Detecting Toxic Comments: How Far Can AI Go?

Sunesh Praveen Raja Sundarasami, Aaron Cuthinho, Prof. Dr. Jörn Hees, MSc Tim Metzler

Abstract

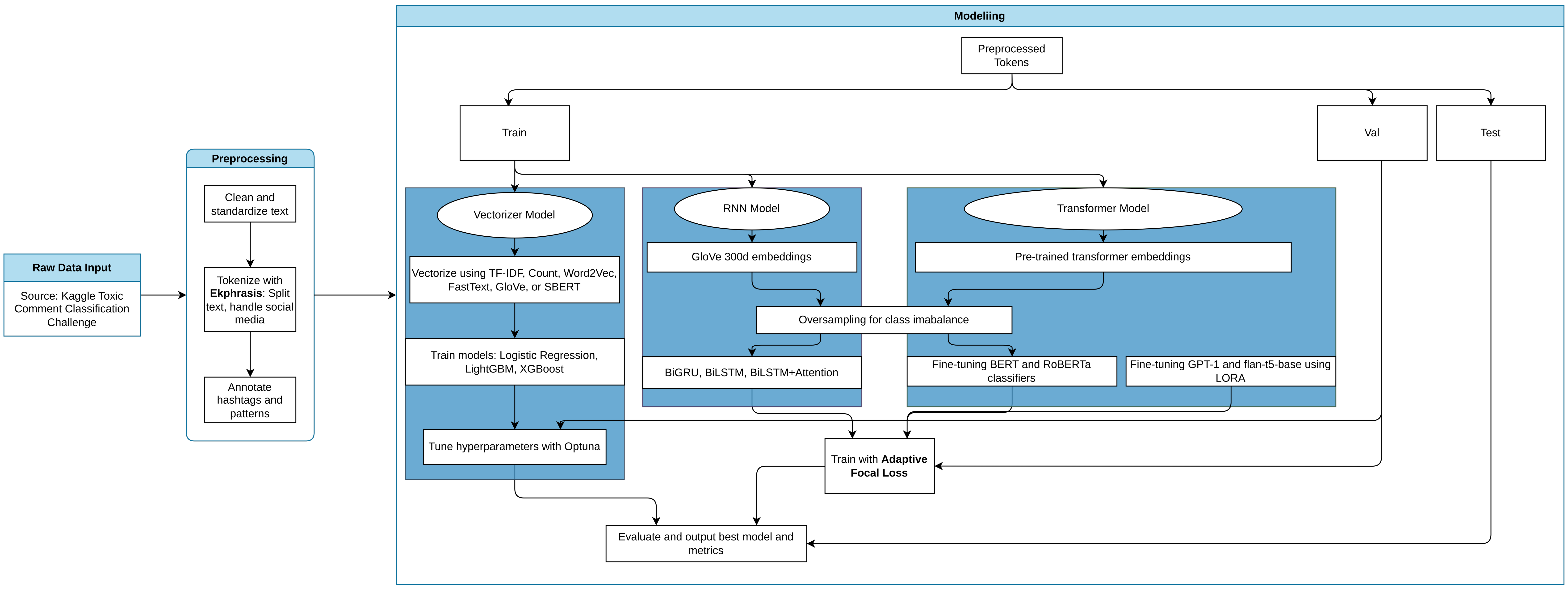
Online platforms require robust automated systems to moderate user-generated content and maintain healthy online communities. In this project, we address multi-label toxic comment classification using a comprehensive NLP pipeline. Our workflow begins with raw data preprocessing, including text cleaning, social media-specific tokenization with Ekphrasis, and annotation of hashtags and patterns. We explore three modeling strategies: traditional vectorizer-based models (TF-IDF, Word2Vec, FastText, SBERT with Logistic Regression, LightGBM, XGBoost), RNN architectures (BiGRU, BiLSTM, BiLSTM+Attention with GloVe embeddings), and state-of-the-art transformer models (BERT, RoBERTa, GPT-1, FLAN-T5) leveraging pretrained embeddings and fine-tuning with advanced techniques such as LORA and adaptive focal loss. Oversampling is applied to address class imbalance. Hyperparameter optimization is performed using Optuna. Our evaluation demonstrates that transformer-based models significantly outperform traditional approaches, highlighting the effectiveness of modern NLP architectures for nuanced toxic comment detection.

Methodology

Pipeline stages:

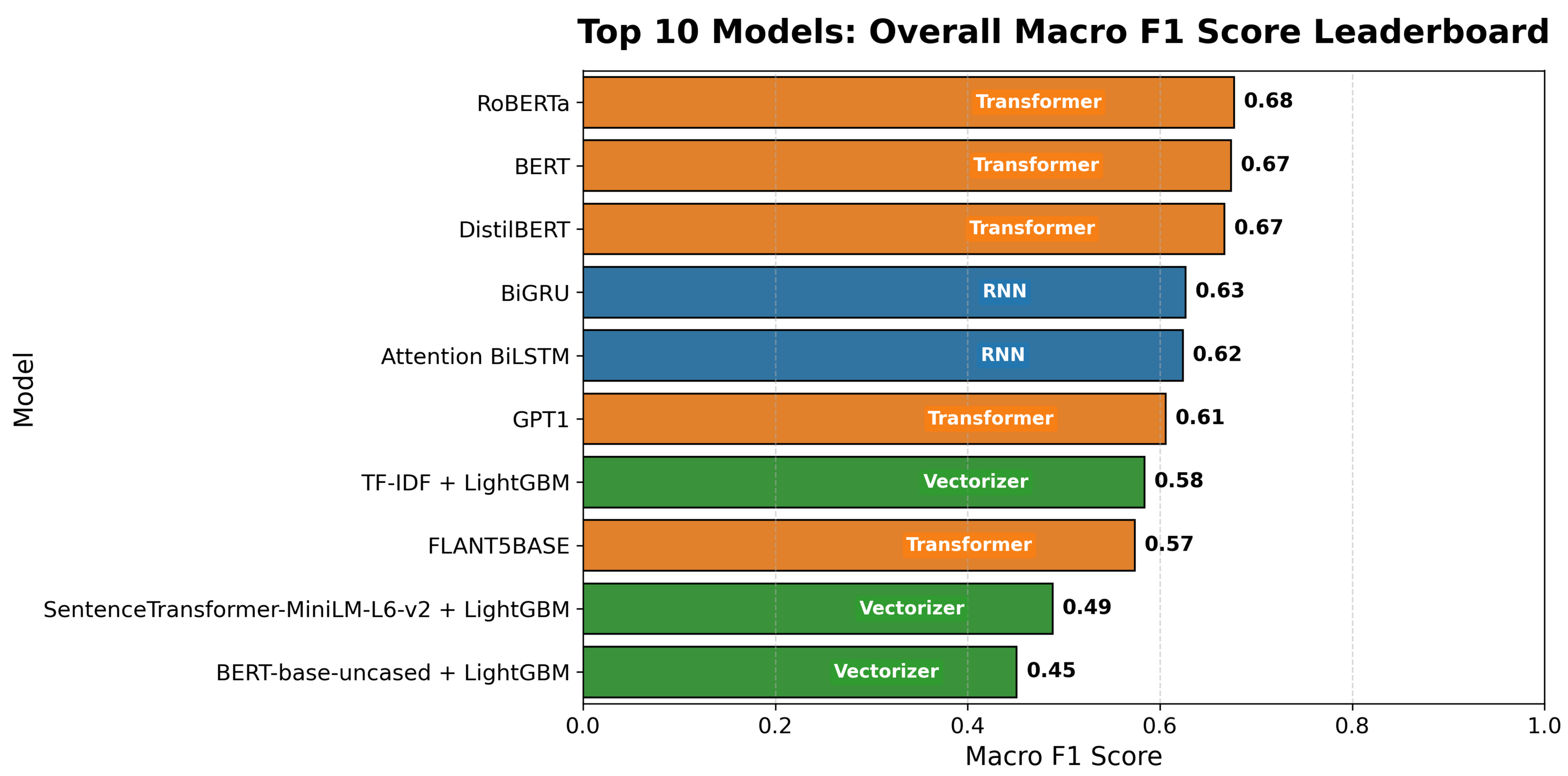
1. **Preprocessing:** Clean text and tokenize with Ekphrasis.
2. **Feature Representation:** Use TF-IDF, Word2Vec, GloVe, SBERT, or transformer embeddings.
3. **Modeling:** Train vectorizer models (Logistic Regression, LightGBM, XGBoost), RNNs (BiGRU, BiLSTM, Attention), and fine-tune transformers (BERT, RoBERTa, GPT-1, FLAN-T5 with LoRA).
4. **Optimization:** Apply adaptive focal loss, oversampling, and threshold tuning.
5. **Evaluation:** Assess with F1-score, ROC-AUC, and per-class analysis.

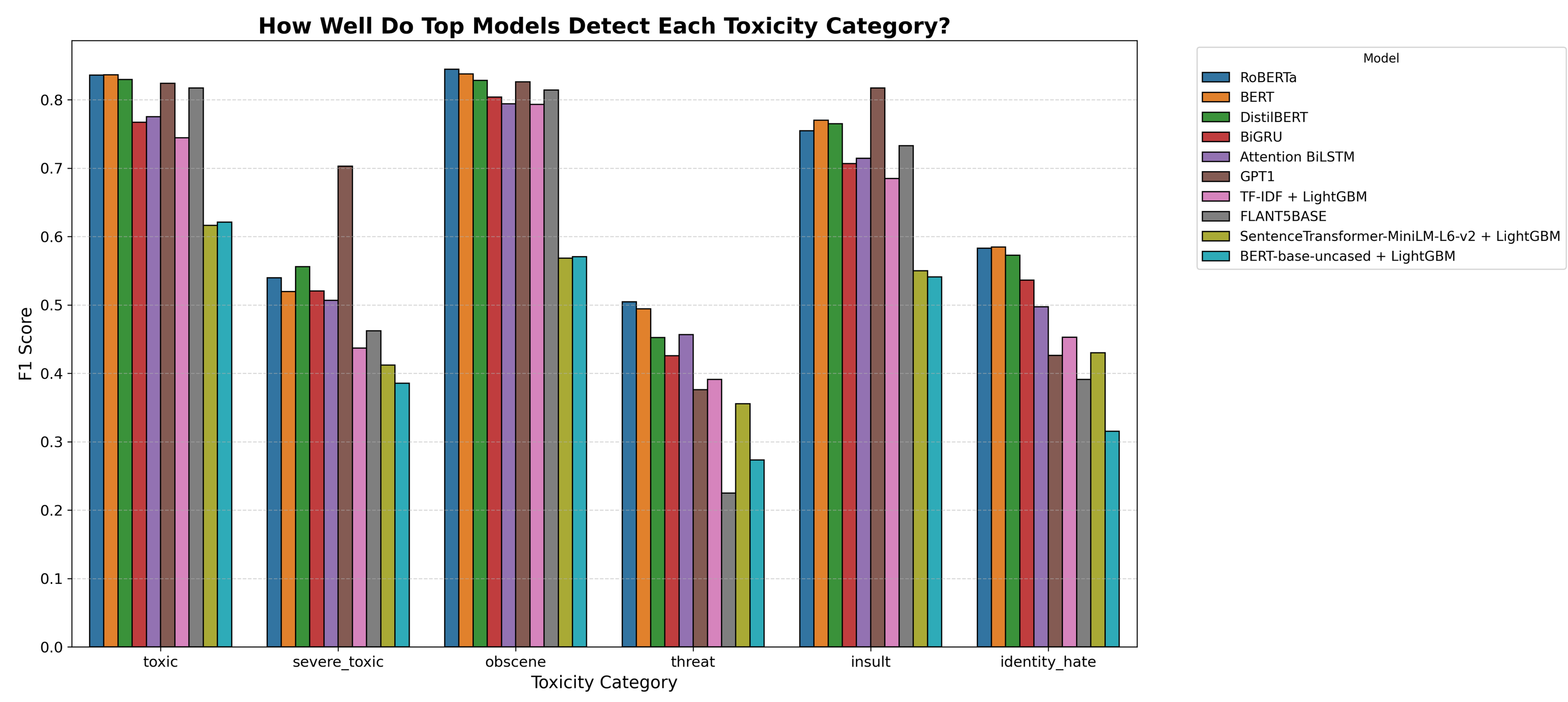
Pipeline Overview



Overview of the model pipeline for toxic comment classification

Results





Summary: Our results show that transformer models (RoBERTa, BERT, DistilBERT, GPT1) consistently outperform RNNs and vectorizer-based models both overall and across toxicity categories. However, rare classes like *threat* and *identity hate* remain difficult for all models. I found that even high-performing models tend to incorrectly flag neutral identity statements (e.g., "I'm Muslim", "I'm gay") as toxic, indicating bias learned from the dataset. In contrast, zero-shot large language models (LLMs) classify such statements as non-toxic, as expected. This highlights the importance of careful evaluation for unintended bias in toxic comment classifiers.

Link to Code and Usage Guidelines

Please scan the QR code to the side or visit the link mentioned below to access the code and usage guidelines.
`https://github.com/SuneshSundarasami/Multi_Label_Toxic_Comment_Classifier/`

References

[1] cjadams, Jeffrey Sorensen, Julia Elliott, Lucas Dixon, Mark McDonald, nithum, and Will Cukierski. Toxic comment classification challenge. `https://kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge`, 2017. Kaggle.

[2] Jigsaw Wikipedia and Google. Wikipedia comments dataset. `https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge/data`, 2017. Kaggle.

Acknowledgement

We would like to thank Prof. Dr. Jörn Hees and MSc Tim Metzler for giving us this opportunity and providing valuable guidance during this project.

Contact

Prof. Dr. Jörn Hees : joern.hees@h-brs.de
MSc Tim Metzler: tim.metzler@h-brs.de
Sunesh Praveen Raja Sundarasami: sunesh.sundarasami@smail.inf.h-brs.de
Aaron Cuthinho: aaron.cuthinho@smail.inf.h-brs.de

b-it Bonn-Aachen International Center for Information Technology

Hochschule Bonn-Rhein-Sieg University of Applied Sciences