# Tweets Toxicity Analysis

**GROUP 1**

Bombino Biancamaria,

Fabbri Lucia,

Mastrorilli Alessandro,

Ricci Davide,

Sarbach-Pulicani Vincent

# INTRODUCTION

## GOAL

Analyze all the tweets in order to:

- discover if a tweet is considered to be 'toxic' or 'non toxic'
- Highlight different categories of toxicity that recur most frequently in toxic tweets

## WORKFLOW

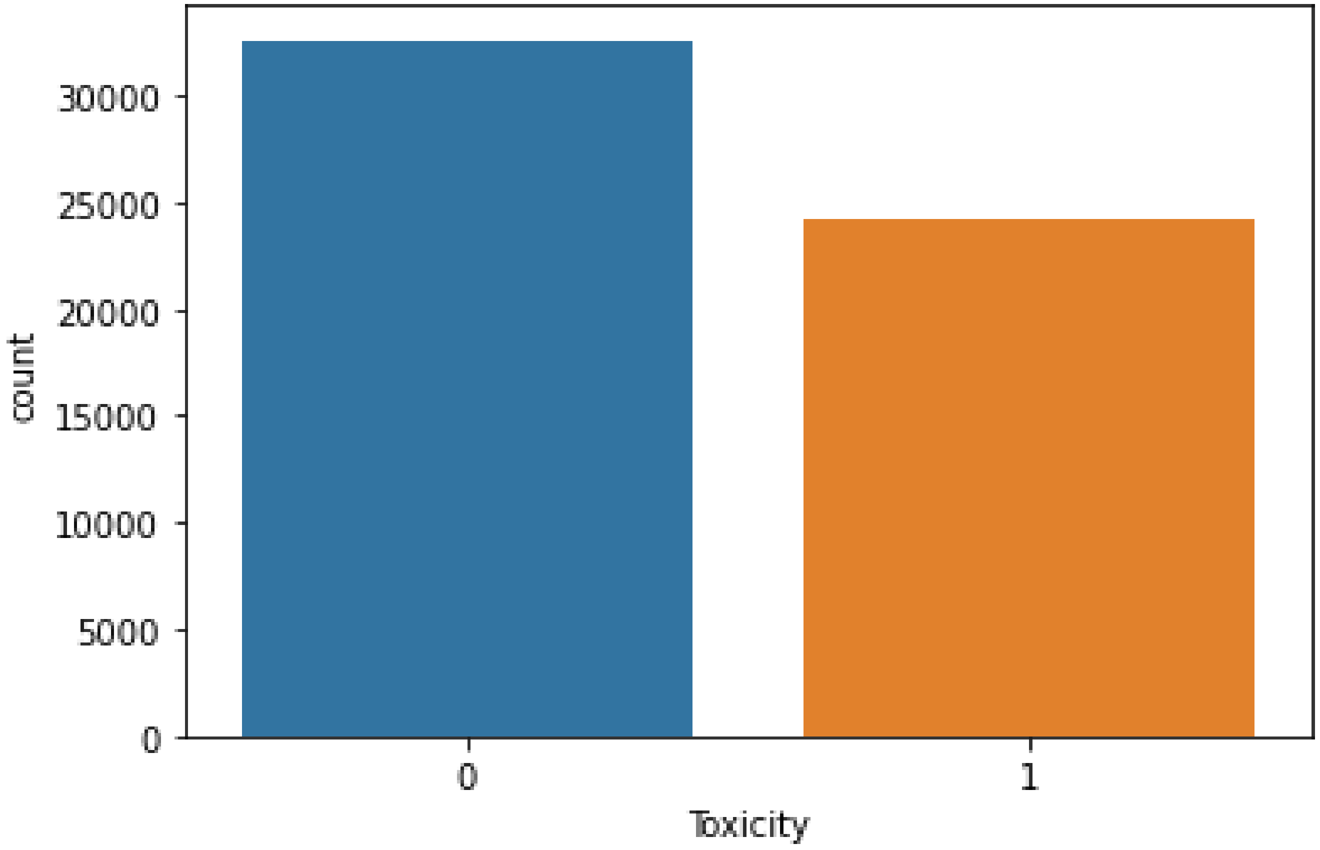
1. Data Understanding & Preparation
2. Topic Modeling
3. Simple Classifiers
4. Neural Networks
5. BERT
6. Advanced Topics

# DATA UNDERSTANDING

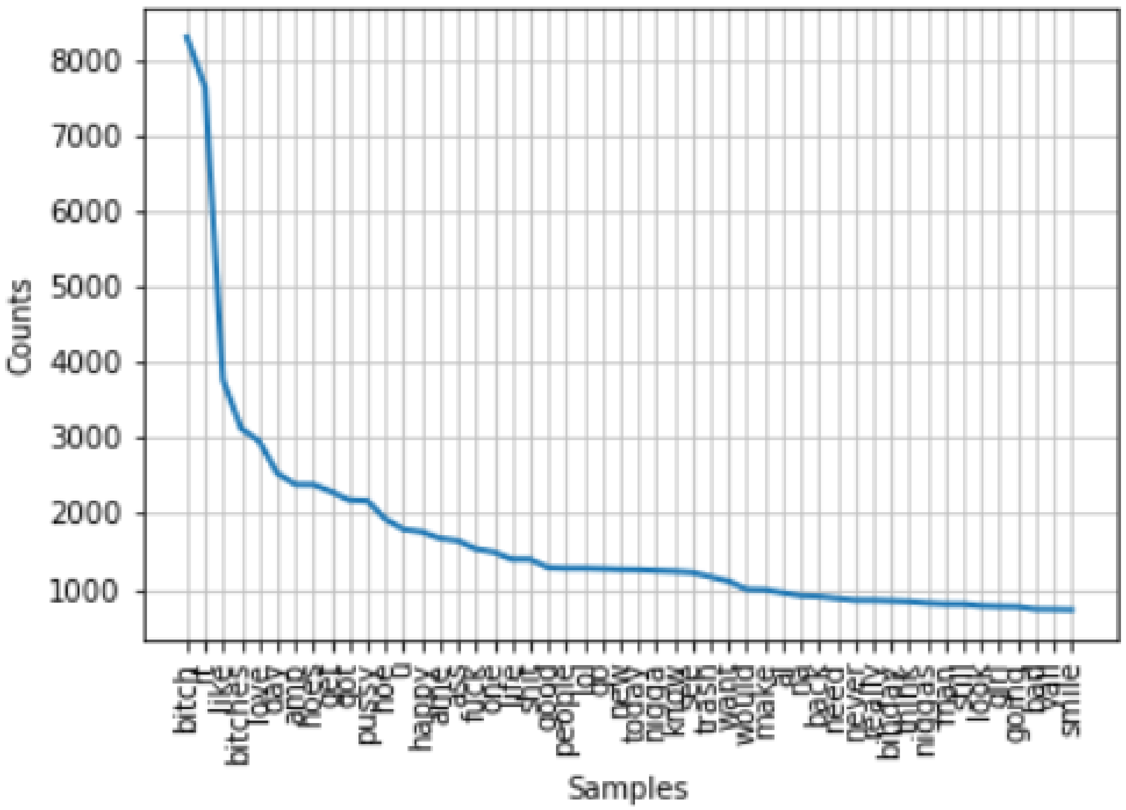
*Toxic Tweet Dataset* describes a collection of tweets

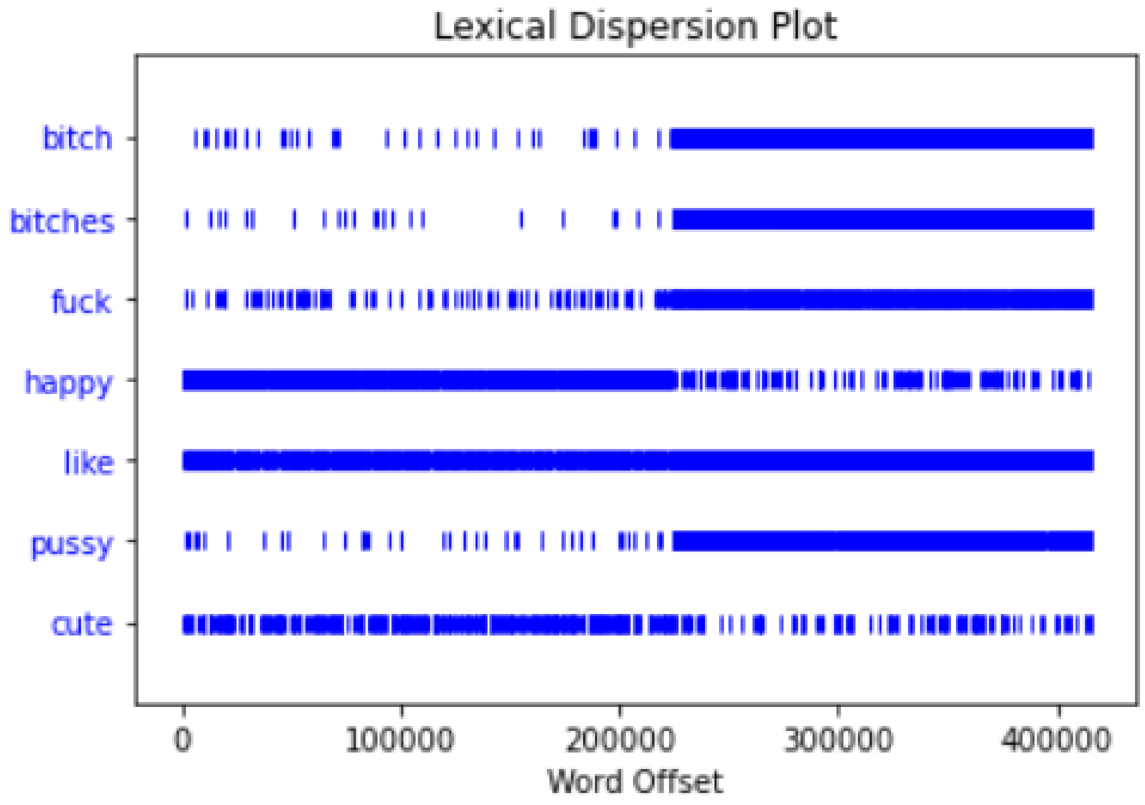
- **Toxicity**: 0 for not toxic tweet and 1 for toxic tweet
- **Tweet**: short sentences that describe the text posted by users

Balanced dataset containing
32592 **non-toxic** tweets
and 24153 **toxic** tweets.



# FREQUENCY COUNTING & LEXICAL DISPERSION



# DATA PREPARATION

DATASET

- 54313 records
- Tweets are very short messages

CLEANING

- Remove emoticon, punctuation, links, mentions, numbers, UTF8/ASCII characters, special characters
- Remove Stopwords

CONVERT TO LOWER CASE

- Strings where all characters are lower case.
- Substitution of words from the contracted form to the extended form

TOKENIZATION & LEMMATIZATION

- On the cleaned dataset

NEW DATASET

- It is used for subsequent analyses

# TOPIC MODELING
## *Overview*

- The objective is to determine *classes* or *types* of toxicity inherent in our dataset.
- Perform a **topic modeling** analysis can help to achieve this goal, a probabilistic approach based on a document-term matrix.
- The evaluation method is mainly **empirical** even if the **topic coherence** exists ⟶ if the problem of *self-evaluation* is time and non-reproducibility, this is more than enough in our study where the TM is not the main component

# TOPIC MODELING
## *Dataset preparation*

| Topic 1 | Topic 2 | Topic 3 | Topic 4 | Topic 5 | Topic 6 |
|---|---|---|---|---|---|
| bitch | bitches | not | fucking | stop | talking |
| rt | niggas | pussy | love | take | niggah |
| am | trash | do | lmao | ghetto | nigger |
| like | good | can | dick | gone | yeah |
| hoes | really | want | cunt | ho | trust |
| hoe | people | ya | cuz | play | stay |
| ass | see | ame | shes | sometg | find |
| get | white | even | tonight | ama | nobody |
| shit | wanna | ill | hope | looking | haha |
| nigga | damn | fat | watch | ah | faggots |

- On the first step, the idea was to choose a subset of dataset between the simple *cleaned* corpus and the *lemmatized* one
- As an unsupervised machine learning technique, topic modeling is about an **empirical interpretation of the results**
- The choice was made for the *cleaned* subset, to conserve the **semantic wealth** of the corpus

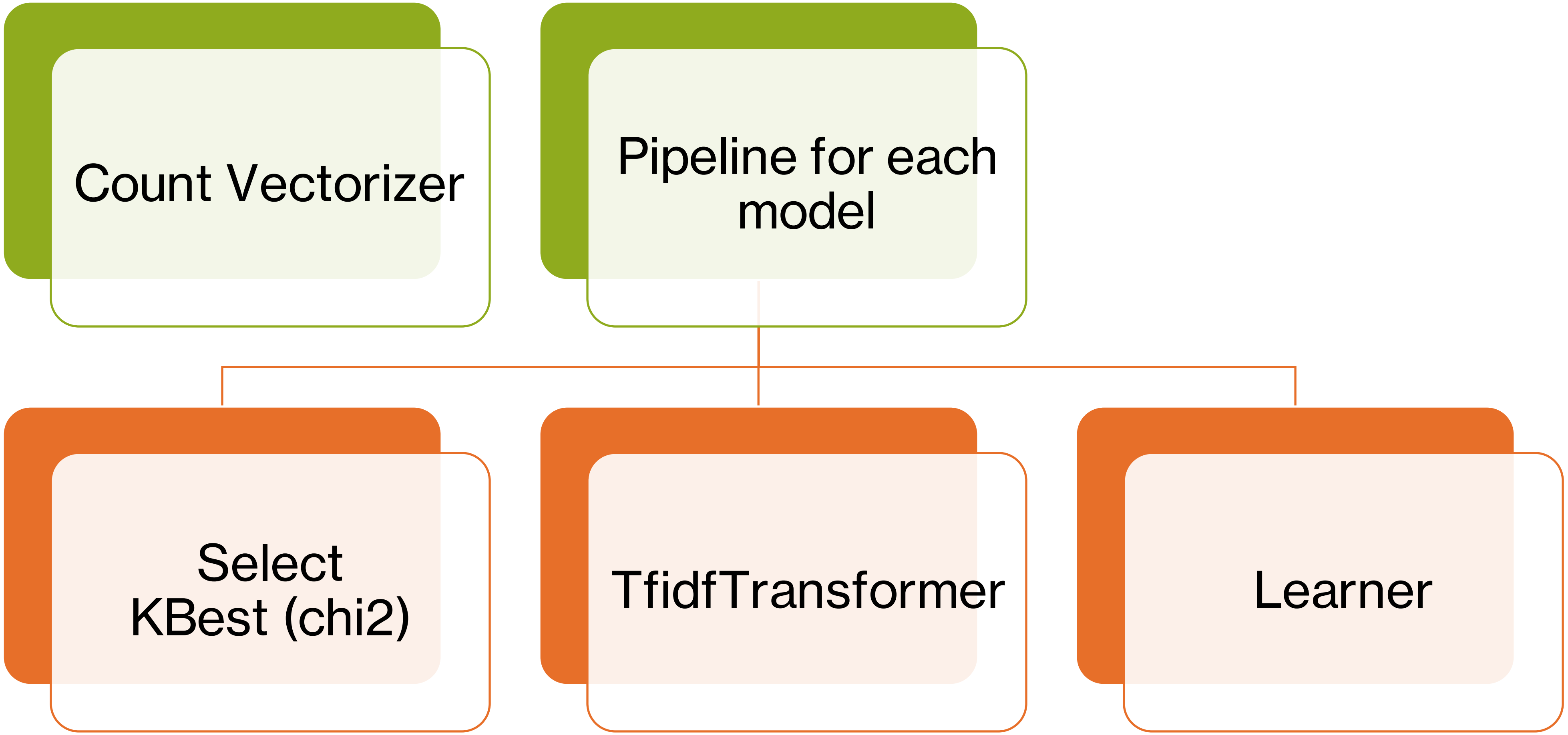
# TOPIC MODELING
## *Results*


- Toxicity types
  - **Cyberbullying**
  - **Racism**
  - **Misogyny**

- These detection doesn't mean that other types aren't present, it means that those types are more present in this *specific* dataset.

**Improvements →**

# SUPERVISED LEARNING



# SIMPLE CLASSIFIERS (SC)

| Naive Bayes | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,92 | 0,89 | 0,90 | 8713 |
| Class 1 | 0,87 | 0,90 | 0,88 | 6940 |
| macro avg | 0,89 | 0,90 | 0,89 | 15653 |

| SVM | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,91 | 0,96 | 0,93 | 8713 |
| Class 1 | 0,95 | 0,88 | 0,91 | 6940 |
| macro avg | 0,93 | 0,92 | 0,92 | 15653 |

| KNN | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,86 | 0,97 | 0,91 | 8713 |
| Class 1 | 0,95 | 0,81 | 0,87 | 6940 |
| macro avg | 0,91 | 0,89 | 0,89 | 15653 |

| Decision Tree | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,89 | 0,97 | 0,93 | 8713 |
| Class 1 | 0,95 | 0,86 | 0,90 | 6940 |
| macro avg | 0,92 | 0,91 | 0,92 | 15653 |

# RESULTS

Improvements →

- Optimization with **GridSearchCV** e **RandomizedSearchCV** (*n_repetion = 200* and *n_split=5)*
- It is possible to notice how **LinearSVC** and **Decision Tree** have increased the *F1-Score*

**Conclusions**
In both cases, the results obtained were very good, most likely dictated by the balancy of the initial dataset.

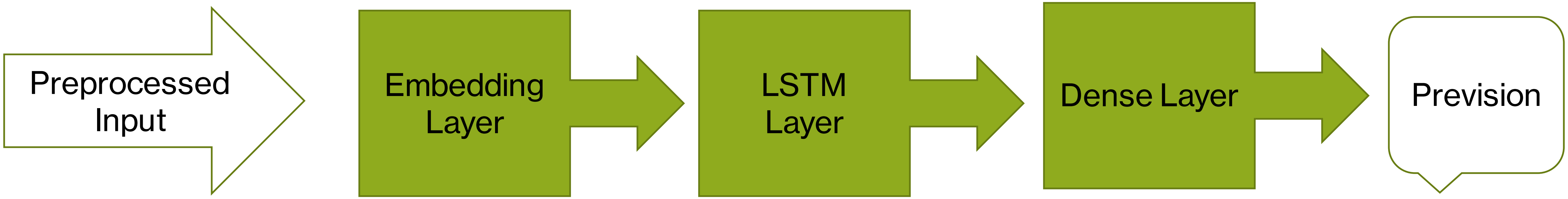
| SVC | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 (Non Tossico) | 0,91 | 0,96 | 0,94 | 8713 |
| Class 1 (Tossico) | 0,95 | 0,88 | 0,91 | 6940 |
| macro avg | 0,93 | 0,92 | 0,93 | 15653 |

| KNN | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,87 | 0,95 | 0,91 | 8713 |
| Class 1 | 0,93 | 0,82 | 0,88 | 6940 |
| macro avg | 0,90 | 0,89 | 0,89 | 15653 |

| Naive Bayes | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,92 | 0,89 | 0,90 | 8713 |
| Class 1 | 0,87 | 0,90 | 0,88 | 6940 |
| macro avg | 0,89 | 0,90 | 0,89 | 15653 |

| Decision Tree | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 | 0,90 | 0,96 | 0,93 | 8713 |
| Class 1 | 0,94 | 0,87 | 0,90 | 6940 |
| macro avg | 0,92 | 0,91 | 0,92 | 15653 |

# NEURAL NETWORK CLASSIFIERS (NNC)

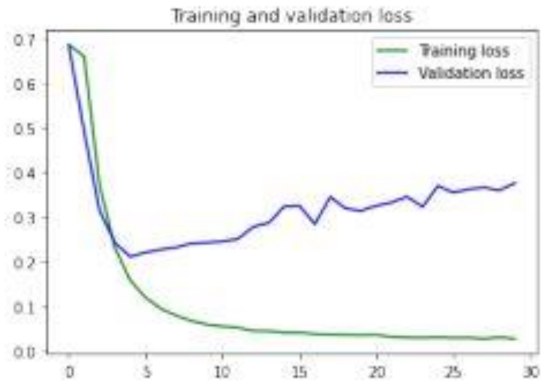



- Sequential and recurrent structure
- Binary crossentropy loss function
- Adam Optimizer for back-propagation, if enabled

# 2 TYPES OF MODELS

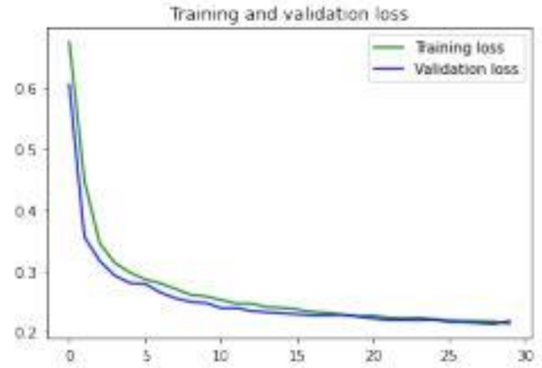
Improvements →

Randomic Embedding Weight Matrix




| | PRECISION | RECALL | F1-SCORE |
|---|---|---|---|
| CLASS 0 | 0.92 | 0.93 | 0.93 |
| CLASS 1 | 0.91 | 0.90 | 0.91 |
| Macro-avg | 0.92 | 0.92 | 0.92 |

Pre-Trained Embedding GloVe Weight Matrix




| | PRECISION | RECALL | F1-SCORE |
|---|---|---|---|
| CLASS 0 | 0.91 | 0.94 | 0.93 |
| CLASS 1 | 0.92 | 0.89 | 0.90 |
| Macro-avg | 0.92 | 0.92 | 0.92 |

# BERT: BINARY CLASSIFICATION

- Bidirectional machine learning model pre-trained in raw text only:
  - It uses the transformer mechanism of attention to learn the dependences between words by reading simultaneously the entire sequence of input and in a bidirectional way.

- Innovative learning strategies:
  - *Masked language modeling (MLM)*: taking a sentence, the model randomly masks 15% of the words in the input
    -> predict the original value of the masked word, according to the context given by other terms.
  - *Next sentence prediction (NSP)*: the model receives several pairs of input sentences
    -> predict whether the second sentence follows the first (50% of the inputs are sequential).

# BERT PRE-PROCESSING

**Model**: *BertForSequenceClassification* composed by 12 encoder layer with an bert-base-uncased vocabulary.

**Input**: preprocessed data, but need other transformations in order to be applyed.

**Output:** add a linear classification layer on top of the model to map the final states of BERT into the target labels.

Tokenization of sentences

Insertion of special tokens [CLS] and [SEP]

Set a MAX_LEN for sentences

Word distribution has been plotted in order to get the best lenght for tokens: 25

Conversion of tokens into IDs of the BERT vocabulary

Add 0 value in order to reach the same lenght (25)

Creation of attention masks to separate real tokens (1) from the padding tokens (0)

# BERT MAIN STEPS

**Initialization**

- Convert our data to tensors, which are the input format for the model
- Creation of the iterator DataLoader for training, validation and test sets
- Call the pre-trained BERT model: *BertForSequenceClassification*
- Setting of additional hyperparameters and grabbing training parameters from the pretrained model

**Training and Validation**

- For each epoch=2, we have a function train() that iterates over the batch_size= 32 and then we immeditaly evaluate the model through a function evaluate()

# BINARY BERT EVALUATION & RESULTS

| BINARY BERT | Precision | Recall | F1-Score | Support |
|---|---|---|---|---|
| Class 0 (Non Tossico) | 0,94 | 0,96 | 0,95 | 8734 |
| Class 1 (Tossico) | 0,94 | 0,93 | 0,94 | 6919 |
| macro avg | 0,94 | 0,94 | 0,94 | 15653 |

**Conclusions**

- Binary BERT on the target variable *Toxicity* gives the best result if compared with previous classifiers

Improvements →

# EMOTION DETECTION

## Goal

- Associating emotions with each tweets
- Granularity: the scale or level of detail in a set of data => "Sentence Level"

## Trasfer learning

- Pre-trained language model and a dataset labeled for the task

## EmoRoBERTa

- Pre-trained model
- It is a variant of RoBERTa.
  - RoBERTa is trained on an order of data magnitude more than BERT, for a longer period of time. This allows RoBERTa representations to generalize better at downstream activities than BERT
- It is trained on labeled text data with emotions, so it has both information about language and context and information about emotions
- EmoRoBERTa's input is a text string representing a sentence or document, and the output is a prediction of the emotions associated with that text. These predictions can be in classification form (for example, "happy" or "sad") or in probability form for each emotional class.

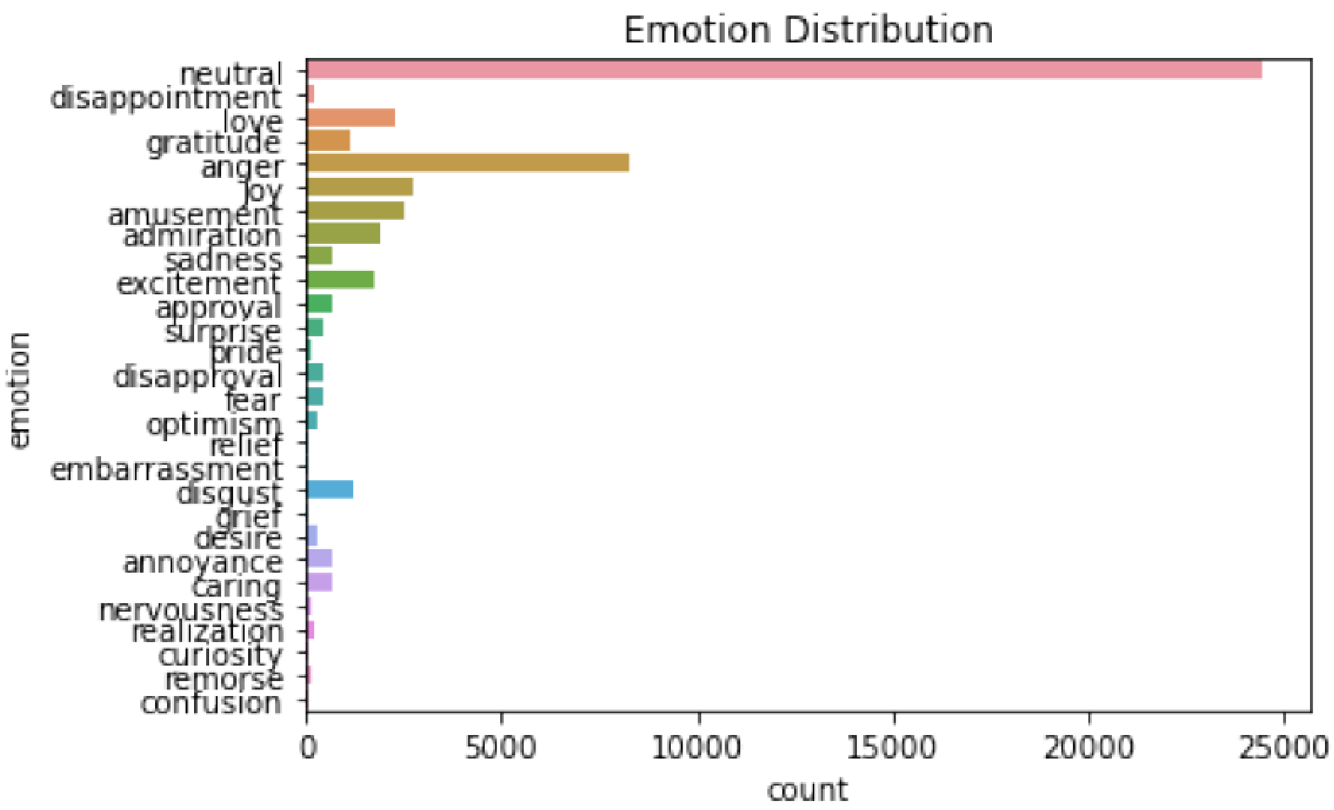
## GoEmotions

- Dataset labeled
- 58000 Reddit comments with 28 emotions
- more detailed taxonomy (classification of sequences and their possible combinations)

# EMOTION DETECTION RESULTS

- About 50 percent of the records are characterized by a **neutral** emotion. It is related to 9320 toxic tweets and 15106 non-toxic tweets.

- The second recurrent emotion is **anger**, an expected result given by the large amount of data labeled as toxic comments.
- Examples of anger tweets:
  - *"look like stop talk fuck bitch"*
  - *"youre retard hope get type diabetes die sugar"*



| | PRECISION | RECALL | F1-SCORE | SUPPORT |
|---|---|---|---|---|
| **admiration** | 0.60 | 0.70 | 0.64 | 571 |
| **amusement** | 0.82 | 0.86 | 0.84 | 763 |
| **anger** | 0.74 | 0.87 | 0.80 | 2490 |
| **annoyance** | 0.50 | 0.00 | 0.01 | 201 |
| **approval** | 0.45 | 0.02 | 0.04 | 214 |
| **caring** | 0.51 | 0.24 | 0.32 | 198 |
| **desire** | 0.36 | 0.32 | 0.34 | 96 |
| **disappointment** | 0.80 | 0.06 | 0.11 | 65 |
| **disapproval** | 0.72 | 0.09 | 0.16 | 141 |
| **disgust** | 0.50 | 0.53 | 0.51 | 364 |
| **excitement** | 0.66 | 0.52 | 0.58 | 525 |
| **fear** | 0.25 | 0.20 | 0.22 | 140 |
| **gratitude** | 0.81 | 0.86 | 0.84 | 336 |
| **joy** | 0.67 | 0.78 | 0.72 | 828 |
| **love** | 0.76 | 0.83 | 0.80 | 687 |
| **neutral** | 0.83 | 0.86 | 0.84 | 7328 |
| **optimism** | 0.86 | 0.07 | 0.13 | 84 |
| **pride** | 1.00 | 0.03 | 0.05 | 38 |
| **realization** | 1.00 | 0.07 | 0.12 | 60 |
| **remorse** | 0.81 | 0.36 | 0.50 | 36 |
| **sadness** | 0.44 | 0.67 | 0.53 | 214 |
| **surprise** | 0.56 | 0.57 | 0.57 | 131 |
| Macro-avg | 0.52 | 0.34 | 0.35 | 15653 |

# BERT MULTICLASS

- After searching for the emotions associated with each tweet and adding this information to the dataset, the **multiclass BERT** was run with this nonbinary target variable.
- The ***data pre-processing*** and ***methods*** used in this model are the same as those described for the binary BERT.
- Only those emotions that reported an f1_score value different from zero are shown in the table. In fact, having a value of f1_score equal to zero means that some labels in the test set probably do not appear among those predicted.
- The **results** obtained are not as satisfactory as those obtained for the toxicity variable because of the unbalanced nature of the dataset.

# FUTURE IMPROVEMENTS

### Topic modeling

- Further analysis with pyLDAvis and the *relevance* parameter (difficult to show)
- Better preparation, deleting more stopwords or by improving the parameters for the model
- Maybe consolidate the evaluation of the model with a better usage of the *topic coherence*

### Simple Classifiers

- Usage of other models (Random Forest)
- Usage of OvO and OvR for emotions classification

### NN Classifiers

- Test different configurations of hyperparameters, layers, units
- Extraction and <u>selection</u> of meaningful feature
- Usage of others pre-trained word embedding matrix (word2Vec, FastText, doc2Vec)
- CNN

### BERT

- Try other parameters configurations: modify model dimensions and add new layers to improve classification accuracy (computational costs need to be considered)
- FastText

### Emotion Detection

- NRC Emotion Lexicon: English word list and theirs associations with eight basic emotions (anger, fear, anticipation, trust, surprise, sadness, joy and disgust) and two feelings (negative and positive).