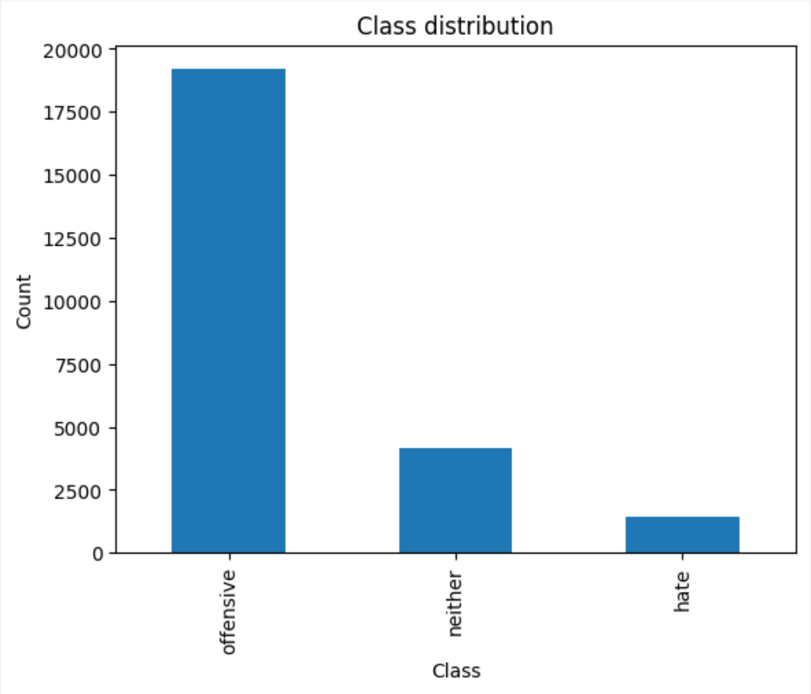
I HATE YOU

Hate speech detection + lexical explanations (TF–IDF models)

Text Mining and Sentiment Analysis • University of Milan

Elaheh Zohdi • Matricola 13731A • AY 2023–2024

- Task: classify tweets into hate / offensive / neither
- Goal: not only performance, but also interpretation
- Explainability: LR coefficients + SHAP for XGBoost



Class imbalance (hate / offensive / neither)

The project

Research question, measurable objectives, and what will be shown

- Build models to classify hate speech, offensive language, or neither
- Evaluate with macro-F1 and F1(hate) under strong imbalance
- Extract relevant terminology per class (lexical explanations)

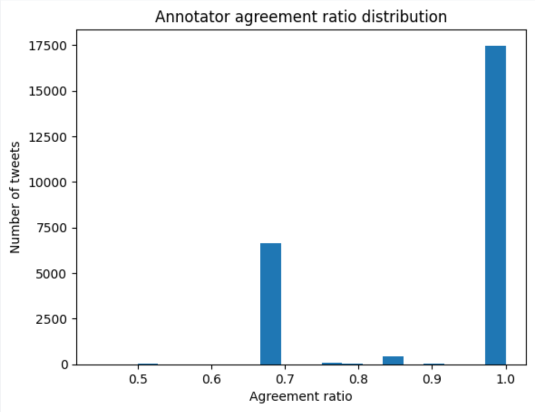
Research question

How well do TF–IDF baselines distinguish hate speech from general offensive language, and which terms drive model decisions?

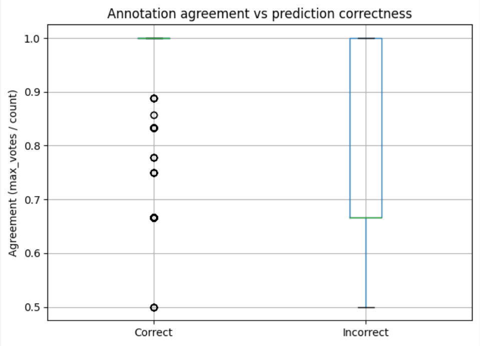
Why hate speech detection is hard

Overlap, ambiguity, and label noise

- No single universal definition; labels depend on guidelines
- Lexical overlap: profanity appears in multiple classes
- Minority class (hate) is small → accuracy can be misleading
- Some tweets are ambiguous even for humans



Annotation agreement ratio

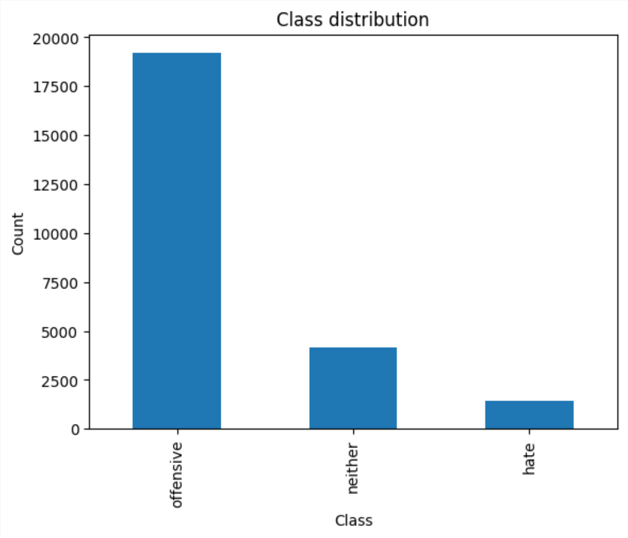


Agreement vs correctness

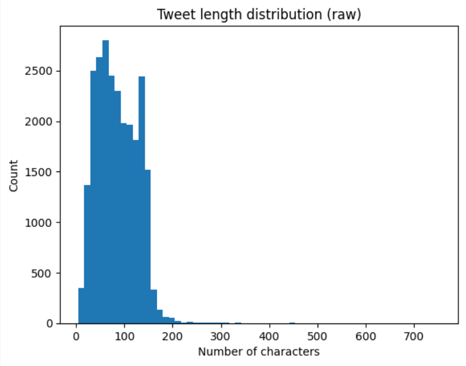
Dataset

Davidson et al. Twitter dataset (3 classes)

- 24,783 tweets (English) labeled by majority vote
- Classes: hate (5.8%), offensive (77.4%), neither (16.8%)
- No missing tweets, no duplicates in the file



Strong class imbalance

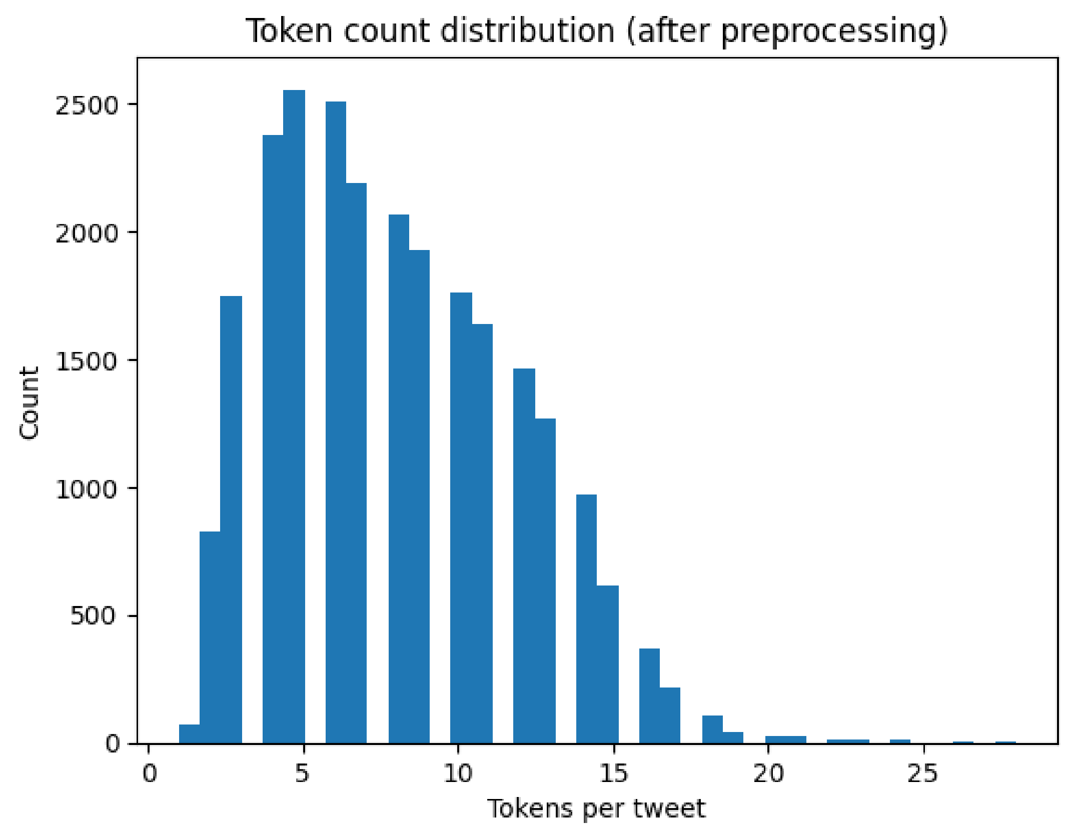


Raw tweet length distribution

Text preprocessing

Normalize tweets while keeping discriminative cues

- Lowercasing + contractions expansion
- Remove URLs; replace @mentions with mentiontoken; remove RT
- Digits → words; punctuation/special chars removed
- Reduce long character repeats (e.g., soooo → soo)
- Tokenize → remove stopwords → Snowball stemming



Tokens per tweet after preprocessing

Feature representation

TF–IDF on word n-grams (1–2)

Why TF–IDF here?

- Strong baseline for short texts (tweets)
- Sparse + fast to train; interpretable weights
- Fit vectorizer on train only (avoid leakage)

TF–IDF setup (this project)

ngram_range=(1,2) • min_df=3 • max_df=0.9 • sublinear_tf=True

Train: 19,826 × 10,813 • Test: 4,957 × 10,813

Vocabulary size: 10,813 (after preprocessing + stemming)

Models

Linear baseline + boosted trees; handle imbalance explicitly

Compared models

- Logistic Regression (class_weight='balanced')
- XGBoost baseline (multi-class softprob)
- XGBoost + class-balanced sample weights

Metrics

Accuracy + macro-F1 + F1(hate)

Main error to watch: hate ↔ offensive confusion

Results

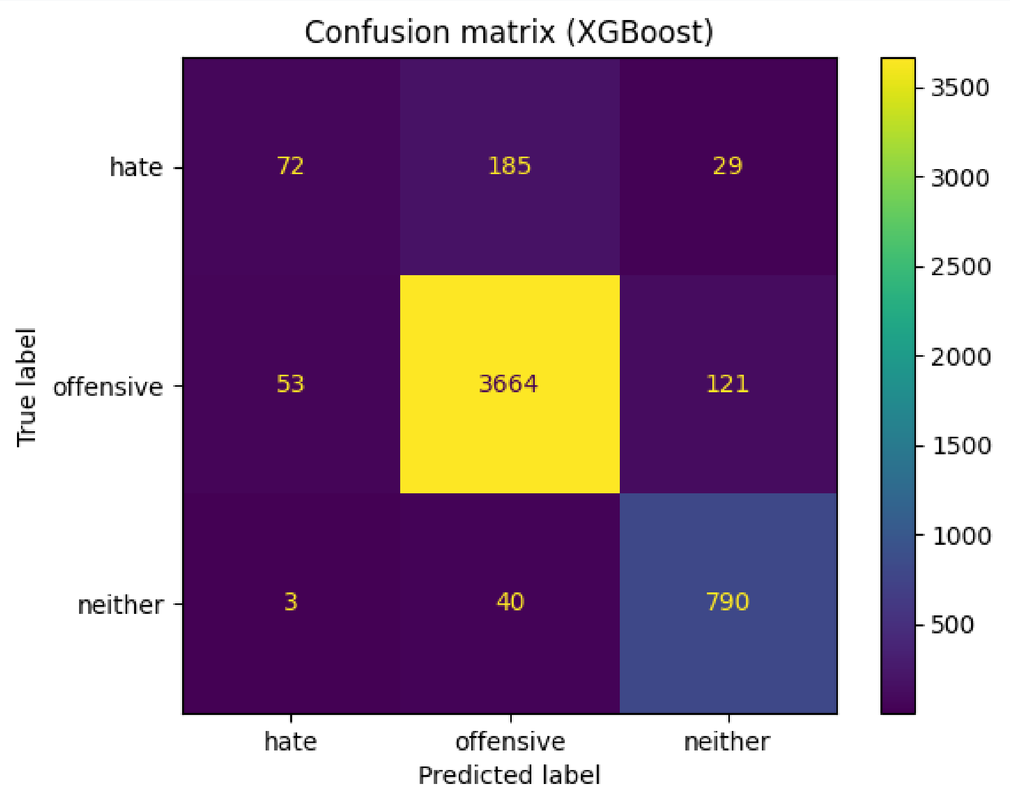
Performance on the held-out test set

Model	Accuracy	Macro-F1	F1(hate)	Note
LogReg (balanced)	0.870	0.743	0.443	Best F1(hate)
XGBoost	0.913	0.729	0.348	High accuracy, low hate recall
XGBoost + weights	0.873	0.742	0.424	**Better hate detection**

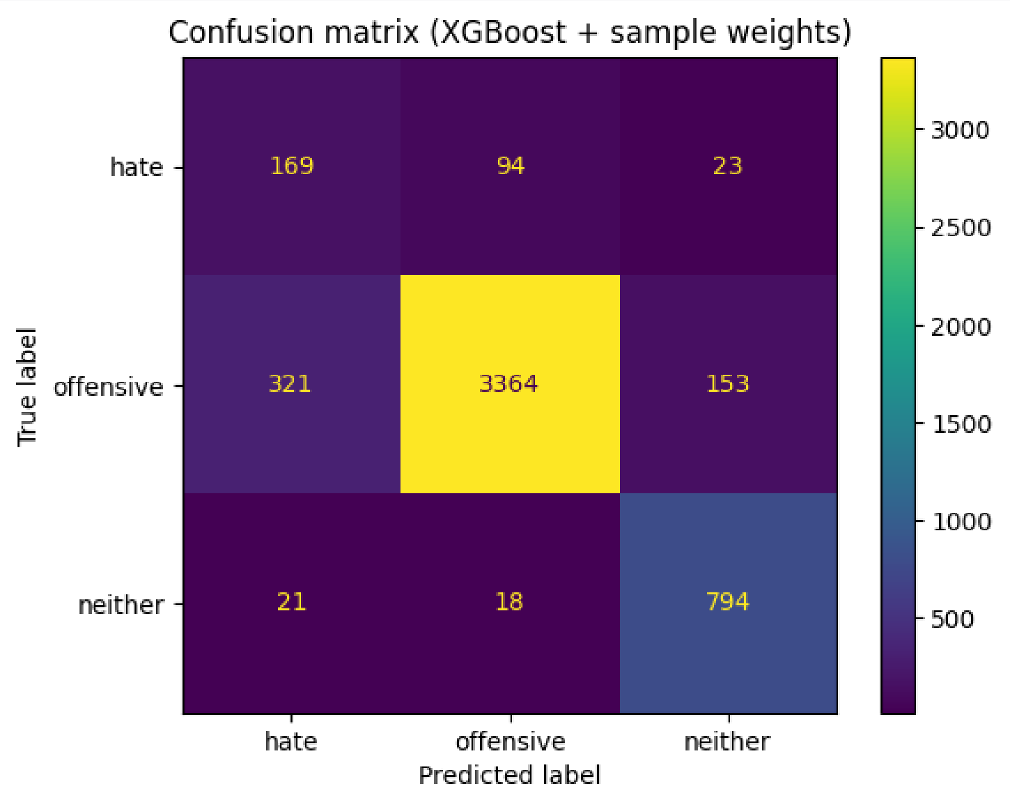
Takeaway: weighting trades a bit of accuracy for much better hate recall.

Where the model makes mistakes

Hate is often confused with offensive



XGBoost (unweighted)

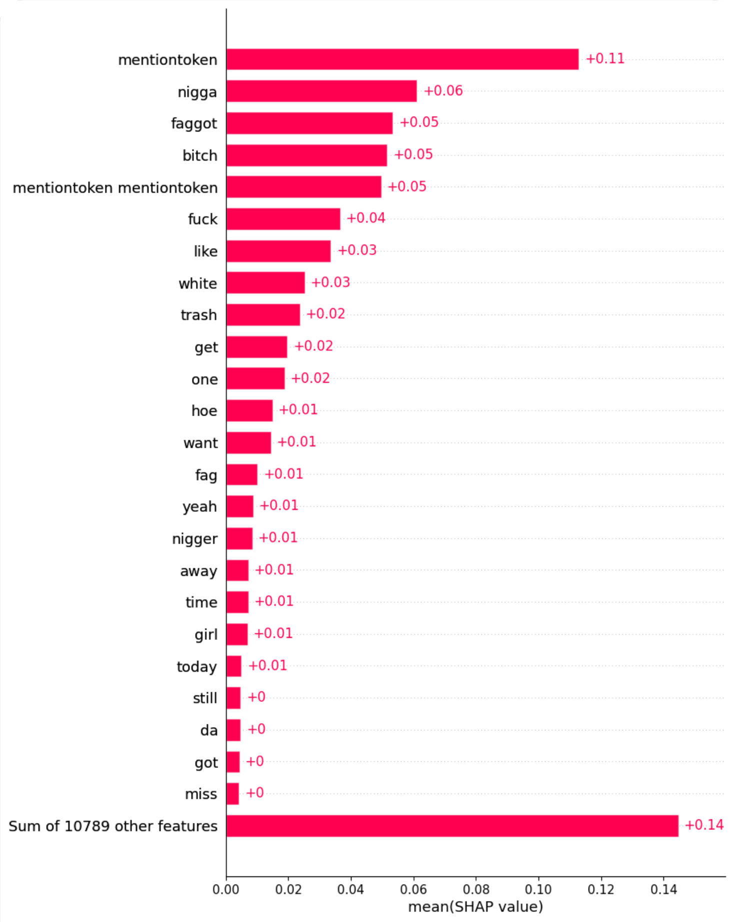


XGBoost + sample weights

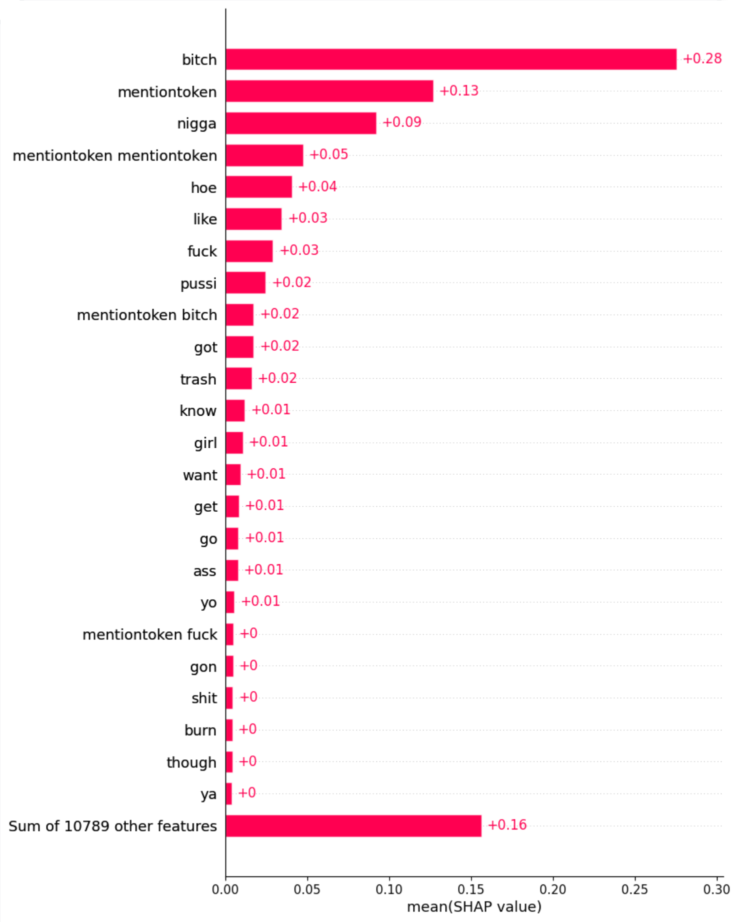
- Hate true-positives improve: 72 → 169 (but more offensive → hate)
- Remaining main error: hate → offensive (185 → 94)

Relevant terminology (explainability)

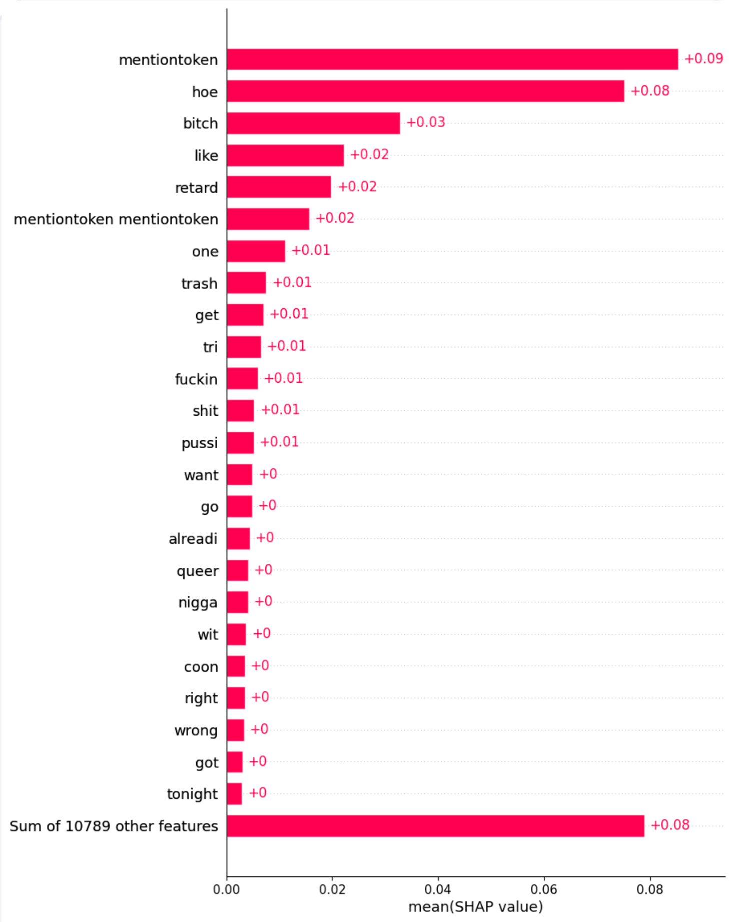
LR coefficients + SHAP for weighted XGBoost



SHAP (mean |SHAP|) — Hate



SHAP (mean |SHAP|) — Offensive

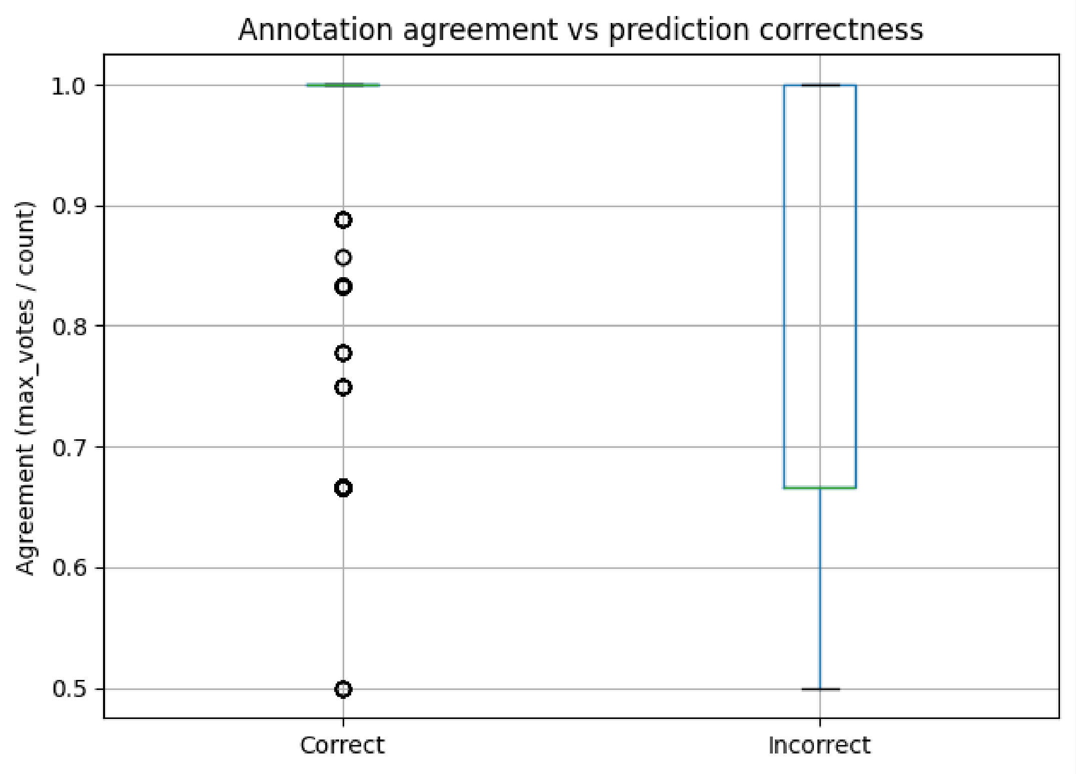


SHAP (mean |SHAP|) — Neither

Error analysis

Mistakes concentrate on low-agreement tweets

- Mean agreement (correct): 0.925
- Mean agreement (incorrect): 0.776
- Many errors are borderline / noisy labels
- Hate vs offensive remains the key confusion



Agreement vs prediction correctness

Conclusions & future work

What I learned + next steps

Key takeaways

- TF–IDF baselines are strong, but hate vs offensive is still tricky
- Cost-sensitive training improves minority-class detection
- Lexical explanations are coherent, but highlight dataset bias

Future work ideas

- Try contextual embeddings (BERT-like) and compare explainability
- Check user-level leakage (same authors in train/test)
- Study time drift: lexicon changes over years
- Explore clearer label definitions / re-annotation

Thank you!