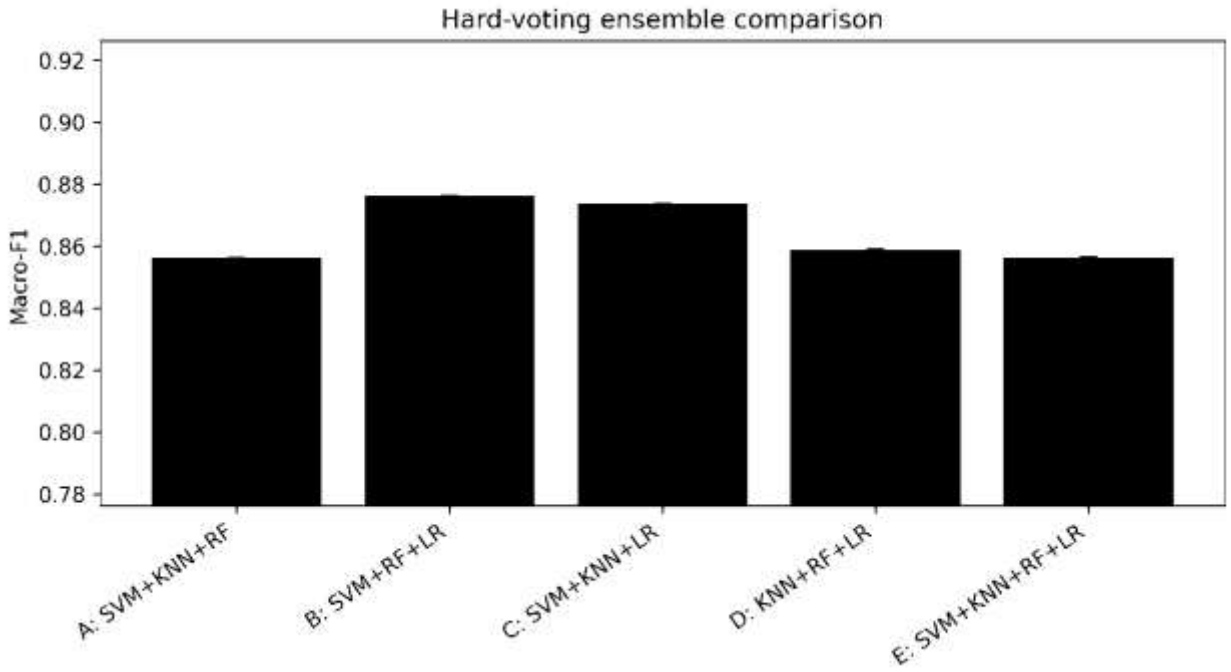

Figure 5. Macro-F1 comparison of the five hard-voting ensemble configurations.

3.8. Out-of-Fold Error Analysis

The pooled out-of-fold confusion matrix for the final RF configuration is presented in Table 10.

Table 10. Pooled out-of-fold confusion matrix for the final RF configuration.

Actual \ Predicted	Non-abusive	Abusive
Non-abusive	17,218	363
Abusive	581	1,952

Across the 20,114 pooled out-of-fold predictions, RF correctly identifies 17,218 non-abusive and 1,952 abusive records. There are 363 false positives and 581 false negatives. The pooled errors therefore contain more false negatives than false positives in this evaluation.

3.9. Repeated Evaluation and Statistical Diagnostics

Repeated five-fold evaluation and the paired RF-versus-LR statistical diagnostics are summarized in Table 11.

Table 11. Out-of-fold and stability diagnostics for Random Forest.

Diagnostic	Result
Pooled OOF observations	20,114
True negatives	17,218
False positives	363
False negatives	581
True positives	1,952
Repeated 5-fold Macro-F1	0.8883 ± 0.0064
Repeated 5-fold abusive recall	0.7709 ± 0.0124
RF − LR Macro-F1 difference	+0.0111
95% bootstrap CI for difference	[0.0063, 0.0159]
Two-sided paired permutation p-value (accuracy)	0.00020

Repeated five-fold evaluation gives macro-F1 = 0.8883 ± 0.0064 and abusive recall = 0.7709 ± 0.0124. Paired OOF analysis gives an RF–LR macro-F1 difference of +0.0111 (95% bootstrap CI [0.0063, 0.0159]). A separate two-sided paired permutation test gives p = 0.00020 for the accuracy difference. These diagnostics are consistent with the observed RF performance advantage; nested evaluation would provide a stricter post-selection estimate.

4. Discussion

4.1. Why Hybrid Word and Character TF-IDF Performs Well

The feature-ablation results show that the hybrid representation performs better than either individual TF-IDF representation in this experiment. Macro-F1 increases from 0.8665 for word TF-IDF and 0.8855 for character TF-IDF to 0.8893 for the hybrid representation. The gain over character TF-IDF is modest, but the hybrid configuration is the highest-scoring feature configuration tested.

4.2. Preprocessing Matters, but the Gain Is Small

Lemmatization improves the no-preprocessing hybrid RF result from 0.8852 to 0.8893 macro-F1. Stemming is close at 0.8887, while stopword removal alone produces 0.8858. The narrow differences indicate that preprocessing is not the dominant source of performance in this dataset. Nevertheless, the controlled ablation provides evidence for selecting lemmatization rather than assuming that a conventional preprocessing recipe is optimal.

4.3. Additional Handcrafted Features Do Not Justify Their Cost

The progressively enriched E3–E7 configurations do not improve on E2. The full E7 configuration achieves macro-F1 = 0.8861, below the hybrid result of 0.8893. Thus, none of the tested auxiliary feature additions provided an incremental macro-F1 gain in this dataset.

4.4. Individual Models and Ensembles Offer Different Operating Points

RF is the strongest individual model by macro-F1, whereas LR has the highest abusive recall and lower measured prediction time. SVM has intermediate abusive recall but much higher measured training and prediction times in the present implementation. KNN has very high specificity but very low abusive recall. The best hard-voting ensemble, SVM + RF + LR, reaches macro-F1 = 0.8762, below RF, although its abusive recall is 0.7931. Thus, none of the tested voting ensembles improves macro-F1 over RF.

4.5. Error Characteristics and Practical Implications

The pooled RF confusion matrix contains 581 false negatives and 363 false positives. LR has higher abusive recall than RF, together with lower specificity and precision in the classifier comparison. Therefore, the tested models provide different operating points: RF has the highest macro-F1, while LR has the highest abusive recall and lower measured prediction time.

4.6. Limitations

The dataset contains short English social-media text and may not generalize to other languages, platforms, or cultural registers. The 12.59% abusive prevalence creates substantial class imbalance; macro-F1 and class-specific metrics are therefore more informative than accuracy alone. The available dataset contains no author,

thread, or timestamp metadata, preventing user- or thread-aware cross-validation and leaving possible record dependence unresolved.
The preprocessing and feature choices are selected sequentially using the same cross-validation framework used for the reported performance. A nested cross-validation experiment would provide a stricter post-selection generalization estimate. The ensemble experiment uses the original word-level TF-IDF pipeline, so its results should be interpreted as a complementary ensemble baseline rather than a direct ensemble extension of the final lemmatized hybrid pipeline.
The exploratory abuse-term feature uses a small internally defined list and should not be interpreted as a validated external lexicon. Transformer-based contextual models and realistic production throughput/latency testing are outside the scope of this study.

5. Conclusions

This study provides a controlled empirical evaluation of preprocessing, text representation, classical machine learning, and ensemble voting for abusive-language detection on 20,114 labeled English social-media text records. The experiments show a clear hierarchy of findings. First, lemmatization is the strongest of the six tested preprocessing strategies, improving the hybrid RF macro-F1 from 0.8852 to 0.8893. Second, the hybrid word- and character-level TF-IDF representation is the strongest tested feature configuration: it outperforms both individual TF-IDF representations and all progressively enriched configurations. Third, the tested lexical, punctuation/emoji, exploratory lexicon, sentiment, and POS additions do not improve the hybrid representation in macro-F1.

RF provides the strongest balanced performance (macro-F1 0.8893 ± 0.0091; accuracy 0.9531 ± 0.0042; precision 0.8437 ± 0.0249; specificity 0.9794 ± 0.0038; abusive recall 0.7706 ± 0.0129). LR has higher abusive recall (0.8160 ± 0.0157) and much faster prediction. The best hard-voting ensemble, SVM + RF + LR, reaches macro-F1 0.8762 and therefore does not improve on RF, although its abusive recall is 0.7931.

Repeated evaluation remains close to the main result (macro-F1 0.8883 ± 0.0064). The pooled OOF confusion matrix contains 581 false negatives and 363 false positives. Overall, the evidence supports lemmatized hybrid word/character TF-IDF + RF as the strongest balanced classical configuration, while LR remains attractive when abusive recall and inference speed are prioritized.

Future work should evaluate the selected pipeline using nested cross-validation, obtain author- or thread-aware metadata where ethically and legally possible, compare against transformer-based contextual models, investigate the linguistic causes of false negatives and false positives, and conduct realistic latency/throughput experiments for moderation workloads.

Author Contributions

Conceptualization, N.G. and S.O.; methodology, N.G.; software, N.G.; validation, N.G. and S.O.; formal analysis, N.G.; investigation, N.G.; data curation, N.G.; writing—original draft preparation, N.G.; writing—review and editing, N.G. and S.O.; visualization, N.G.; supervision, S.O. Both authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding.

Data Availability Statement

The source code, dataset, dataset schema, annotation guidelines, preprocessing procedure, and derived dataset statistics are available through the authors' public repository [13]. The corpus was collected from publicly accessible third-party platforms, and redistribution of individual source messages is subject to the terms and rights applicable to the original platforms. The repository includes the dataset used for the reported experiments together with the associated annotation definitions, preprocessing procedure, and experimental configurations.

Acknowledgments

The authors would like to thank MKSSS's Cummins College of Engineering for Women, Pune, for providing the facilities and support required to carry out this research.

Conflicts of Interest
The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Generative AI Statement
During the preparation of this work, the authors used OpenAI ChatGPT to assist in improving the presentation, organization, and academic writing of the manuscript. The research concept, system design, methodology, implementation, experimental results, and conclusions are the authors' original work. After using this tool, the authors reviewed and edited all AI-assisted content as needed and take full responsibility for the content of this publication.

References

1. E. Spertus, "Smokey: Automatic Recognition of Hostile Messages," in Proceedings of the Innovative Applications of Artificial Intelligence Conference (IAAI-97), Providence, RI, USA, 1997, pp. 1058–1065.
2. D. Yin, Z. Xue, L. Hong, B. D. Davison, A. Kontostathis, and L. Edwards, "Detection of Harassment on Web 2.0," in Proceedings of the Content Analysis in the Web 2.0 Workshop, Madrid, Spain, 2009, pp. 1–7.
3. K. Dinakar, R. Reichart, and H. Lieberman, "Modeling the Detection of Textual Cyberbullying," in Proceedings of the International AAAI Conference on Web and Social Media, vol. 5, no. 3, pp. 11–17, 2011, doi:10.1609/icwsm.v5i3.14209.
4. M. O. Ibrohim and I. Budi, "A dataset and preliminaries study for abusive language detection in Indonesian social media," Procedia Computer Science, vol. 135, pp. 222–229, 2018, doi:10.1016/j.procs.2018.08.169.
5. A. Alakrot, M. Fraifer, and N. S. Nikolov, "Machine Learning Approach to Detection of Offensive Language in Online Communication in Arabic," in 2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering (MI-STA), Tripoli, Libya, 2021, pp. 244–249, doi:10.1109/MI-STA52233.2021.9464402.
6. I. Chung and C.-J. Lin, "TOCAB: A Dataset for Chinese Abusive Language Processing," in 2021 IEEE 22nd International Conference on Information Reuse and Integration for Data Science (IRI), Las Vegas, NV, USA, 2021, pp. 445–452, doi:10.1109/IRI51335.2021.00069.
7. A. D. Asti, I. Budi, and M. O. Ibrohim, "Multi-label Classification for Hate Speech and Abusive Language in Indonesian-Local Languages," in 2021 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 2021, pp. 1–6, doi:10.1109/ICACSIS53237.2021.9631316.
8. S. Khan and A. Qureshi, "Cyberbullying Detection in Urdu Language Using Machine Learning," in 2022 International Conference on Emerging Trends in Electrical, Control, and Telecommunication Engineering (ETECTE), Lahore, Pakistan, 2022, pp. 1–6, doi:10.1109/ETECTE55893.2022.10007379.
9. K. Maity, S. Bhattacharya, S. Saha, and M. Seera, "A Deep Learning Framework for the Detection of Malay Hate Speech," IEEE Access, vol. 11, pp. 79542–79552, 2023, doi:10.1109/ACCESS.2023.3298808.
10. G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," Information Processing & Management, vol. 24, no. 5, pp. 513–523, 1988, doi:10.1016/0306-4573(88)90021-0.
11. L. Breiman, "Random Forests," Machine Learning, vol. 45, pp. 5–32, 2001, doi:10.1023/A:1010933404324.
12. F. Pedregosa et al., "Scikit-learn: Machine Learning in Python," Journal of Machine Learning Research, vol. 12, pp. 2825–2830, 2011
13. N. Gaikwad and S. Ohatkar, "SendWise Dataset and Reproducibility Materials," GitHub, 2026. [Online]. Available: https://github.com/NamrataG7/Abusive_Language_Detection.git [Accessed: Aug. 27, 2026]