Project on
Malignant Comments Classifier

Presented by
Amar Kumar

Problem Statement

The proliferation of social media enables people to express their opinions widely online. However, at the same time, this has resulted in the emergence of conflict and hate, making online environments uninviting for users. Although researchers have found that hate is a problem across multiple platforms, there is a lack of models for online hate detection.

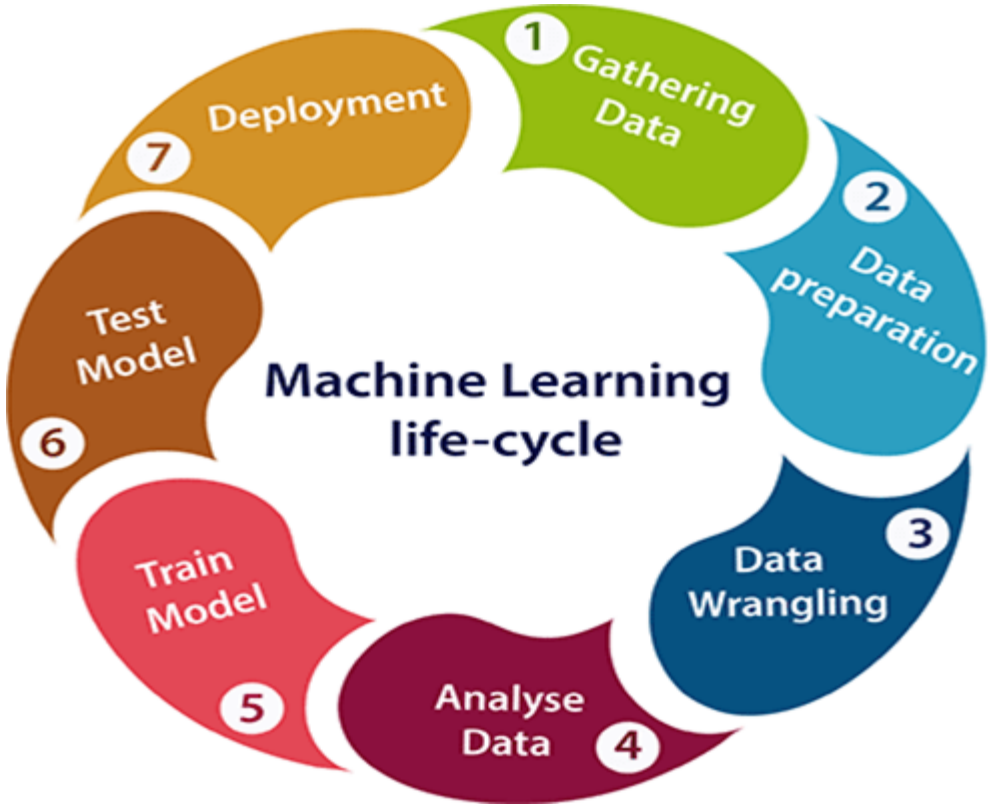
Online hate, described as abusive language, aggression, cyberbullying, hatefulness and many others has been identified as a major threat on online social media platforms. Social media platforms are the most prominent grounds for such toxic behaviour.

There has been a remarkable increase in the cases of cyberbullying and trolls on various social media platforms. Many celebrities and influences are facing backlashes from people and have to come across hateful and offensive comments. This can take a toll on anyone and affect them mentally leading to depression, mental illness, self-hatred and suicidal thoughts.

Internet comments are bastions of hatred and vitriol. While online anonymity has provided a new outlet for aggression and hate speech, machine learning can be used to fight it. The problem we sought to solve was the tagging of internet comments that are aggressive towards other users. This means that insults to third parties such as celebrities will be tagged as unoffensive, but “u are an idiot” is clearly offensive.

Steps Involved in Building Machine Learning Project

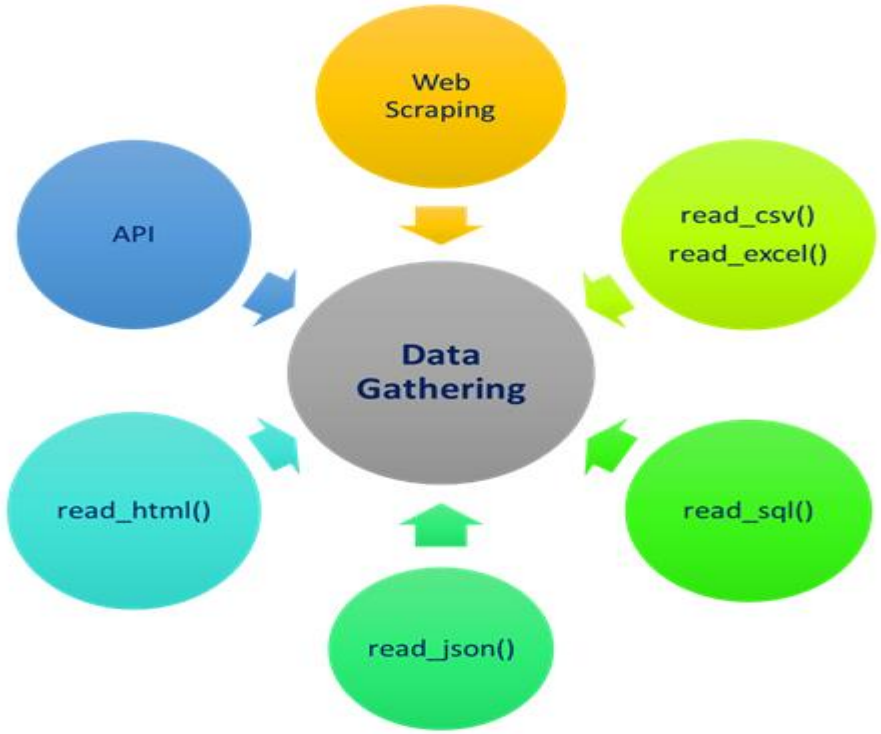
- Gathering Data
- Data Preparation
- Data Wrangling
- Analyze Data
- Train the model
- Test the model
- Deployment



Gathering Data

In this step, we need to identify the different data sources, as data can be collected from various sources such as files, database, internet, or mobile devices.

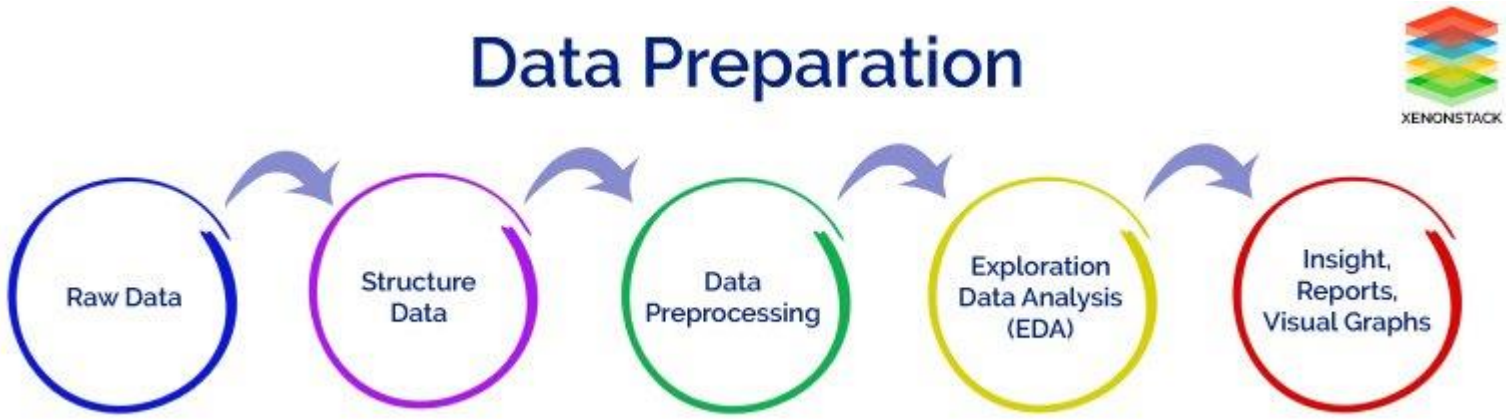
In our project, we have collected the data from files through available resources and stored in our local storage.



Data Preparation

After collecting the data, we need to prepare it for further steps. Data preparation is a step where we put our data into a suitable place and prepare it to use in our machine learning training.
This step can be further divided into two processes:

- **Data exploration:**
  It is used to understand the nature of data that we have to work with. We need to understand the characteristics, format, and quality of data.
  A better understanding of data leads to an effective outcome. In this, we find Correlations, general trends, and outliers.
- **Data pre-processing:**
  Now the next step is preprocessing of data for its analysis.



Data wrangling

Data wrangling is the process of cleaning and converting raw data into a useable format. It is the process of cleaning the data, selecting the variable to use, and transforming the data in a proper format to make it more suitable for analysis in the next step. It is one of the most important steps of the complete process. Cleaning of data is required to address the quality issues.

It is not necessary that data we have collected is always of our use as some of the data may not be useful. In real-world applications, collected data may have various issues, including:

- Missing Values
- Duplicate data
- Invalid data
- Noise

So, we use various filtering techniques to clean the data.
It is mandatory to detect and remove the above issues because it can negatively affect the quality of the outcome.

Analyze Data

Now the cleaned and prepared data is passed on to the analysis step. This step involves:

- Selection of analytical techniques
- Building models
- Review the result

The aim of this step is to build a machine learning model to analyze the data using various analytical techniques and review the outcome. It starts with the determination of the type of the problems, where we select the machine learning techniques such as Classification, Regression, Cluster analysis, Association, etc. then build the model using prepared data, and evaluate the model.

In our project, we used principal component analysis to reduce dimensions in our dataset.

Natural Language Processing (NLP)

Natural Language Processing (NLP) is a subfield of artificial intelligence (AI). It helps machines process and understand the human language so that they can automatically perform repetitive tasks. Examples include machine translation, summarization, ticket classification, and spell check.

One of the main reasons natural language processing is so critical to businesses is that it can be used to analyse large volumes of text data, like social media comments, customer support tickets, online reviews, news reports, and more.

All this business data contains a wealth of valuable insights, and NLP can quickly help businesses discover what those insights are.

It does this by helping machines make sense of human language in a faster, more accurate, and more consistent way than human agents.

NLP tools process data in real time, 24/7, and apply the same criteria to all your data, so you can ensure the results you receive are accurate – and not riddled with inconsistencies.

Once NLP tools can understand what a piece of text is about, and even measure things like sentiment, businesses can start to prioritize and organize their data in a way that suits their needs.

Challenges of NLP

While there are many challenges in natural language processing, the benefits of NLP for businesses are huge making NLP a worthwhile investment.
However, it's important to know what those challenges are before getting started with NLP.
Human language is complex, ambiguous, disorganized, and diverse. There are more than 6,500 languages in the world, all of them with their own syntactic and semantic rules.
Even humans struggle to make sense of language.
So for machines to understand natural language, it first needs to be transformed into something that they can interpret.
In NLP, syntax and semantic analysis are key to understanding the grammatical structure of a text and identifying how words relate to each other in a given context. But, transforming text into something machines can process is complicated.
Data scientists need to teach NLP tools to look beyond definitions and word order, to understand context, word ambiguities, and other complex concepts connected to human language.

How Does Natural Language Processing Work?

In natural language processing, human language is separated into fragments so that the grammatical structure of sentences and the meaning of words can be analysed and understood in context. This helps computers read and understand spoken or written text in the same way as humans.
Here are a few fundamental NLP pre-processing tasks data scientists need to perform before NLP tools can make sense of human language:

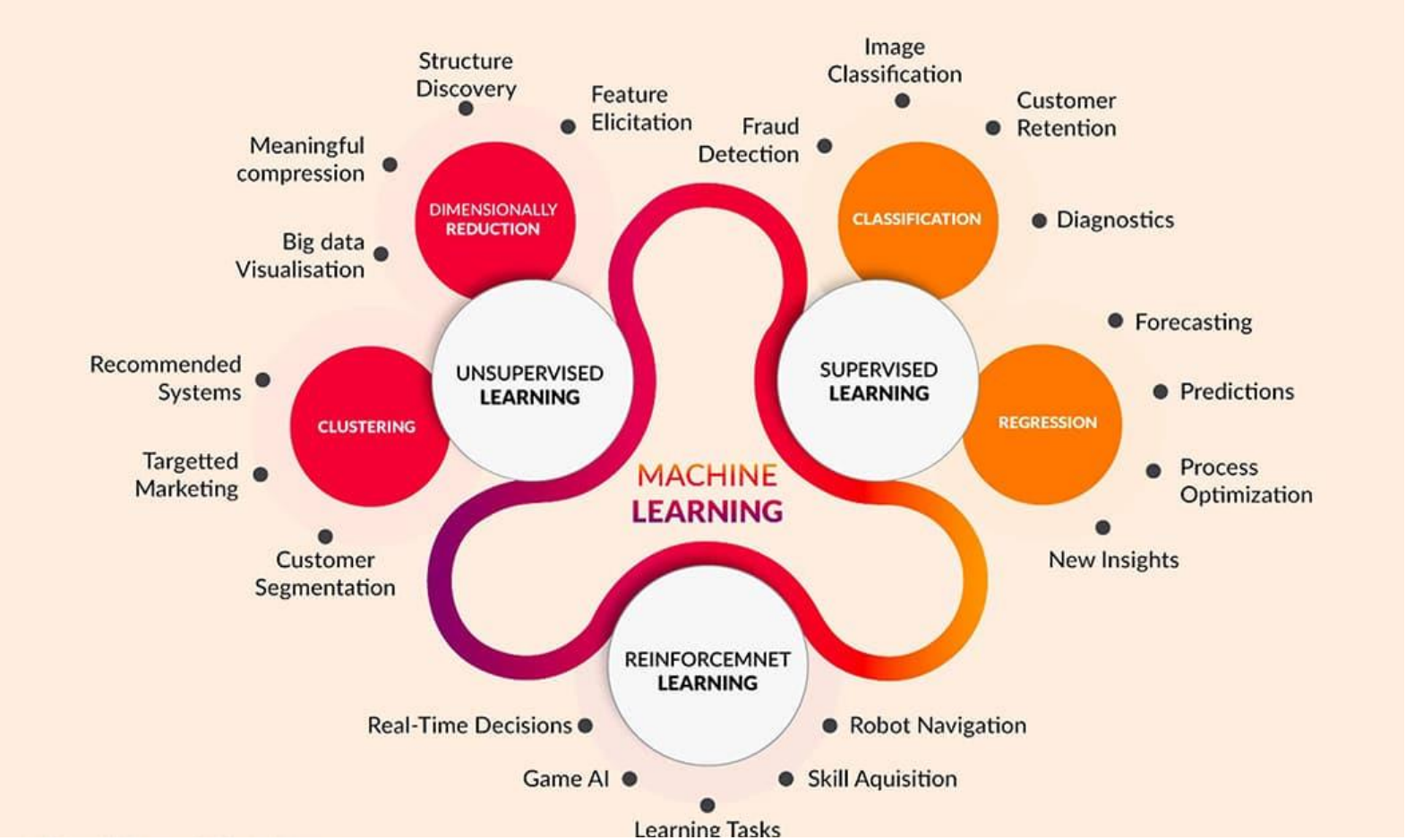
- **Tokenization:** breaks down text into smaller semantic units or single clauses
- **Part-of-speech-tagging**: marking up words as nouns, verbs, adjectives, adverbs, pronouns, etc
- **Stemming and lemmatization:** standardizing words by reducing them to their root forms
- **Stop word removal**: filtering out common words that add little or no unique information, for example, prepositions and articles (at, to, a, the).

Only then can NLP tools transform text into something a machine can understand.

Model Building

We build our model using Classification Models.
Various classifications we used in model building are:

- Logistic Regression
- Random Forest Classifier
- K Nearest Neighbors Model
- Decision Tree Classifier



Logistic regression

Logistic regression is a statistical analysis method used to predict a data value based on prior observations of a dataset. Logistic regression has become an important tool in the discipline of machine learning. The approach allows an algorithm being used in a machine learning application to classify incoming data based on historical data. As more relevant data comes in, the algorithm should get better at predicting classifications within data sets. Logistic regression can also play a role in data preparation activities by allowing data sets to be put into specifically predefined buckets during the extract, transform, load (ETL) process in order to stage the information for analysis.

Random Forest Classifier

Random Forest is an example of ensemble learning, in which we combine multiple machine learning algorithms to obtain better predictive performance.

Why the name “Random”?

Two key concepts that give it the name random:

A random sampling of training data set when building trees.

Random subsets of features considered when splitting nodes.

A technique known as bagging is used to create an ensemble of trees where multiple training sets are generated with replacement.

In the bagging technique, a data set is divided into **N** samples using randomized sampling. Then, using a single learning algorithm a model is built on all samples. Later, the resultant predictions are combined using voting or averaging in parallel.

k-Nearest Neighbors (KNN)

The k-nearest neighbors (KNN) algorithm is a simple, supervised machine learning algorithm that can be used to solve both classification and regression problems. It's easy to implement and understand, but has a major drawback of becoming significantly slows as the size of that data in use grows.

Decision Tree Classifier

Decision Tree algorithm belongs to the family of supervised learning algorithms. Unlike other supervised learning algorithms, the decision tree algorithm can be used for solving regression and classification problems too.
The goal of using a Decision Tree is to create a training model that can use to predict the class or value of the target variable by learning simple decision rules inferred from prior data (training data).
In Decision Trees, for predicting a class label for a record we start from the root of the tree. We compare the values of the root attribute with the record’s attribute. On the basis of comparison, we follow the branch corresponding to that value and jump to the next node.

Train Model

Now the next step is to train the model, in this step we train our model to improve its performance for better outcome of the problem.
We use datasets to train the model using various machine learning algorithms. Training a model is required so that it can understand the various patterns, rules, and, features.

Test Model

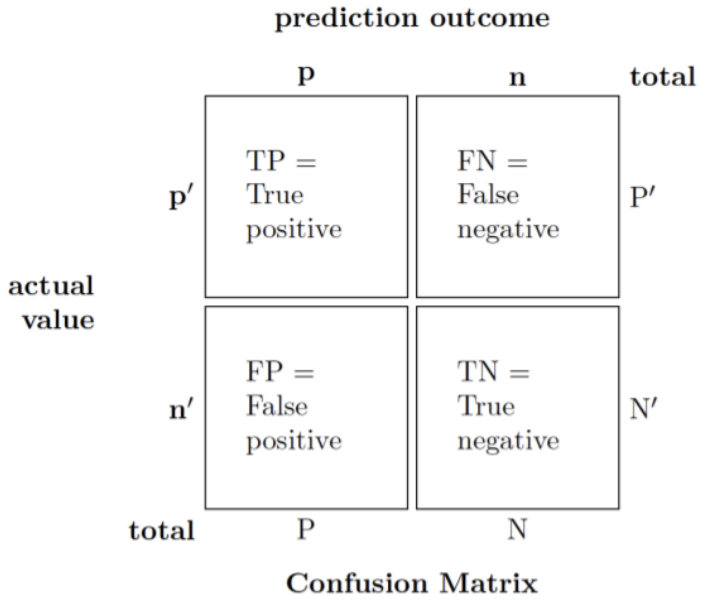
Once our machine learning model has been trained on a given dataset, then we test the model. In this step, we check for the accuracy of our model by providing a test dataset to it.
Testing the model determines the percentage accuracy of the model as per the requirement of project or problem.

Metrics:

In order to be able to evaluate the performance of each algorithm, several metrics are defined accordingly.

Confusion Matrix:

It is very informative performance measures for classification tasks. Ci,j an element of matrix tells how many of items with label i are classified as label j. Ideally we are looking for diagonal Confusion matrix where no item is miss-classified. The matrix in Figure 1 is a good representation for our binary classification. Positive (P) represents toxic label and n (negative) represents non-toxic label.



Confusion Matrix

Elements of confusion matrix; P (positive) represents toxic label and n (negative) represents non-toxic label.

Accuracy:

This metric measures how many of the comments are labeled correctly. However, in our data set, where most of comments are not toxic, regardless of performance of model, a high accuracy was achieved.

$$Precision := \frac{TP + TN}{N' + P'}$$

Precision and Recall:

Precision and recall in were designed to measure the model performance in its ability to correctly classify the toxic comments. Precision explains what fraction of toxic classified comments are truly toxic, and Recall measures what fraction of toxic comments are labeled correctly.

$$Precision := \frac{TP}{P} \quad Recall := \frac{TP}{P'}$$

F Score:

Both Precision and Recall are important for checking the performance of the model. However, implementing a more advanced metric that combines both Precision and Recall together is quite informative and applicable. In this equation, setting β = 1 leads equation to return harmonic mean of Precision and Recall.

$$F_{beta} = (1 + \beta)^2 . \frac{Precision.recall}{(\beta^2 . precision) + recall} \quad Recall := \frac{TP}{P'}$$

Deployment

The last step of machine learning life cycle is deployment, where we deploy the model in the real-world system.

If the above-prepared model is producing an accurate result as per our requirement with acceptable speed, then we deploy the model in the real system. But before deploying the project, we will check whether it is improving its performance using available data or not. The deployment phase is similar to making the final report for a project.

THANK YOU