

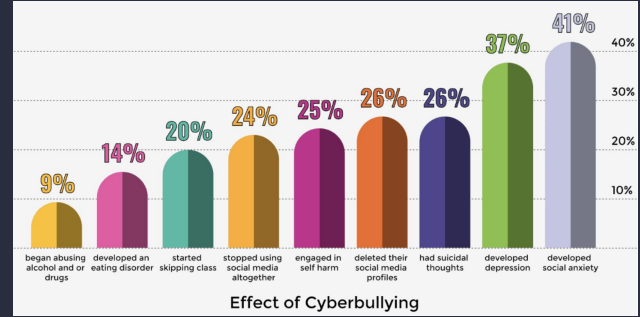
MSE 446 Final Project – Cyberbullying Comment Classification

By: Cindy Peng, Erin Lee, Florence Yuen, Keegan Liu, Nameerah Razi

GitHub: <https://github.com/k3yg4n/cyberbully-classifier>

Project Goal

- To develop a machine learning based system to detect cyberbullying that is more scalable and context-aware compared to traditional moderating methods.
- By leveraging TF-IDF for feature extraction from comments and comparing multiple classification models, the project aims to identify an efficient approach to automatically flag harmful content.



Problem Statement & Justification

As online environments, such as video games and social media grow, cyberbullying has emerged as an increasingly severe and damaging issue. A 2022 Pew Research Center report found that nearly 50% of U.S. teens have faced online harassment. Victims of online abuse often experience severe psychological impacts, and alarmingly, both the victims and perpetrators of cyberbullying are twice as likely as their peers to attempt suicide.

Cyberbullying takes many forms, including: gossip, threats and discrimination, making it difficult to detect with traditional keyword filters or manual moderation. Platforms that fail to address toxic behavior risk losing users, revenue, and public trust.

This project addressing these challenges with a machine learning approach that detects and flags online harassment in real-time. Unlike static filters, our approach adapts to context and evolving language, making it a scalable and effective tool, with the goal of promoting healthier online communities. On social media platforms, such models could automatically remove harmful comments. Similarly, in video games, these models could help impose penalties such as chat restrictions/account suspensions on players exhibiting these toxic behaviours. This intelligent moderation technique aims to ensure that younger and more vulnerable users can safely enjoy online communities.



Dataset Description

We leveraged a dataset from a **Toxic Comment Classification Challenge** on Kaggle.

- [Toxic Comment Classification Challenge | Kaggle](#)
- Contains a large number of Wikipedia comments which have been rated by humans for toxic behaviour.
- 159571 rows, 8 columns
- Each entry has a unique identifier, the comment text, and several fields that classify the comment as zero or more kinds of toxicity, such as obscene, threat, insult, or identity hate using 0 or 1.
- A single comment can be classified as 0 or more types of toxicity.

id	comment_text	toxic	severe_toxic	obscene	threat	insult	identity_hate
000099...	Explan...	0	0	0	0	0	0
000103...	D'aww!...	0	0	0	0	0	0
000113...	Hey ma...	0	0	0	0	0	0
0001b4...	"\nMor...	0	0	0	0	0	0
0001d9...	You, s...	0	0	0	0	0	0

RangeIndex: 159571 entries, 0 to 159570
Data columns (total 8 columns):

#	Column	Non-Null Count	Dtype
0	id	159571 non-null	object
1	comment_text	159571 non-null	object
2	toxic	159571 non-null	int64
3	severe_toxic	159571 non-null	int64
4	obscene	159571 non-null	int64
5	threat	159571 non-null	int64
6	insult	159571 non-null	int64
7	identity_hate	159571 non-null	int64

The data type and non-null count of each column.

```
id                                0036621e4c7e10b5
comment_text  Would you both shut up, you don't run wikipedia, especially a stupid kid.
toxic                                                1
severe_toxic                                         0
obscene                                              0
threat                                               0
insult                                              1
identity_hate                                       0
```

An example of an entry that has been labelled as toxic and an insult.

Machine Learning Approach

Preprocessing Steps

1 – Aggregate & Drop Cols

Aggregate the toxicity labels into a new *cyberbullying* column.

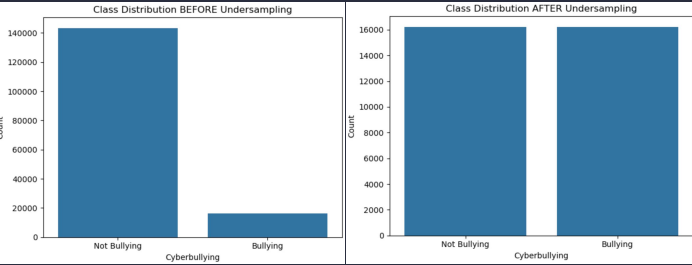
If any of toxic, severe_toxic, obscene, threat, insult, or identity_hate are 1, then *cyberbullying* will be set to 1.

Drop the irrelevant columns, namely the *id* column and the toxicity labels used in aggregation.

2 – Undersampling

~90%: Non-cyberbullying comments, ~10% Cyberbullying.

Address using undersampling to randomly drop non-cyberbullying comments until proportions are equal. Results in 16225 data points of each label.



3 – Apply TF-IDF

The product of two statistics, term frequency and inverse document frequency.

A statistical measure used in NLP to evaluate the importance of a word in a document, relative to a collection of documents (corpus).

Document: Comment, **Corpus:** Set of all comments

$$TF(t, d) = \frac{\text{Number of times term } t \text{ appears in document } d}{\text{Total number of terms in document } d}$$

$$IDF(t, D) = \log \frac{\text{Total number of documents in corpus } D}{\text{Number of documents containing term } t}$$

When using TF-IDF: we limit vocabulary size to 10,000 unique unigrams/bigrams with at least 3 characters (at least 1 letter), and ignore common words that do not carry significance (ex: "and") or appear in fewer than 2 docs. The resultant dataset has the 10,000 unigrams/bigrams as features, where the value represents the TF-IDF for that "word" in the comment.

zealand	zero	zionist	zoe	zone	zoo	zuck	zuckerberg	your	cyberbullying
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0
0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0

4 – Apply Dataset

Features: 10,000 Unigrams/Bigrams extracted from comments.

Target Variable: A single binary variable *cyberbullying*

- If *cyberbullying* = 1, else 0

Dataset trained to various ML approaches for binary classification.


Methodology:


- 80/20 train test split
- Stratified 5-fold CV


- Allows for:
- **More accurate comparisons** of model performance
 - **Minimizes overfitting** considering the **bias-variance tradeoff**


Metrics: Accuracy, Recall, Precision, F1-Score


Models Tested:

 **KNN**

 **Logistic Regression**

 **FFNN**

 **SVC**

 **Random Forest**

KNN

Pros:

- Easy to interpret
- Simple to implement (no model training required)
- Works well with nonlinear decision boundaries
- Distance weighted gives more influence to closer points

Cons:

- Relies on memorization not patterns
- Slow prediction time on large datasets
- Sensitive to noise/ bad generalization

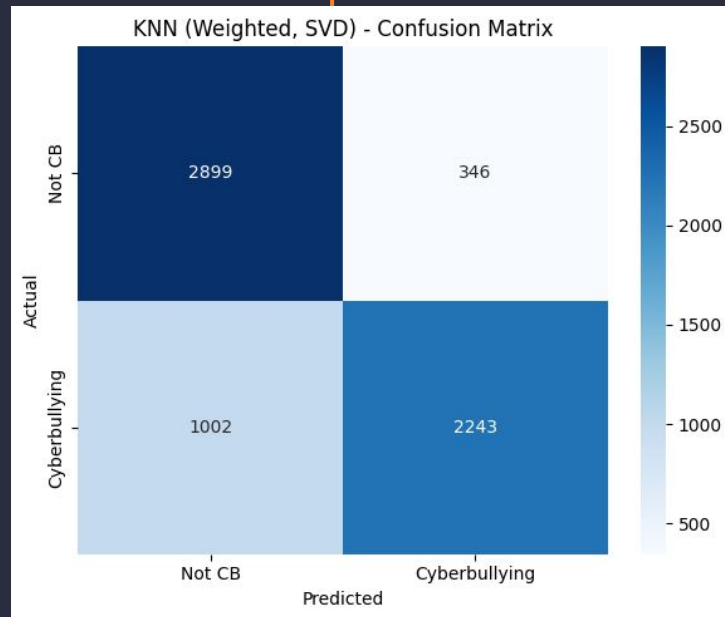
Best Params:

- $k = 12$ (tested 1 to 25)
- Weights = 'distance'
- Used TruncatedSVD ($n_components = 100$)

Performance (based on average of 5 folds)

- Overall Accuracy = 0.79
- Best F1-weighted Score = 0.78

Still struggles with false negatives on cyberbullying class



Best k: 12

Best CV F1 (macro): 0.7798

Classification Report:

	precision	recall	f1-score	support
Not CB	0.74	0.89	0.81	3245
Cyberbullying	0.87	0.69	0.77	3245
accuracy			0.79	6490
macro avg	0.80	0.79	0.79	6490
weighted avg	0.80	0.79	0.79	6490

Random Forest

Best Params:

- Max_depth = 15
- min_sample_leaf = 1
- Min_sample_splits = 2
- N_estimators = 100
- Weights = balanced

Pros:

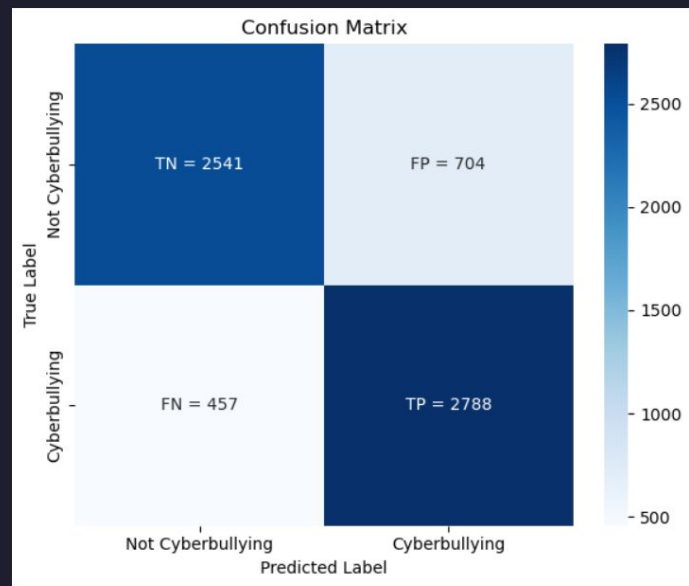
- High accuracy -> Combines multiple decision trees to reduce overfitting and improve prediction performance
- Robust to overfitting: by averaging many decision trees, the model generalizes well
- Can model complex decision boundaries unlike logistic regression

Cons:

- Slower training time, computationally intensive on large datasets
- Requires tuning of parameters n_estimators, max_depth, and min_samples_split which can be memory intensive

Performance Analysis

- High true negatives and high true positives, meaning the model is effectively identifying both classes
- Prioritizes recall for cyberbullying(86%), which is good where it's better to flag than miss harmful context
- Balanced class performance, neither class is heavily favoured



Classification Report:

	precision	recall	f1-score	support
Not Cyberbullying	0.85	0.78	0.81	3245
Cyberbullying	0.80	0.86	0.83	3245
accuracy			0.82	6490
macro avg	0.82	0.82	0.82	6490
weighted avg	0.82	0.82	0.82	6490

Accuracy: 0.8211

Mean Squared Error: 0.1789

F1 Score: 0.8277

FFNN

Hyperparameters:

```
'hidden_layer_sizes': [(64,), (128,), (64, 32)],  
'learning_rate_init': [0.001, 0.01],  
'activation': ['relu', 'logistic'],  
'max_iter': [300]
```

Pros:

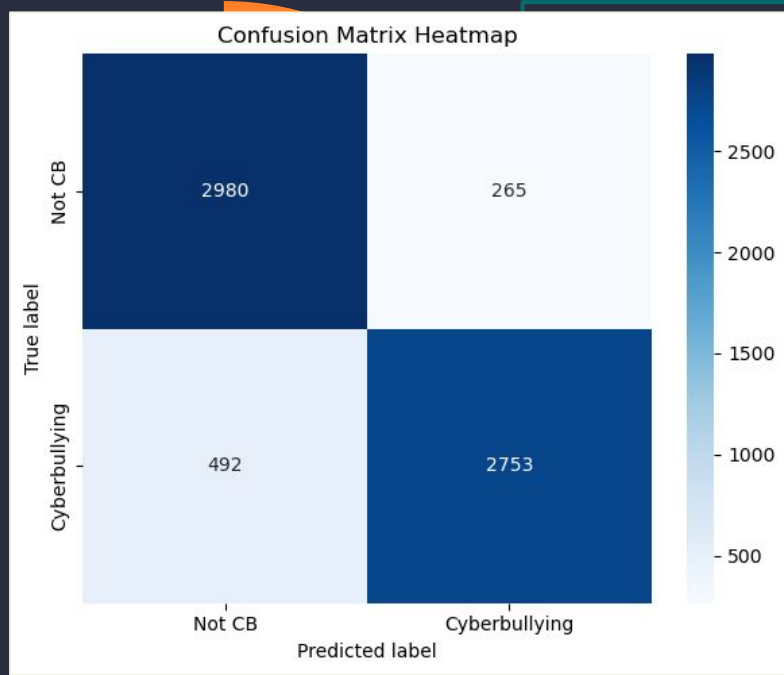
- Model complex non-linear relationships
 - Detect subtle patterns linear models may miss
- Customizable # of layers, units, activation functions
- Handle high-dimensional inputs such as TF-IDF vector

Cons:

- Underperforms on small datasets → risk of overfitting
- Longer computation times
- Less transparency
- Sensitive to hyperparameters, requires tuning

Performance analysis

- High accuracy (88%) and precision
- Balanced class support
- False negative recall is lower (85%)



Training Accuracy: 0.8860

Test Accuracy: 0.8834

Test F1 Score (macro): 0.8832

Classification Report:

	precision	recall	f1-score	support
0	0.86	0.92	0.89	3245
1	0.91	0.85	0.88	3245
accuracy			0.88	6490
macro avg	0.89	0.88	0.88	6490
weighted avg	0.89	0.88	0.88	6490

Logistic Regression

Linear classifier that finds a weight for every feature and uses a sigmoid function to turn the weighted sum into a probability

Pros:

- Training a single linear model on a TF-IDF matrix is faster than building lots of deep trees
- Easy to interpret, coefficients reveal top toxic terms

Cons:

- Linear boundary can miss sarcastic comments

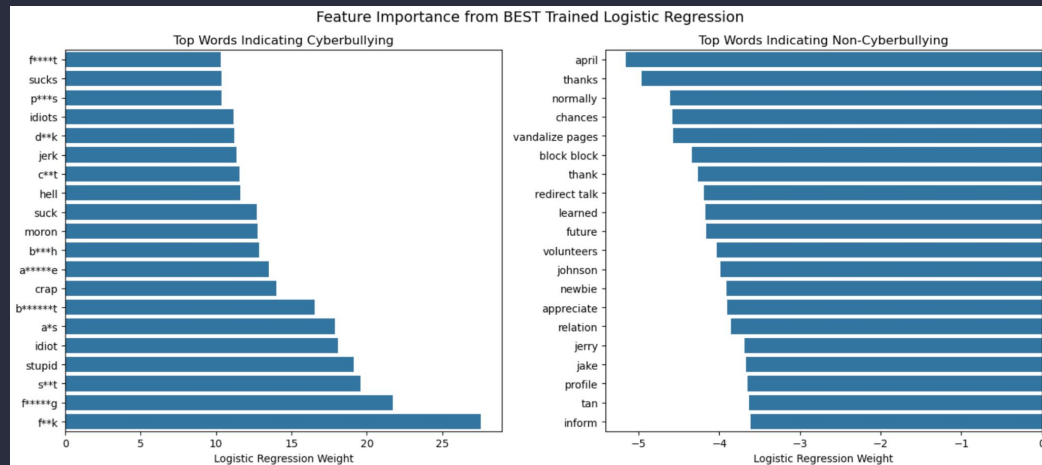
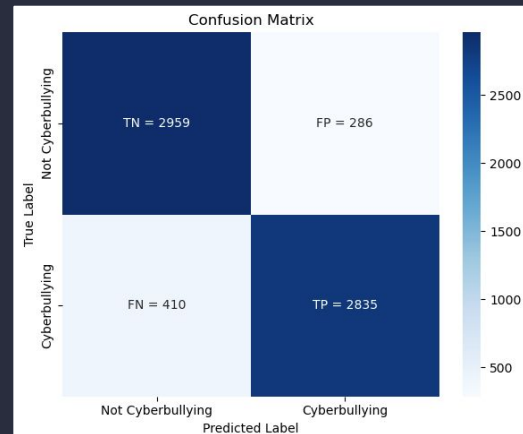
Performance Analysis:

- Accuracy: 89%
- Relatively high recall and precision for both classes.
- Misses ~13% of toxic comments (recall = 0.87)
 - Want to reduce false negatives

Classification Report:				
	precision	recall	f1-score	support
Not Cyberbullying	0.88	0.91	0.89	3245
Cyberbullying	0.91	0.87	0.89	3245
accuracy			0.89	6490
macro avg	0.89	0.89	0.89	6490
weighted avg	0.89	0.89	0.89	6490

Best parameters:

- $C = 10$
- Penalty: L2
- Higher $C \approx$ weaker regularisation



SVC (Linear SVM)

Support Vector Classifier: Finds a hyperplane that maximizes margin between classes, allowing some violations. New data points are classified based on the side of the hyperplane they fall under.

C: Parameter that controls the number and severity of the violations to the margin. Used grid search CV to determine the best value for C out of {0.01, 0.1, 1, 10, 100} based on f1-score.

Pros:

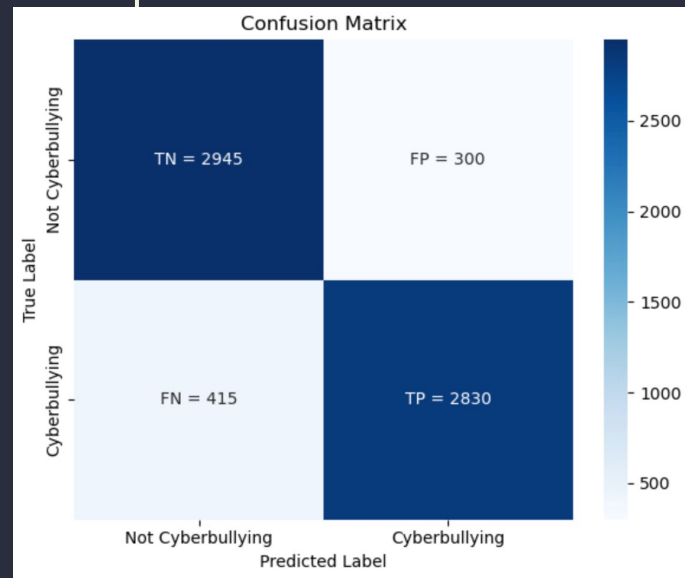
- Able to handle high-dimensional sparse data well. Our data is incredibly sparse because the feature space has an extremely large number of features (10,000).
- Robust against individual observations

Cons:

- Does not provide probabilities unlike logistic regression.
- Harder to interpret than models with transparent coefficients (like logistic regression).

Performance Analysis

- Accuracy is 89%, recall and precision are also quite high (0.87-0.91).
- Indicates strong overall predictive power, balanced class performance.
- Cyberbullying Class Precision > Recall indicates the model is cautious in avoiding FPs



Classification Report on Test Set:				
	precision	recall	f1-score	support
Not Cyberbullying	0.88	0.91	0.89	3245
Cyberbullying	0.90	0.87	0.89	3245
accuracy			0.89	6490
macro avg	0.89	0.89	0.89	6490
weighted avg	0.89	0.89	0.89	6490

Comparative Analysis

	KNN	Rand Forest	FFNN	Log Reg.	SVC
Accuracy*	0.79	0.8211	0.88	0.89	0.89
Precision*	0.79	0.825	0.885	0.895	0.89
Recall*	0.80	0.82	0.885	0.89	0.89
F1 Score*	0.78	0.82	0.885	0.89	0.89

*Note: Averaging out "not cyberbullying" and "cyberbullying classification results" equally to show 1 value.

Best Model Performance: Logistic Regression

Why we think this model performed the best:

- Logistic regression works the best in sparse feature spaces, which is what TF-IDF produces
- Strong regularization support - regularization via hyperparameter C helps prevent overfitting
- Balance bias-variance tradeoff
- Logistic regression offers best balance between performance, simplicity, and generalization for TF-IDF based text classification

Bibliography

Acharya, Nirajan. "Balancing Act: A Guide to Undersampling for Improved Machine Learning Performance." *Medium*, Medium, 6 Mar. 2024, medium.com/@nirajan.acharya777/balancing-act-a-guide-to-undersampling-for-improved-machine-learning-performance-2b3ecd08037b.

Comment, et al. "Understanding TF-IDF (Term Frequency-Inverse Document Frequency)." *GeeksforGeeks*, 7 Feb. 2025, www.geeksforgeeks.org/machine-learning/understanding-tf-idf-term-frequency-inverse-document-frequency/.

E. A. Vogels, "Teens and Cyberbullying 2022," Pew Research Center, 15 December 2022. [Online]. Available: <https://www.pewresearch.org/internet/2022/12/15/teens-and-cyberbullying-2022/>. [Accessed 3 June 2025].

Royal Canadian Mounted Police, "Just the facts: Cyberbullying," Royal Canadian Mounted Police, 1 February 2022. [Online]. Available: <https://rcmp.ca/en/gazette/just-facts-cyberbullying>. [Accessed 3 June 2025].

Staff, E. I. S. (2022, April 21). Topical issue: All the latest cyberbullying statistics and what they mean in 2022. <https://extrainningsoftball.com/topical-issue-all-the-latest-cyber-bullying-statistics-and-what-they-mean-in-2022/> [Accessed 28 June 2025].

"Toxic Comment Classification Challenge." *Kaggle*, www.kaggle.com/competitions/jigsaw-toxic-comment-classification-challenge/data. Accessed 29 July 2025.

Thank you for your time!

Any questions?