Hate Speech Classification: Effectiveness-Efficiency Trade-offs on Twitter

Davide Marchi

Master's Degree Student, University of Pisa

Email: d.marchi5@studenti.unipi.it

TABLE I
LABEL DISTRIBUTION AND TOKEN LENGTH IN THE TRAINING DATA.

label	count	percent	tokens mean	tokens std
Not hateful	29720	92.985	12.517	5.644
Hateful	2242	7.015	13.319	5.160

***Abstract*—We compare five text classifiers for hate speech detection on Twitter: a regex/keyword baseline, linear SVM with TF-IDF features, MiniLM sentence embeddings with logistic regression, an LSTM sequence model, and a zero-shot classifier based on DistilBERT fine-tuned on MNLI. Effectiveness is measured by macro-F1 and accuracy; efficiency is measured by estimated carbon emissions (kg CO_2e) for training and testing, alongside time. Pairwise McNemar tests assess statistical significance. A Pareto analysis of macro-F1 vs test emissions highlights practical trade-offs between effectiveness and environmental cost.**

I. Introduction

We study the effectiveness-efficiency trade-off in hate speech classification with five representative models, from keyword rules to pretrained transformers. We report macro-F1 and accuracy alongside estimated emissions, and test for significance with McNemar's test. Code and artifacts: github.com/davide-marchi/hate-speech-model-comparison.

II. Data

We use the Twitter hate speech dataset (training split) as provided on Kaggle [1]. Table I shows class distribution with token length statistics. We apply standard text cleaning (URLs, mentions, lowercasing).

III. Methods

We compare the following models:

- **Regex/keyword**: a lexical baseline built from indicative keywords and patterns; see early rule-based toxic language detection [2].
- **SVM + TF-IDF**: a linear SVM on word/character TF-IDF features, a strong classical baseline for text classification [3].
- **MiniLM + Logistic Regression**: sentence embeddings from `all-MiniLM-L6-v2` [4] followed by logistic regression.
- **LSTM**: a recurrent model over token sequences [5].
- **Zero-shot (DistilBERT-MNLI)**: using the pretrained model `distilbert-base-uncased-mnli` [6] for zero-shot classification via NLI.

We perform grid search over model-specific hyperparameters; performance is reported on the test set (see Table II).

IV. Experimental Setup

All runs are performed on CPU only for consistency across models. We track both *time* and *carbon emissions (kg CO_2e)* during training and testing. Emissions are estimated via a tracker implemented with the CodeCarbon library [7], which accounts for runtime and power draw; therefore, we emphasize emissions as an efficiency metric rather than wall-clock time alone [8].

V. Results

A. Effectiveness and Emissions

Table II reports test macro-F1, accuracy, test/train time and emissions, plus an efficiency metric (F1 per kg CO_2e) for testing. The SVM+TF-IDF model achieves the highest macro-F1 while keeping emissions low relative to neural alternatives. The keyword baseline is the least energy consuming but lags in macro-F1. The zero-shot DistilBERT provides modest macro-F1 and comparatively higher emissions.

Figure 1 shows the Pareto frontier of macro-F1 vs test emissions.

B. Statistical Significance

We apply McNemar's test to paired predictions. Table III lists pairwise results. The only comparison that is not significant at $\alpha = 0.05$ is between LSTM and MiniLM+LR ($p \approx 0.627$), meaning we cannot conclude a true difference in performance despite the small macro-F1 gap. All other reported pairwise differences are significant; in particular, SVM+TF-IDF significantly outperforms all the alternatives.

C. Error Analysis

Beyond the metrics reported here, each classifier's confusion matrix and the list of misclassified examples are available in the repository. For illustration, we show the confusion matrix of the best-performing classifier (SVM+TF-IDF) in Figure 2.

TABLE II
EFFECTIVENESS AND EMISSIONS SUMMARY.

model	macro-F1	accuracy	test time (s)	train time (s)	test emissions (kg)	train emissions (kg)	efficiency (F1/kg)
SVM+TF-IDF	0.864	0.965	0.114	7.633	8.212 121e−7	2.566 488e−5	1 052 443.865
MiniLM+LR	0.728	0.888	12.070	274.935	2.229 687e−5	0.000 388	32 670.486
LSTM	0.717	0.886	0.500	800.165	1.335 282e−6	0.001 126	536 784.599
Regex/Keyword	0.705	0.936	0.075	1.398	7.687 961e−7	2.590 393e−6	916 840.448
Zero-shot	0.593	0.860	253.772	257.134	0.000 452	0.000 455	1312.272

TABLE III
MCNEMAR PAIRWISE TESTS (P-VALUE AND SIGNIFICANCE).

model_a	model_b	p_value	significant
SVM+TF-IDF	Zero-shot	2.356e−128	True
LSTM	SVM+TF-IDF	3.540e−92	True
MiniLM+LR	SVM+TF-IDF	1.687e−86	True
Regex/Keyword	Zero-shot	3.594e−66	True
LSTM	Regex/Keyword	6.524e−29	True
Regex/Keyword	SVM+TF-IDF	4.319e−28	True
MiniLM+LR	Regex/Keyword	3.763e−26	True
MiniLM+LR	Zero-shot	1.970e−7	True
LSTM	Zero-shot	1.150e−6	True
LSTM	MiniLM+LR	0.627	False

VI. DISCUSSION

Our results suggest that strong linear baselines remain competitive on short social media text: SVM+TF-IDF provides the best macro-F1 at a favorable emissions footprint. Pretrained embeddings (MiniLM) and LSTM narrow the effectiveness gap but incur higher emissions per test example. Zero-shot classification offers convenience and label flexibility without task-specific training, but its emissions are comparatively higher. The keyword baseline is the least energy consuming.

Generalization to other hate speech datasets without retraining is plausible, but sensitivity to context matters. When syntax and lexicon shift, syntax-based approaches such as regex and SVM+TF-IDF are likely to degrade the most, while semantic approaches like sentence embeddings and LSTM (leveraging pretrained word embeddings) should be more robust.

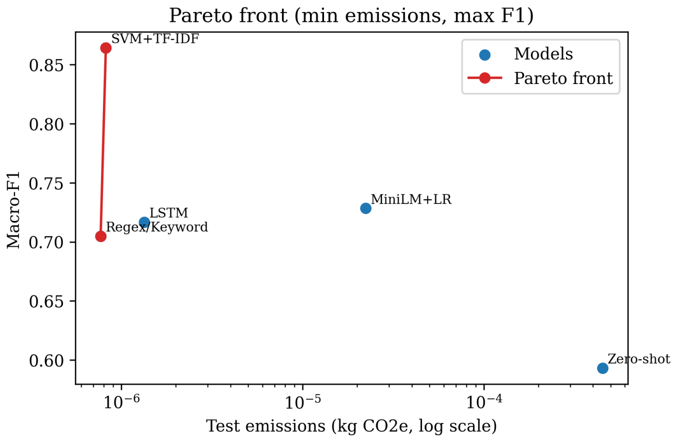

Fig. 1. Macro-F1 vs test emissions with a Pareto frontier.

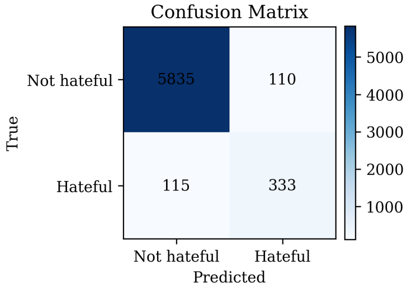

Fig. 2. Confusion matrix for SVM+TF-IDF on the test set.

VII. LIMITATIONS

Neural models such as LSTM, zero-shot transformers, and sentence-embedding pipelines can scale far beyond what we ran here: many hyperparameters, larger architectures, and heavier training that often improve accuracy but raise emissions. Due to hardware limits, we explored small configurations, whereas the SVM+TF-IDF baseline is near its optimal regime on this dataset. With more compute and a larger corpus, SVM+TF-IDF might improve slightly, but the other approaches could surpass it, at a higher carbon footprint.

VIII. CONCLUSION

On this dataset, SVM+TF-IDF is a strong effectiveness-efficiency choice, while with sufficient time and compute fully fine-tuned semantic models (e.g., BERT) would likely outperform it and generalize better across datasets.

REFERENCES

[1] “Twitter Hate Speech,” Kaggle, available at: https://www.kaggle.com/vkrahul/twitter-hate-speech (accessed Nov. 2025).

[2] E. Spertus, “Smokey: Automatic Recognition of Hostile Messages,” in IAAI, 1997.

[3] T. Joachims, “Text Categorization with Support Vector Machines: Learning with Many Relevant Features,” in ECML, 1998.

[4] W. Wang, F. Wei, L. Dong, H. Bao, N. Yang, and M. Zhou, “MiniLM: Deep Self-Attention Distillation for Task-Agnostic Compression of Pre-Trained Transformers,” arXiv:2002.10957, 2020.

[5] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Computation, 9(8):1735-1780, 1997.

[6] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, “DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,” arXiv:1910.01108, 2019.

[7] CodeCarbon, “CodeCarbon: Track Carbon Emissions from Machine Learning,” available at: https://github.com/mlco2/codecarbon (accessed Nov. 2025).

[8] A. Lacoste, A. Sasha Luccioni, V. Schmidt, and T. Dandres, “Quantifying the Carbon Emissions of Machine Learning,” arXiv:1910.09700, 2019.