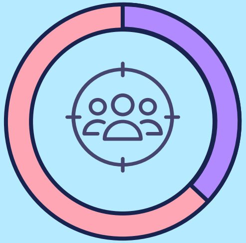
Overcoming Biases in Toxicity Models for Inclusive Conversations

Ruchi, Caroline, Nikhil

AGENDA

01

Framing the Problem



21%

Tweens (9-12 Years Old) have been **cyberbullied**, cyberbullied others, or seen cyberbullying

25%

Students indicated experiencing cyberbullying from **mean & hurtful comments**

33%

Adolescents reported **bias-based** school bullying

* Cyberbullying Research Statistics

Problem Framing

Why this project was undertaken

- **Context**: School online forums contain toxic behavior and cyberbullying.
- **Challenge**: Manual moderation process is time-consuming, inefficient, and subject to human biases.
- **Goal:** Minimize unintended biases towards certain identities with automated toxicity detection and filtering solutions like ML, and NLP.

Problem Framing

Unintended biases in machine learning models can lead to unfair treatment of certain groups or identities.

Gender Bias
The model may incorrectly label statements like
"I am a feminist and support equal rights for all"
as toxic if it associates gender-related terms with negative connotations.

Religious Bias
The model may exhibit biases against religious affiliations, wrongly classifying comments like
"I am a practicing Muslim and proud of my faith"
as toxic if associating the term "Muslim" with negative sentiments.

Problem Framing

- **Decision**: How to accurately identify and filter toxic comments in online school forums while minimizing unintended bias towards certain identities, with a goal to automate the process.
- **Decision Maker**: School administrators, forum moderators, and an AI system.
- **Value:** Improve safety and inclusivity of the online forum, enhancing the overall learning experience for students.

02

Gathering & Exploring Data

Data Source

Kaggle Dataset: Jigsaw Unintended Bias in Toxicity Classification

- **Data Shape**: 1,804,874 rows x 45 columns.
- **Variables**: comment id, comment text content, toxicity score, toxicity category scores, identity targeted scores, and sentiment scores.
  - **Target**: toxicity score
  - **Input**: comment text content, toxicity category scores, identity targeted scores
- **Data Types**: numeric, categorical/string, timestamp

Data Preparation

Filter Irrelevant Data

Remove columns that are not highly relevant to the problem statement e.g. comment_id, publication_id, parent_id, article_id

Handle Missing Values

Mainly Identity target scores columns
Replace nulls with 0
E.g. asian, atheist, bisexual, black, buddhist, female, muslim,etc

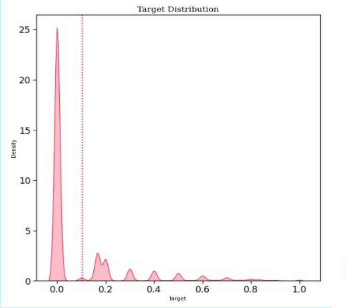
Summary Statistics & Univariate Analysis

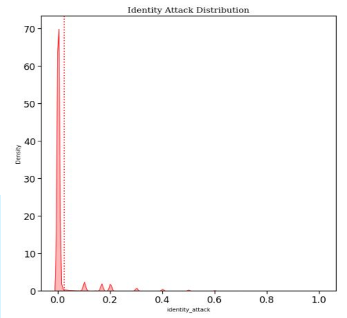
Divide dataset into **Categorical-only** & **Numerical-only datasets**
To perform summary statistics, univariate analysis

	target	severe_toxicity	obscene
count	1.804874e+06	1.804874e+06	1.804874e+06
mean	1.030173e-01	4.582099e-03	1.387721e-02
std	1.970757e-01	2.286128e-02	6.460419e-02
min	0.000000e+00	0.000000e+00	0.000000e+00
25%	0.000000e+00	0.000000e+00	0.000000e+00
50%	0.000000e+00	0.000000e+00	0.000000e+00
75%	1.666667e-01	0.000000e+00	0.000000e+00
max	1.000000e+00	1.000000e+00	1.000000e+00

	target	comment_text	created_date	rating	funny
count	1804874	1804874	1804874	1804874	1804874
unique	2913	1780823	1804362	2	61
top	0.0	Well said.	2015-10-13 18:40:35.757707+00	approved	0
freq	1264764	184	4	1684758	1549879

Summary Statistics



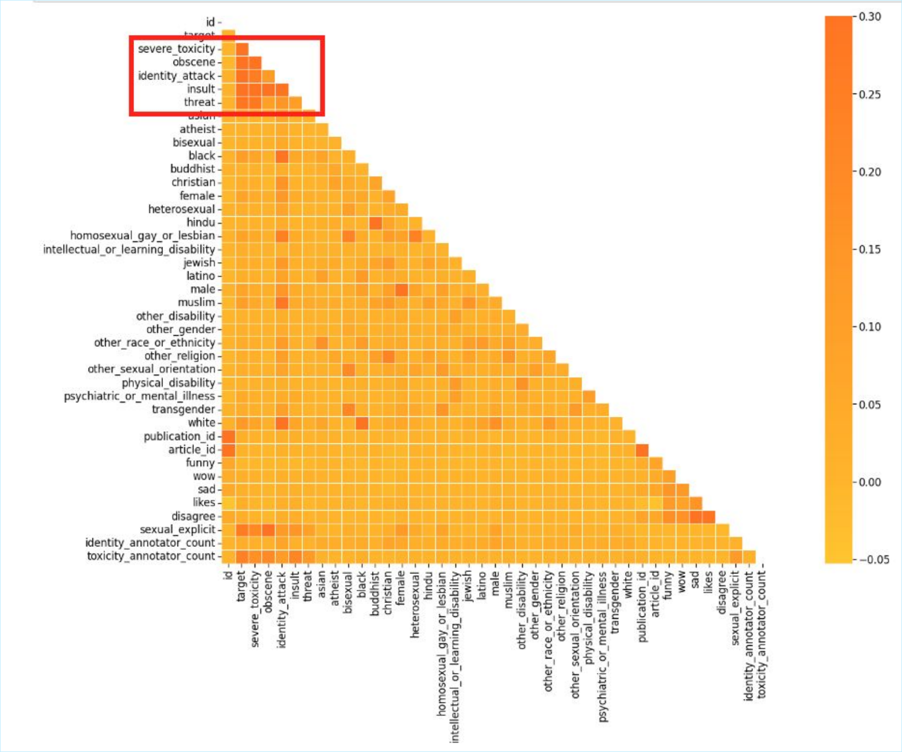


Univariate Analysis

No outliers treatment conducted: each separate value represents a certain evaluation of the toxicity for each comment text.
Tradeoff between data quality & cost: prioritize data & training quality over cost

Bivariate Analysis

Toxicity category scores are more correlated with **Toxicity target score**
→ input columns for model training



Comment Preprocess

1. Convert To Lower Case

2. Tokenize Words

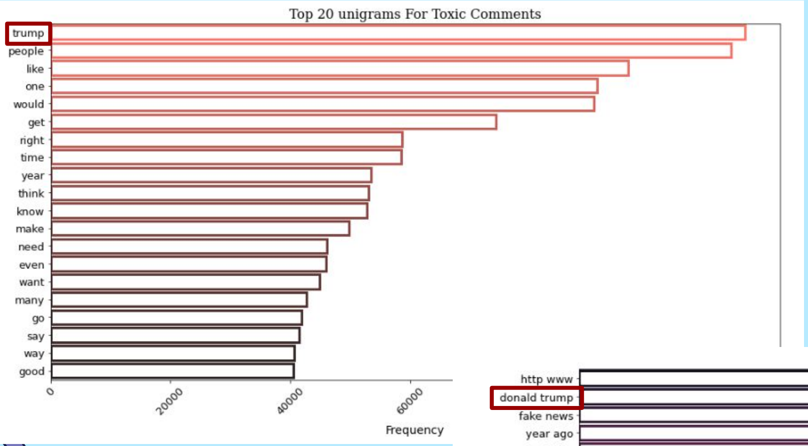
3. Remove Stopwords

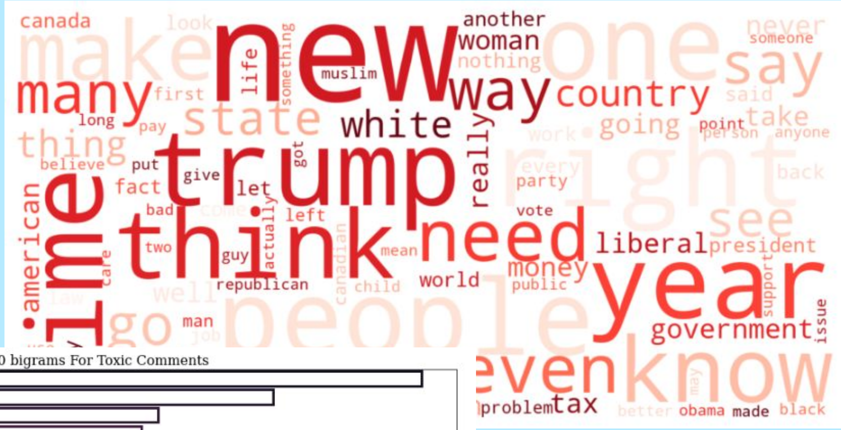
4. Lemmatization

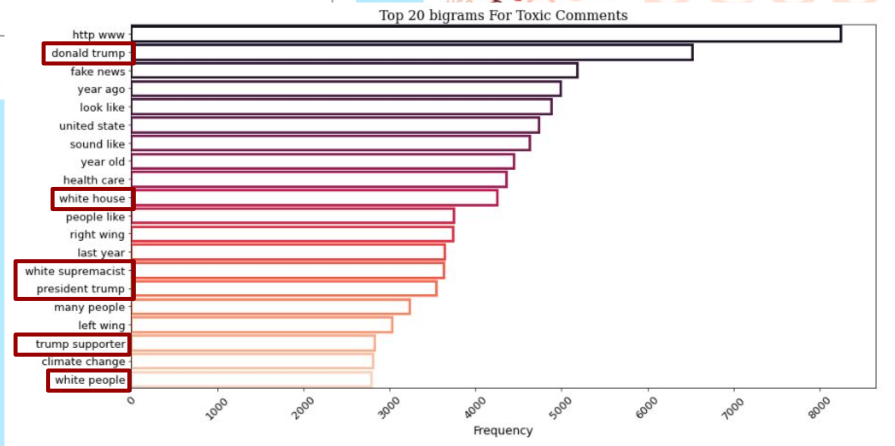
```
Before preprocessing:

 Canada is north of the USA border,  its colder in Canada.

After preprocessing:

canada north usa border colder canada
```

Word Level Comment Analysis

A lot of toxic comments are associated with **“Trump”** and **political discussions**.
→ policy or rules on political discussion in school forum







03

Modeling

Modeling

Word-Level Classification Techniques

Considered word frequencies and importance in the text, by utilizing word-level features like term frequency-inverse document frequency (TF-IDF) representations.

Context-Level Classification Techniques

Captured contextual information, taking into account not only individual words but also their relationships with surrounding words or phrases in a sentence or text.

Gave better results ✔️

Transformer Models: BERT, RoBERTa, and DistilBERT

1. Load pre-trained model and tokenizer, preprocess text-data
2. Split data into training and validation sets
3. Tokenize, convert text to input IDs, attention masks, and token_type_ids (if needed)
4. Train model
5. Evaluate trained model on validation data

Metrics

Overall AUC

Measures model's ability to **distinguish toxic and non-toxic** comments for all examples.

Bias AUCs

Assess model's **performance for specific groups** of people.

Generalized Mean of Bias AUCs

Combines **Bias AUCs for different groups**, prioritizing groups with worse performance

Final Metric

Combines **Overall AUC (25% weight) and Generalized Mean of Bias AUCs (25% weight)** to improve model performance and reduce unintended bias.

*More details in the appendix

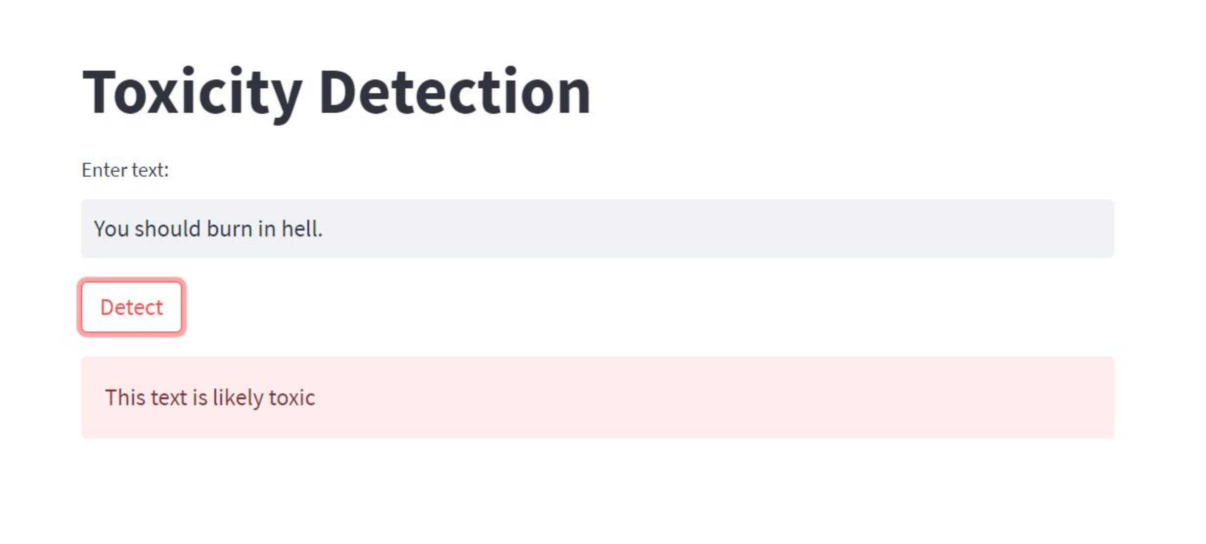
Results

	Final Metric
Logistic Regression	0.5027
Naive Bayes	0.4879
MLP	0.4975
Linear SVC	0.5015
Decision Tree	0.4478
Random Forest	0.4341
KNeighbors	0.5028
bert-base-uncased	0.9210
roberta-base	0.6723
distilbert-base-uncased	**0.9346**

```
---------- MODEL: distilbert-base-uncased ----------


---------- EPOCH: 1 ----------

AUC Score = 0.9562896310967292

---------- EPOCH: 2 ----------

AUC Score = 0.9449628169101552

---------- EPOCH: 3 ----------

AUC Score = 0.9356225016283083

---------- Model Performance: distilbert-base-uncased ----------

                      subgroup  subgroup_size  subgroup_auc  bpsn_auc  bnsp_auc
6                        black             36      0.812500  0.860606  0.950842
7                        white             64      0.842593  0.869568  0.953896
0                         male            102      0.906094  0.929049  0.945203
2       homosexual_gay_or_lesbian          36      0.909091  0.860329  0.976425
1                       female            160      0.923218  0.925913  0.960829
5                       muslim             51      0.944444  0.909846  0.969973
3                    christian            109      0.980952  0.945448  0.984158
4                       jewish             20      1.000000  0.918844  0.984448
8  psychiatric_or_mental_illness           10      1.000000  0.959649  0.981403
Final Metric: 0.9346879749831075

The best performing model is distilbert-base-uncased with a final metric of 0.9346879749831075
```

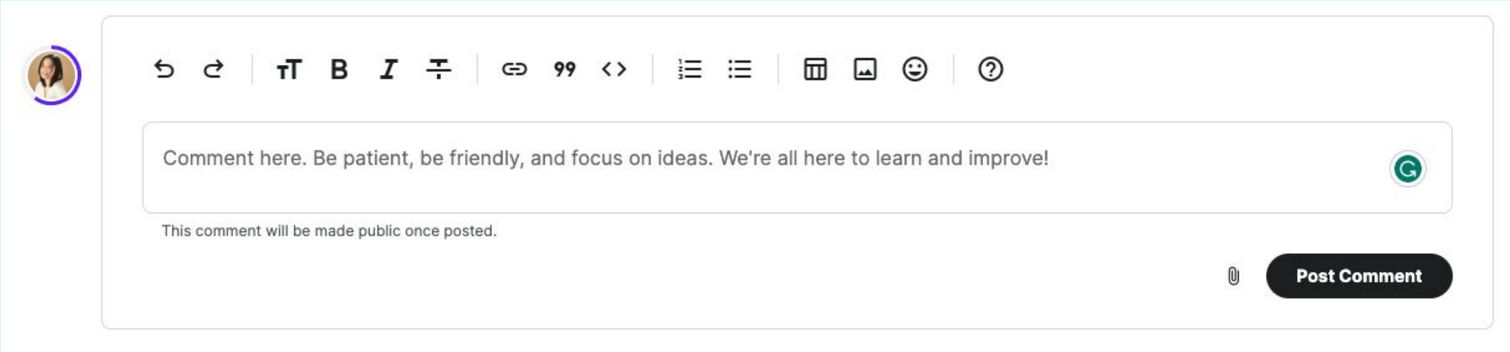
Model Deployment

Streamlit

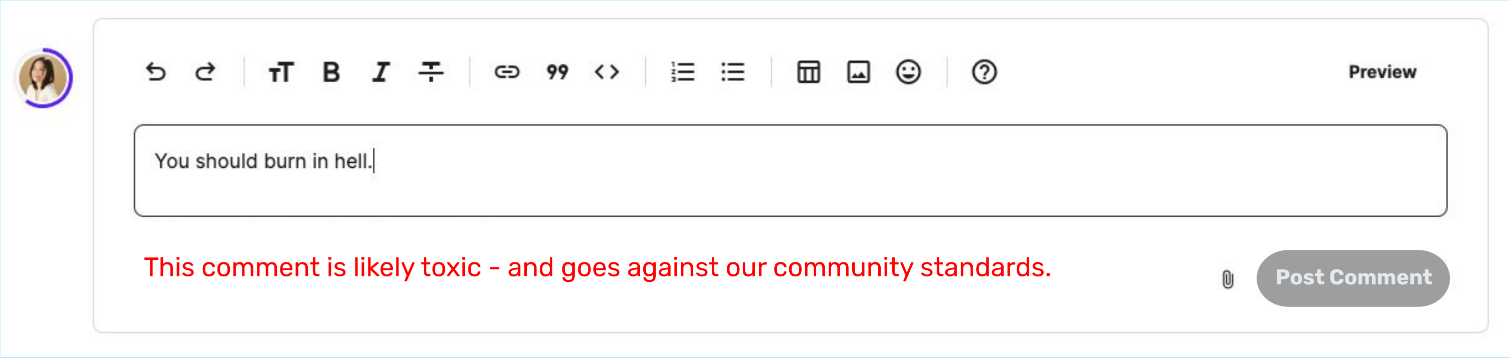


Recommendations

- Positive Placeholder text



- Connect the deployed Streamlit model to the forum backend to **automatically detect toxicity** and prevent students from posting by **disabling the button**.



05

Project Plan For Version 2

Next Steps ...

Enhance dataset

Optimize model

Broaden capabilities

Stage, Milestone, Deliverables

S 1

Data Collection and Preprocessing

Milestone: Acquire additional data & preprocess it.
Deliverable: A processed & cleaned dataset prepared for modeling.

S 2

Model Selection and Optimization

Milestone: Assess, choose, and optimize the best model based on performance metrics, additional training data, and hyperparameter tuning.
Deliverable: Performance metrics for all tested models & an optimized model with improved accuracy and minimized bias.

S 3

Model Enhancement

Milestone: Expand the model to identify other forms of online toxicity, such as hate speech, harassment, and cyberstalking.
Deliverable: A comprehensive model capable of detecting various forms of online toxicity.

Resources Required

Staffing

Data Scientist, Machine Learning Engineer / NLP Expert, Data Analyst, Data Labeler/Annotator, etc.

Computing Resources

Compute power(GPU) for training and optimizing models.
Programing tools like Python, Compiler, etc.

Budget

Budget for potential outsourcing & cloud services.
Budget for hiring staff.

Risks and Mitigation Plan

Insufficient data quality or quantity

Collect diverse and representative datasets, and ensure the data is clean, labeled correctly, and balanced before training.

Inadequate computing resources or infrastructure

Consider using cloud computing or distributed training to accelerate training;
Optimize the model architecture and hyperparameters;
Take advantage of GPU acceleration by using PyTorch.

Lack of Stakeholder Buy-In

Engage stakeholders at all stages of the project, communicate the value of the project to the school's goals, and incorporate stakeholder feedback.

THANKS!

Appendix

Metrics

Overall AUC:
This score tells us how well the model can tell the difference between toxic and non-toxic comments for all the examples.

Bias AUCs:
These scores tell us how well the model can tell the difference between toxic and non-toxic comments for specific groups of people. There are three types:
a. Subgroup AUC: This score is for comments that mention a specific group. A low score means the model struggles to tell if comments about that group are toxic or not.

b. BPSN AUC: This score is for non-toxic comments about a specific group and toxic comments not about that group. A low score means the model mistakes non-toxic comments about the group for toxic comments not about the group.

c. BNSP AUC: This score is for toxic comments about a specific group and non-toxic comments not about that group. A low score means the model mistakes toxic comments about the group for non-toxic comments not about the group.

Metrics

Generalized Mean of Bias AUCs:
This is a single score that combines all the Bias AUCs for different groups. It gives more importance to groups with worse model performance.

$$M_p(m_s) = \left(\frac{1}{N}\sum_{s=1}^{N} m_s^p\right)^{\frac{1}{p}}$$

where:

M_p = the pth power-mean function

m_s = the bias metric m calulated for subgroup s

N = number of identity subgroups

We use a p value of -5 to improve the model for the identity subgroups with the lowest model performance.

Final Metric:
This is the final score for the model. It combines the Overall AUC with the Generalized Mean of Bias AUCs. Each score is equally important (25% weight). The goal is to improve the model for everyone and reduce unintended bias.

$$score = w_0 AUC_{overall} + \sum_{a=1}^{A} w_a M_p(m_{s,a})$$

where:

A = number of submetrics (3)

$m_{s,a}$ = bias metric for identity subgroup s using submetric a

w_a = a weighting for the relative importance of each submetric; all four w values set to 0.25