Toxic comment classification

Vinay Sanga, Tom Pruggmayer
Università degli studi dell'Aquila, Italy
Email: vinay.sanga@student.univaq.it, tomalexander.pruggmayer@student.univaq.it

Academic year: 2024–2025

1 Introduction

This project is using the dataset from the Toxic Comment Classification Challenge [1] from kaggle.com and aims to compare different models for toxic comment classification. The data set contains english comments from Wikipedia that have been labeled by humans for toxic contents. There are six different parameters for toxicity: `toxic`, `severe_toxic`, `obscene`, `threat`, `insult`, and `identity_hate`. Each of those parameters can have the value of 0 or 1 depending on whether it applies to a comment or not.

2 Experimentation

To classify toxic comments effectively, we experimented with both classical machine learning (ML) models and large language models (LLMs). The goal was to evaluate their respective performances in detecting different types of toxicity.

2.1 Fixing the Train Data

We observed in the previous experiment with the complete dataset that there was a significant imbalance in the data distribution. This section describes the steps taken to analyze and address this issue.

2.1.1 Analysis of Toxic Labels

The dataset consists of six toxic categories: `toxic`, `severe_toxic`, `obscene`, `threat`, `insult`, and `identity_hate`. The analysis focused on three aspects:

1. **Exclusive Labels:** Comments with exactly one toxic label were identified. The findings showed:
   - **Total single-label comments:** 6,360
   - **Toxic alone:** 5,666 (approximately 89% of exclusive labels)
   - **Severe Toxic:** Never appeared alone (0 exclusive cases)
   - **Threat:** Only 22 exclusive cases

   This suggests that the `severe_toxic` label always appears alongside other toxic categories.
2. **Combination Labels:** Comments with multiple toxic labels were analyzed, revealing:
   - **Total multi-label comments:** 9,865
   - **2-label comments:** 3,480 (35.3% of multi-label cases)
   - **3-label comments:** 4,209 (42.7%, more than 2-label cases)
   - **4-label comments:** 1,760 (17.8%)
   - **5-label comments:** 385 (3.9%)
   - **6-label comments:** 31 (0.3%)

3. **Total Label Distribution:** The overall distribution of labels showed that the `toxic` label was dominant:
    - **Toxic:** 15,294 comments
    - **Obscene:** 8,449 comments (55.2% of toxic cases)
    - **Insult:** 7,877 comments (51.5% of toxic cases)
    - **Severe Toxic:** 1,595 comments (10.4% of toxic cases)
    - **Identity Hate:** 1,405 comments (9.2% of toxic cases)
    - **Threat:** 478 comments (3.1% of toxic cases)

2.1.2 Findings and Insights

The dataset exhibits a **hierarchical nature**, where `toxic` serves as a parent category for most toxic comments. Furthermore, the strong correlation between labels suggests that multi-label classification is critical for accurate predictions. The analysis also highlights significant **class imbalances** in the dataset:

- Disproportionate distribution of single-label vs. multi-label comments
- Uneven frequency of different toxic categories
- Imbalances in label combinations

2.1.3 Dataset Balancing Strategy

To address these imbalances, the dataset was resampled to create balanced subsets with varying toxic-to-non-toxic ratios:

- **1:1 Ratio:** Equal number of toxic and non-toxic comments
- **2:1 Ratio:** Twice as many non-toxic comments as toxic ones
- **3:1 Ratio:** Three times as many non-toxic comments as toxic ones

These balanced datasets will be used in the subsequent training steps to improve model performance. We did not use the 1:1 split because it gave poor results on preliminary test.

2.2 Fixing the Test Data

To ensure the test dataset was clean and properly formatted for model evaluation, preprocessing steps were applied.

2.2.1 Handling Invalid Labels

It was observed that some rows contained labels with a value of `-1`, which were not meaningful for training or evaluation. These `-1` values were introduced as part of the original Kaggle competition, where certain test samples were provided without ground-truth labels. To ensure data quality and prevent misleading evaluation, these rows were removed from the dataset.

- Rows where any of the labels (`toxic`, `severe_toxic`, `obscene`, `threat`, `insult`, `identity_hate`) were `-1` were filtered out.
- The index of the dataset was reset after filtering to maintain consistency.

2.2.2 Finalizing the Processed Test Data

After cleaning, the processed test dataset was saved for use in model evaluation. This dataset ensures that all labels are valid and that the test data accurately reflects the structure of the training data. Table 1 shows the support of processed test data which like the train data, show the imbalance.

Class	Support	Percentage
Toxic	6090	42.0%
Severe Toxic	367	2.5%
Obscene	3691	25.4%
Threat	211	1.5%
Insult	3427	23.7%
Identity Hate	712	4.9%

Table 1: Support of the processed test data

2.3 Using classical ML approaches

In the first part of the project, we used classical ML models, in our case the support vector machine and logistic regression. We also tested the two vectorization methods word2vec and TF-IDF for both models. So, this first approach gave us four models for comparison. All models were separately trained with the 2:1 ratio and the 3:1 data set.

2.3.1 Project overview

- Loading the valid test data and the balanced training data.
- Preprocessing the data by removing special characters, URLs, digits and stop words. In addition, lemmatization was applied to all comments. [2]
- After that, the following models were trained:
  - Logistic regression model with Word2Vec vectorization
  - Support vector machine with Word2Vec vectorization
  - Logistic regression model with TF-IDF vectorization
  - Support vector machine with TF-IDF vectorization
- After each training, we tested the models with our valid test data and generated a classification report containing the precision, recall and f1-score for every toxicity parameter.

2.3.2 Results

The comparison of classification reports across the four models (ref. tables 2 and 3) revealed that models utilizing **TF-IDF vectorization** outperformed those using **Word2Vec**.

- **Logistic Regression (LR) with TF-IDF** consistently achieved the highest scores, surpassing the Support Vector Machine (SVM).
- The models trained on **2:1 and 3:1 dataset ratios** exhibited **similar overall performance**, with only minor variations in specific toxicity categories.
- **LR with TF-IDF** consistently recorded the highest F1 scores for the `toxic` and `obscene` categories across both dataset ratios.
- A significant drawback was observed in the **SVM with Word2Vec** model, which consistently produced an F1 score of **0.00** for the `severe_toxic`, `threat`, and `identity_hate` categories, indicating its inefficiency in handling rare toxicity labels.

These findings highlight the importance of **feature representation** in toxicity classification and suggest that TF-IDF vectorization is a more effective approach than Word2Vec for classical machine learning models.

Models	toxic	severe toxic	obscene	threat	insult	identity hate
LR word2vec	0.57	0.21	0.60	0.22	0.52	0.30
SVM word2vec	0.58	0.00	0.61	0.00	0.52	0.00
LR TF-IDF	0.64	0.34	0.70	0.28	0.63	0.38
SVM TF-IDF	0.62	0.25	0.67	0.31	0.63	0.45

Table 2: Comparison of the F1 score of the classical approaches trained with the 2:1 ratio data set

Models	toxic	severe toxic	obscene	threat	insult	identity hate
LR word2vec	0.60	0.21	0.59	0.23	0.51	0.26
SVM word2vec	0.60	0.00	0.60	0.00	0.51	0.00
LR TF-IDF	0.66	0.34	0.70	0.28	0.63	0.38
SVM TF-IDF	0.64	0.22	0.68	0.31	0.63	0.43

Table 3: Comparison of the F1 score of the classical approaches trained with the 3:1 ratio data set

2.4 Using LLM

The final aim of this project is to implement an application to parse and analyze toxic content using a Large Language Model (LLM). With the insights gained from dataset analysis and balancing, the next steps involve fine-tuning an LLM for robust toxicity detection. We used huggingface [3] libraries for all the LLM related tasks such as tokenization, downloading the checkpoints, training and testing.

2.4.1 Model Variations and Configurations

Multiple variations of language models were experimented with, specifically focusing on **BERT-based** and **DistilBERT-based** architectures. These models were trained using different data balancing strategies to optimize performance.

- **DistilBERT Models:**
  - `distilbert-21-10ep` — Trained with a **2:1** toxic-to-non-toxic ratio over **10 epochs**. Balanced dataset ensured better recall for minority toxic labels.
  - `distilbert-31-10ep` — Trained with a **3:1** toxic-to-non-toxic ratio over **10 epochs**. Enhanced generalization, especially for rare toxic categories.
  - `distilbert-31` — Trained with a **3:1 ratio** toxic-to-non-toxic ratio with **5 epochs**.
  - `distilbert-21` — Trained with a **2:1 ratio** toxic-to-non-toxic ratio for **5 epochs**.
- **BERT Models:**
  - `bert-31-10ep` — Trained with a **3:1 ratio over 10 epochs**, leveraging the **full capacity of BERT** for high accuracy. Achieved the best performance across all toxic categories.
  - `bert-21-10ep` — Trained with a **2:1 ratio over 10 epochs**, offering a balance between speed and accuracy. Performed slightly worse than `bert-31-10ep`, especially in rare toxicity classes.
  - `bert-31` — Trained with a **3:1 ratio** toxic-to-non-toxic ratio for **5 epochs**.
  - `bert-21` — Trained with a **2:1 ratio** toxic-to-non-toxic ratio with **5 epochs**.

The choice of **DistilBERT vs. BERT** was influenced by computational efficiency. While **BERT** models consistently achieved higher accuracy, **DistilBERT** models provided significantly **faster inference times** with only a **small performance drop**.

2.4.2 Results

The following table summarizes the performance of these models based on key evaluation metrics. Here are the finding which are represented in tables 4 and 5.

- **BERT models consistently outperformed DistilBERT**, especially in high-frequency toxicity categories.

- **bert-31-10ep achieved the best validation performance**, particularly for the `severe_toxic`, `threat`, and `identity_hate` categories.
- Models trained with a **3:1 toxic-to-non-toxic ratio** (`bert-31`, `distilbert-31`) performed better in rare toxicity categories than their 2:1 counterparts.
- **bert-31-10ep achieved the highest test scores**, confirming strong generalization to unseen data.
- DistilBERT models **performed worse on low-frequency labels** (`severe_toxic`, `threat`, `identity_hate`) but maintained competitive scores for `toxic`, `obscene`, and `insult`.
- **DistilBERT-31-10ep outperformed some BERT models** in generalization, particularly in the `threat` and `identity_hate` categories.
- The performance gap between validation and test scores was **minimal**, suggesting **low overfitting** and a robust training strategy.
- **Micro-averaged F1 scores for BERT models were consistently higher** ($\sim$0.85–0.87), reinforcing their superior overall classification performance.
- Micro-averaged F1 scores remained consistently high, but macro-averaged F1 scores varied slightly due to differences in classification of rare toxic categories.

Models	Toxic	Severe Toxic	Obscene	Threat	Insult	Identity Hate	Micro Avg	Macro Avg
bert-21	0.89	0.49	0.84	0.56	0.78	0.59	0.82	0.69
bert-21-10ep	0.89	0.49	0.84	0.57	0.78	0.59	0.82	0.69
bert-31	0.89	0.44	0.85	0.50	0.78	0.56	0.82	0.67
bert-31-10ep	0.89	0.50	0.84	0.58	0.77	0.57	0.82	0.69
distilbert-21	0.90	0.46	0.86	0.25	0.79	0.53	0.83	0.63
distilbert-21-10ep	0.89	0.50	0.84	0.55	0.77	0.55	0.82	0.68
distilbert-31	0.89	0.44	0.85	0.53	0.78	0.59	0.82	0.68
distilbert-31-10ep	0.88	0.49	0.84	0.58	0.77	0.57	0.82	0.69

Table 4: Validation F1 Scores for LLM approach

Models	Toxic	Severe Toxic	Obscene	Threat	Insult	Identity Hate	Micro Avg	Macro Avg
bert-21	0.81	0.67	0.92	0.83	0.90	0.83	0.85	0.83
bert-21-10ep	0.82	0.64	0.91	0.79	0.89	0.81	0.85	0.81
bert-31	0.89	0.45	0.85	0.52	0.78	0.59	0.82	0.68
bert-31-10ep	0.86	0.64	0.93	0.80	0.89	0.83	0.87	0.82
distilbert-21	0.90	0.46	0.86	0.25	0.78	0.54	0.83	0.63
distilbert-21-10ep	0.89	0.50	0.85	0.49	0.78	0.60	0.83	0.69
distilbert-31	0.89	0.44	0.85	0.54	0.78	0.59	0.82	0.68
distilbert-31-10ep	0.95	0.59	0.92	0.74	0.89	0.80	0.91	0.81

Table 5: Test F1 Scores for LLM approach

3 Analytical Comparison of Models

This section presents a comparative analysis of classical machine learning (ML) approaches and large language models (LLMs) for toxic comment classification. The primary evaluation metric is the **F1 score**, which measures the balance between precision and recall. The models are assessed based on their ability to classify different toxicity categories, namely *toxic, severe toxic, obscene, threat, insult, and identity hate*.

3.1 Performance of Classical ML Approaches

The classical ML models tested include **Logistic Regression (LR) and Support Vector Machine (SVM)**, each combined with **TF-IDF and Word2Vec** vectorization techniques. These models were trained on datasets with **2:1 and 3:1** toxic-to-non-toxic ratios.

- Models using **TF-IDF** outperformed those using Word2Vec, particularly in precision and recall.
- **Logistic Regression with TF-IDF** achieved the best results among classical ML models.
- SVM with Word2Vec failed to classify rare toxicity categories, producing an F1 score of 0.00 for ”severe toxic,” ”threat,” and ”identity hate.”
- The 3:1 dataset slightly improved classification performance for some toxicity categories compared to the 2:1 dataset.

3.2 Performance of Large Language Models (LLMs)

LLM-based approaches leveraged **BERT** and **DistilBERT** models, trained using the same 2:1 and 3:1 dataset balancing strategies.

- **BERT consistently outperformed DistilBERT** across all toxicity categories.
- The **bert-31-10ep** model achieved the highest classification accuracy.
- DistilBERT provided faster inference but at the cost of slightly lower classification performance.
- Training with **3:1 dataset balancing** improved recall for underrepresented toxicity categories, such as ”severe toxic” and ”threat.”

3.3 Comparative Insights

The comparison (tables 6 and 7) between classical ML approaches and LLMs highlights significant improvements when using transformer-based architectures:

- **LLMs substantially outperform classical ML models** in detecting toxic content, especially in low-frequency categories.
- Classical ML approaches struggle with multi-label classification, whereas LLMs handle overlapping toxic categories more effectively.
- Despite superior performance, BERT models require more computational resources compared to classical ML models and DistilBERT.
- **DistilBERT serves as a viable alternative** when computational efficiency is a priority, offering a balance between speed and accuracy.
- The following tables show clearly that LLMs vastly outperform the classical approach.

Models	toxic	severe toxic	obscene	threat	insult	identity hate
LR	0.64	0.34	0.70	0.28	0.63	0.38
SVM	0.62	0.25	0.67	0.31	0.63	0.45
bert (10ep)	0.82	0.64	0.91	0.79	0.89	0.81
distilbert (10ep)	0.78	0.59	0.89	0.75	0.85	0.77

Table 6: Comparison of the F1 score of all the models trained with the 2:1 ratio data set

Models	toxic	severe toxic	obscene	threat	insult	identity hate
LR	0.66	0.34	0.70	0.28	0.63	0.38
SVM	0.64	0.22	0.68	0.31	0.63	0.43
bert (10ep)	0.86	0.64	0.93	0.80	0.89	0.83
distilbert (10ep)	0.79	0.60	0.90	0.77	0.86	0.78

Table 7: Comparison of the F1 score of all the models trained with the 3:1 ratio data set

3.4 Conclusion

Given the results, LLM-based models, particularly **bert-31-10ep**, is good for real-world toxic comment classification applications.

Future enhancements can be made by opting for better dataset balancing techniques and vectorization methods. There is also room for hyperparameter optimization of the different models that have been used in the project.

We used ChatGPT[4] to write this report.

References

[1] Kaggle. Jigsaw toxic comment classification challenge, 2017. Accessed last time on 4. February 2025.

[2] Vinay Sanga. Tweets-sentiment-analysis, 2024. Accessed last time on 4. February 2025.

[3] Hugging Face. Chapter 3: Transformer models, 2023. Accessed on 4. February 2025.

[4] OpenAI. Chatgpt: A language model for dialogue, 2025.