

**School of Computer Science and Engineering**

VIT Chennai

Vandalur - Kelambakkam Road, Chennai - 600 127

**Final Review Report**

**Programme: BTECH (COMPUTER SCIENCE AND ENGINEERING)**

**Course:** **DATABASE MANAGEMENT SYSTEM**

**Slot:** D2

**Faculty:** PREMALATHA M

**Component:** J

### TITLE: HATE SPEECH / TOXIC COMMENT DETECTION

### TEAM MEMBER(S):

**NAME REG. NO.**

1. **MUSKAAN YADAV 20BCE1622**
2. **JEEL MEHTA 20BCE1117**
3. **SRUJANKUMAR.R.A 20BCE1784**

**ABSTRACT**

With the push towards the decline in data rates and growth of telecommunication networks in the last few years, there has been a huge spike in the number of internet users. Penetration of high bandwidth connection into the remotest part of the world has enabled quite a few users to come up online and express their opinions and thoughts.

However, the another side of coin is that it has lead to increase in instances of hate speech and bullying which not only holds back certain users from opening up freely with ingenious ideas but also spreads negativity, depression, communal/racial hatred and at times can trigger riots. So it becomes imperative for social media ventures to ensure that their platform is free from toxicity to encourage free flow of information and opinions.

**KEYWORDS**

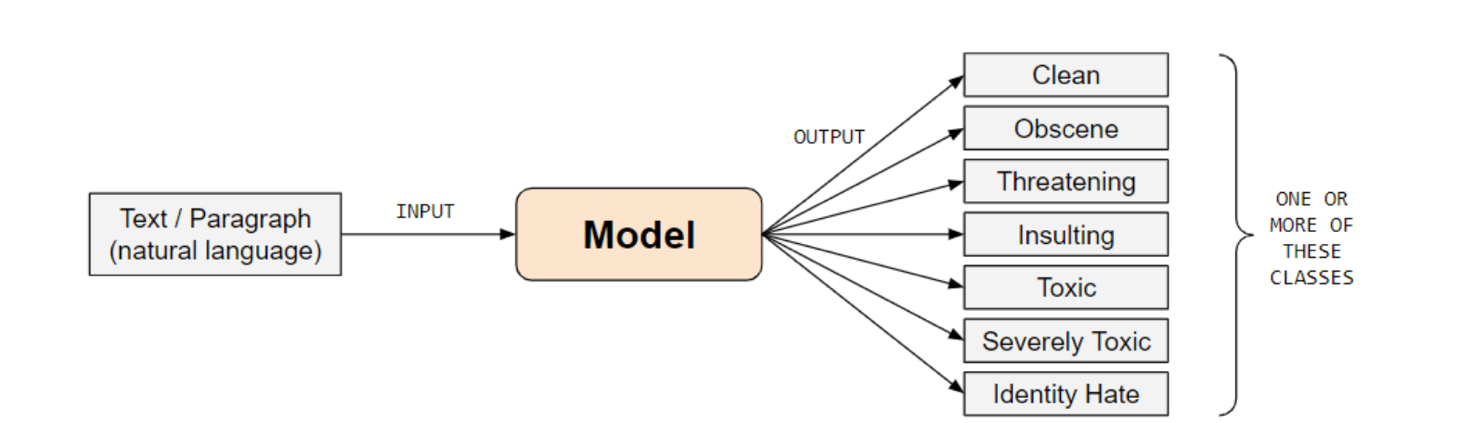
* Data Visualization
* Data Cleaning and Data Pre-processing
* Methodology- Multi-class classification and Multi-label classification problem.
* Algorithms – Decision trees, Binary relevance , Naïve bayes, Classifier chains

**INTRODUCTION**

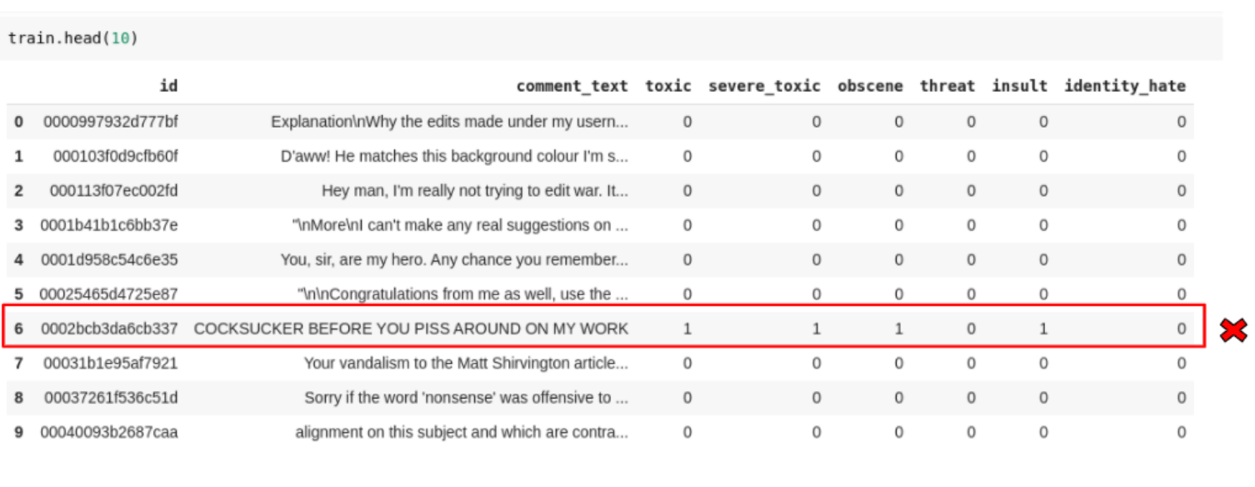
Social networking sites have evolved to become an effective medium of commu- nication all over the world. People can freely share their opinions and ideas with anyone. However, many people are found spreading verbal violence in the form of hate speeches and toxic comments. Thus, it has become a serious challenge for the social networking sites to control the spread of toxic comments. The task of hate speech or toxic comment detection can be posed as a multi-class and multi-label classification task. Companies are building models to automatically flag such content on the sites.

We have taken a collection of posts obtained from various social media sites and build models to classify those posts into six categories namely, toxic, severe toxic, threat, identity hate, obscene and insult. The advantage of this type of data is that these comments represent a true sample of the content present on the social media sites. We began by performing analysis and visualization of the dataset. Since the dataset contained the real comments posted on social media, the data contains noise. We had to preprocess the data to remove any outliers or noise that were present in the dataset. We initially tested the performance of classical models namely, support vector machines, and logistic regression on this task. We then experimented with newer techniques like tree based ensembling methods. We compared the performance of all the models using the mean AUC ROC score as the performance metric.

**OVERVIEW:**



1. **DATA SET DESCRIPTION**:



* The dataset for the problem has been made publicly available by Conversational AI. In the snapshot of the dataset we can see it has approximately 160k training examples. Corresponding to each example we have a set of labels and each label can take two values 0 or 1 depending on whether that particular comment should belong to the class label or not.
* The preparation of the dataset has been done using crowdsourced annotation platforms. This means different examples have been labelled by different users across the internet and hence there is an element of subjectivity in labelling.
* On further exploring the dataset we realise that toxicity is influenced by some other factors too. One such example is the way a comment is rendered. If rendered in a particular way on a particular screen width, the comment takes the form of offensive symbols which are obviously toxic.
* As another example, we have some comments which are very long and detailed, possibly expressing someone’s opinion on some topic. Some of these comments have been classified as toxic without any reason.

**DATA PRE-PROCESSING:**

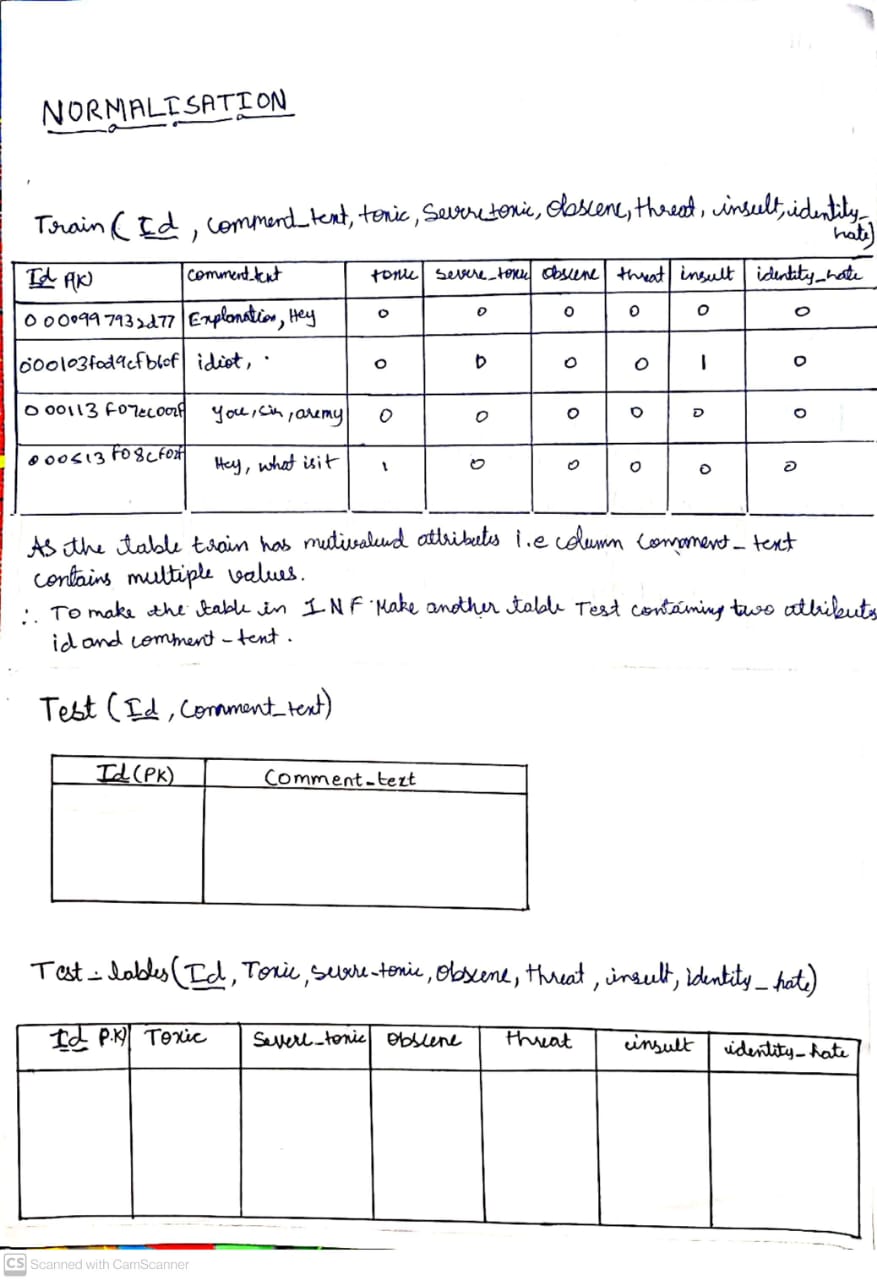
Since the dataset contains the comment text from the internet, it may also consist of the URLs/hyperlinks, IP Addresses, user handles, trailing newline marks, etc. which aren’t of any use in classifying the text. Therefore, we also removed these from the dataset. Lastly, as the case of the word does not matter, we reduced all the letters to the lowercase.

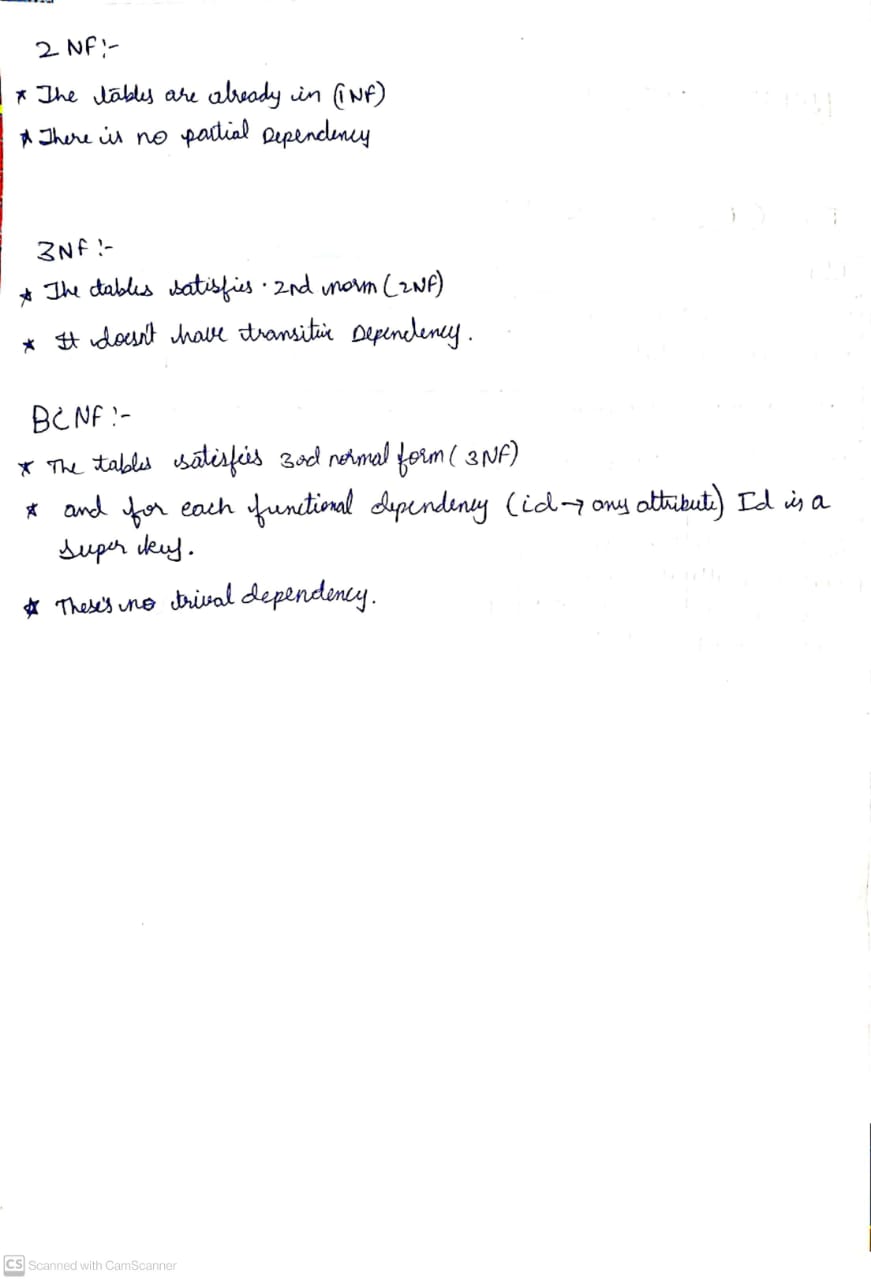
Data Pre-processing mainly consists of two steps :

* Removing the stop words - The stop words are those words in the comment text, which don’t add any physical meaning to the comment, but are generally used as connectors in the natural language in order to make the text grammatically correct. They generally comprise of the auxiliary verbs like is, am, are, was, etc., interrogative pronouns like who, was, were, etc., or any other commonly used words used in the English language. They also have punctuations, that may not be useful. We removed the stop words from the comment text using the NLTK stop word corpus.
* Stemming . - Stemming involves converting a word from its normal form to its base stem. Here, the base stem comprises the set of characters which are required to form a word and its derivatives. In the case of stemming, the base stem need not be a correct English word. Thus, stemming helps in reducing the vocabulary of words in the comment text.

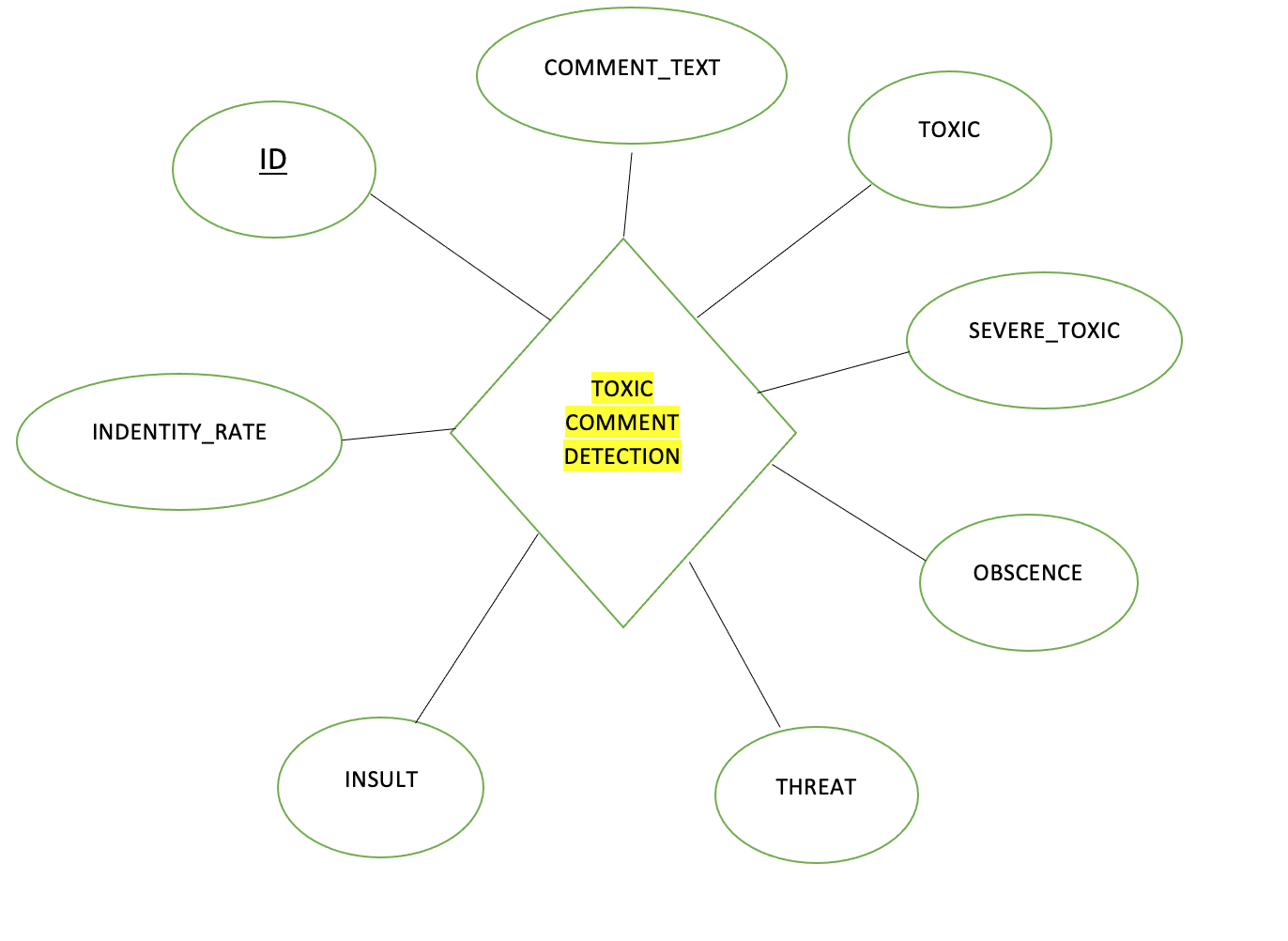
**DATASET URL-** <https://github.com/mrsac7/Data-Mining-Project/tree/master/dataset>

1. **NORMALISATION:**





1. **ER DIAGRAM**

****

Here,

* ID is the primary key.
* Comment\_text is the multivalued attribute.
* Insult , threat, obscence , toxic , severe\_toxic, Identity\_rate are partially dependent on the comment\_text and thus considered as partially dependent attributes.

1. **METHODOLOGY AND ALGORITHM USED:**

**METHODOLOGY:**

**• Multi-class classification** - This type of classification involves putting each data point or example into one of several possible categories as opposed to a binary classification problem in which each example can be classified into only one of the two categories.

• **Multi-label classification** - This type of classification involves examples such that each example can belong to multiple categories and not necessarily only one category. A multi label classification problem can be viewed as a generalised version of the multiclass classification problem in which there is no restriction over how many classes can a training example belong to.

* A comment can belong to any of the seven categories. That’s why the problem is a multiclass classification problem.
* Secondly we note that a comment may be offensive in multiple ways. A comment which is toxic may also be obscene. So a comment can belong to several categories all at the same time and hence, it’s a multilabel classification problem too.

**ALGORITHM USED:**

* DECISION TREES
* BINARY RELEVANCE
* NAÏVE BAYES
* CLASSIFIER CHAINS

1. **IMPLEMENTATION:**

* **PROBLEM DESCRIPTION :**

We are given a set of tweets/comments/posts which may have been collected from a variety of websites including Facebook, twitter, Instagram, etc. The task involves detecting and flagging the comments which may involve display of hate or vulgarity. In particular the model classifies each of the toxic comments into one of the six categories- toxic, severely toxic, threatening, insulting, obscene, identity hate. In practice, if the comment is not toxic, then it is labelled as clean which is added as an additional class label.

* **METHODOLOGIES:**

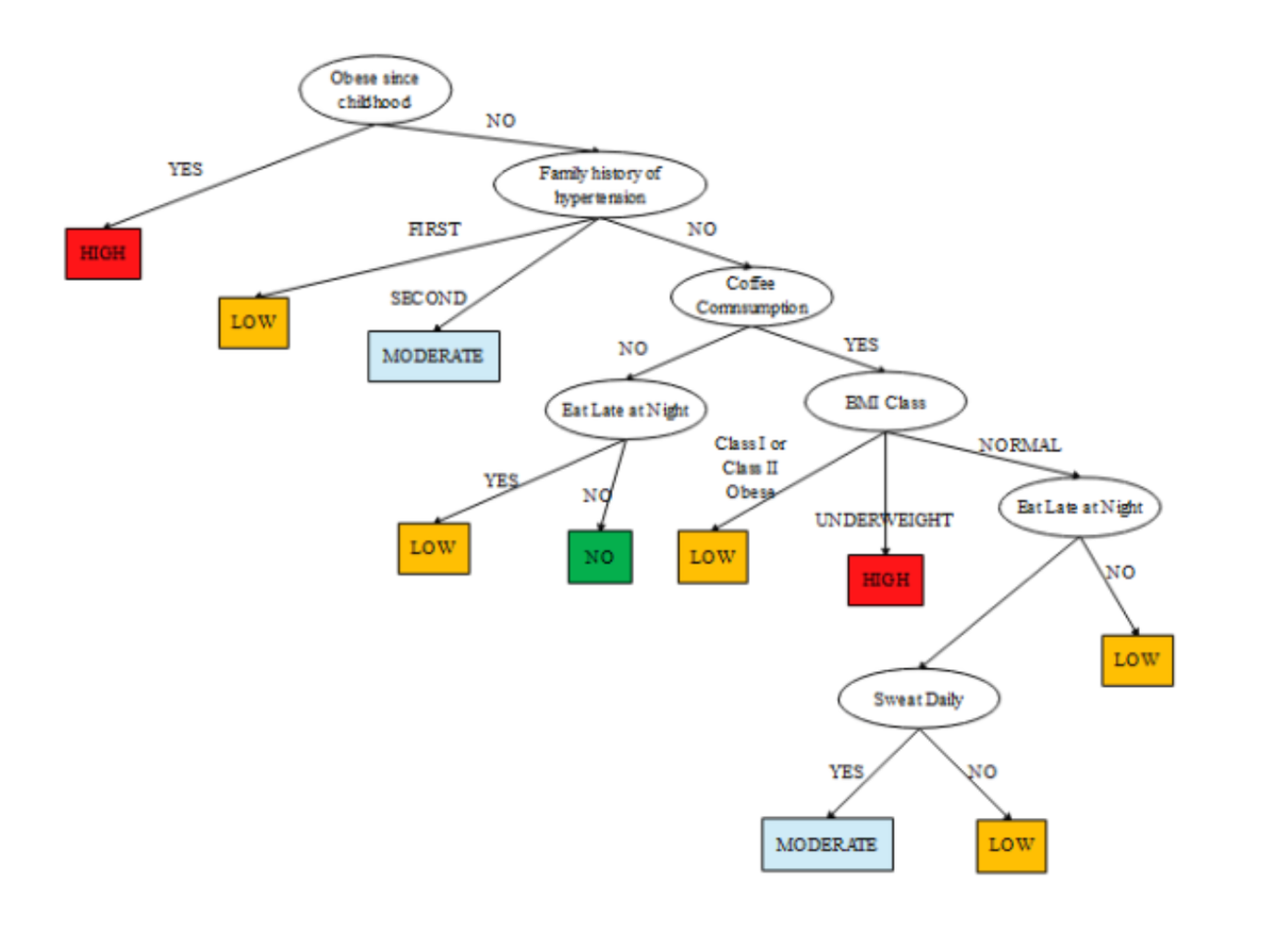


Now according to our problem description, a comment can belong to any of the seven categories. That’s why the problem is a multiclass classification problem. Secondly we note that a comment may be offensive in multiple ways. A comment which is toxic may also be obscene. So a comment can belong to several categories all at the same time and hence, it’s a mutlilabel classification problem too.

* **EXPLAINATION ON ALGORITHM USED:**

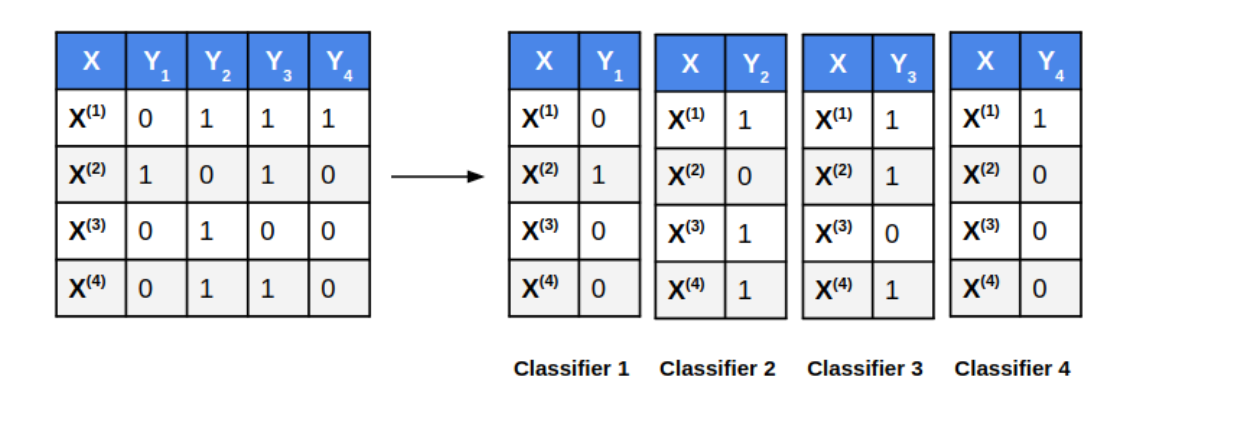
**Decision Trees:** Decision Trees are tree-like structures formed from simple decision rules by analysing the data. Decision trees are supervised learning algorithms. These can be used on both classification as well as regression tasks .

Our approach involves the use of a classification decision tree, in which the decision rules to test are located in the nodes of the tree. Based on the result of the decision node, a particular branch at that node is chosen which leads to some other node other tree with some other decision to make. This process is continued in a recursive fashion until the leaves of the tree are reached at last which contains a class label. This class label is attributed to the data item. Decision trees are well suited for problems in which the data point can belong to several different target classes which is so in our case.



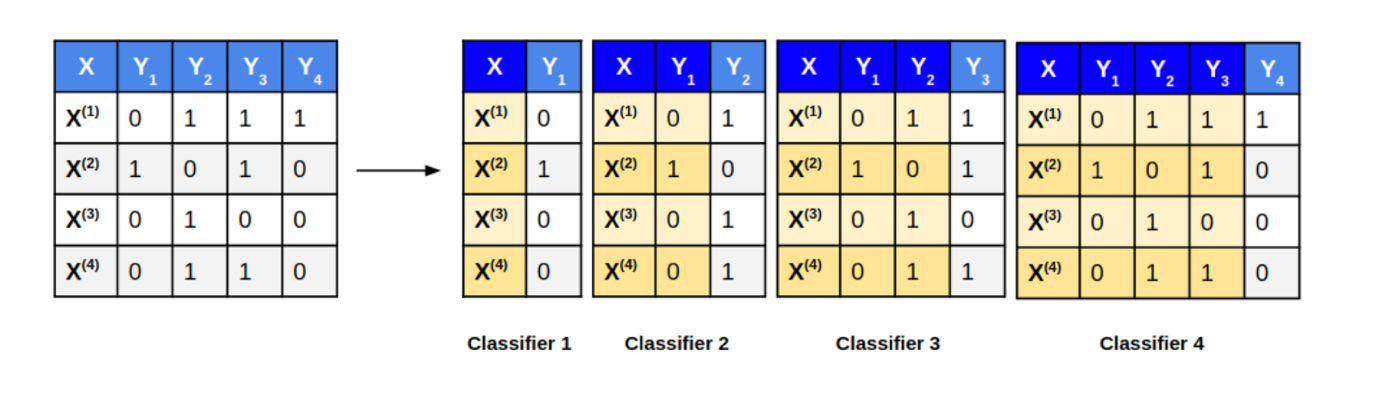
**Binary Relevance:** In this method, we transform the problem into separate single-class classification problems, each of the problems having a single label. We then apply the machine learning algorithm on each problem separately to get the result. As an example, if the problem is to classify a comment with six possible toxic labels, then the problem would be broken into separate six problems, such that each problem has a single label, hence each problem is a single-class classification problem. Thereafter, the results of the six problems can be combined to get all the labels for a comment.

Though this is a simple method to solve such multi-label classification problems, it has drawbacks. Since all the problems are treated as independent problems, this method neglects the correlation between the labels. Hence, it gives poor result if any correlation is observed between the labels.



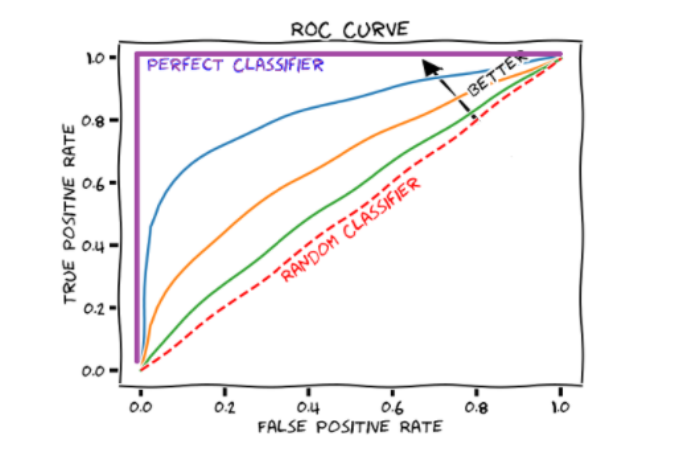
**Classifier Chains:** In this method, we transform the problem into separate single-label classification problems, such that if ith classifier is trained on input variable(s) X, then (i+1) th classifier is trained on input variable X as well as the output produced by ith classifier. Hence, in this method, the first classifier will be the same as the first classifier of Binary Relevance method, but the subsequent classifiers will also include the predictions of the previous classifiers as an input variable.

Thus, this technique also considers the correlation between the labels, since for every new classifier, the predictions of the previous classifiers are taken into account, i.e. for a given target variable, it also considers the correlation between previous target variables. Hence, this method is efficient than the Binary Relevance method when the labels are correlated with each other.



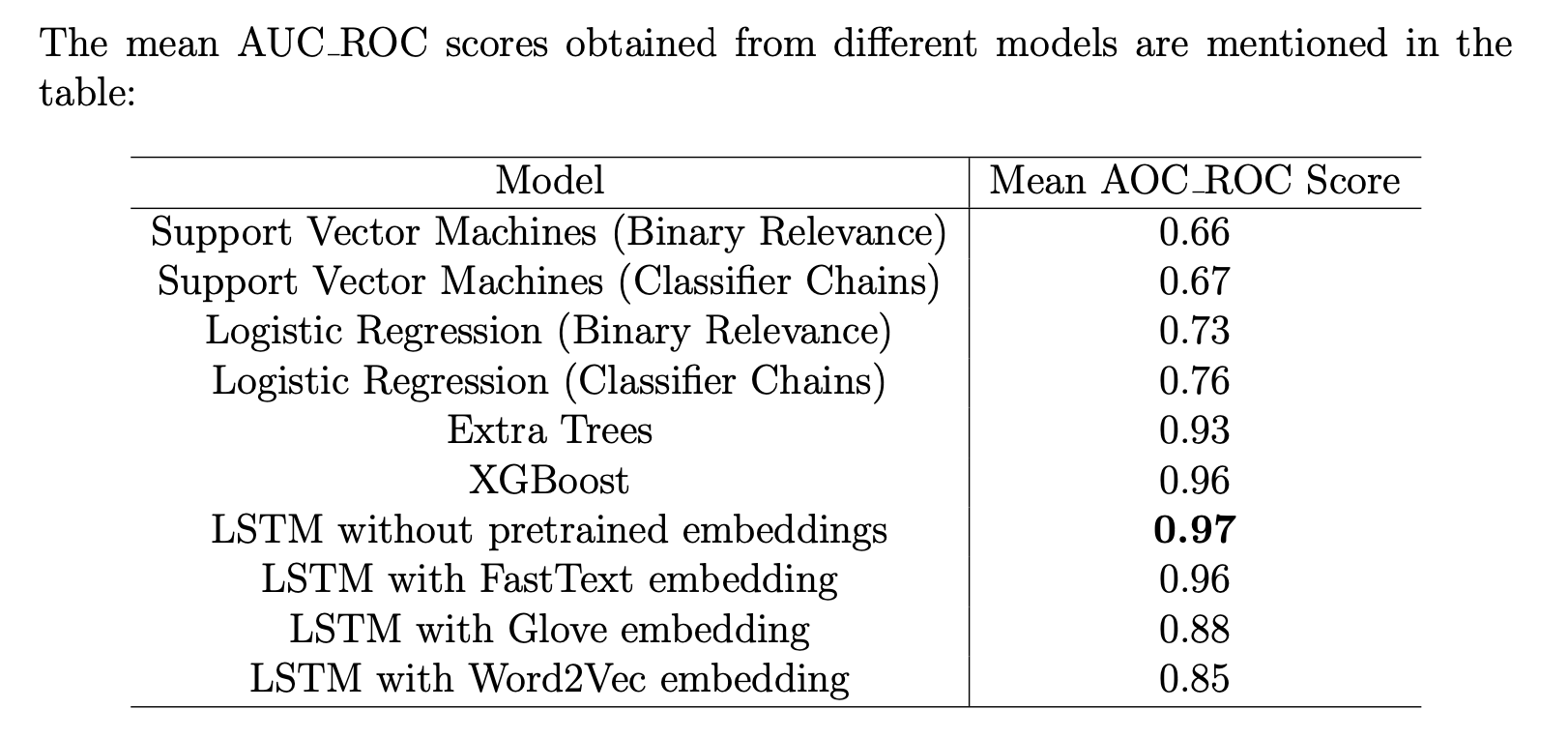
1. **RESULTS AND DISCUSSION:**

The problem involves highly unbalanced dataset. So accuracy is not a well suited performance measure. With only 10% of the training data belonging to the positive class ( hate tags), it is trivial to achieve 90% accuracy by a naive model which simply labels every input as clean. Indeed we verified this by fitting a very simple naive bayes model which achieved a validation accuracy of whopping 97% which looks good on paper only as long as one doesn’t analyse the model predictions manually or by some other performance measures. As quite expected, the recall was merely 65% which means the model is simply unable to recognise 35% of the hate comments. The precision scores were still poorer just 35% indicating even among the comments predicted as hate tags, 65% are misclassified. Needless to say, such a model is not of any use which highlights the essence of a good evaluation metric. Precision-Recall or F1 score seem like the next obvious choice however they have their own share of limitations including selection of threshold value and relative importance to be given to precision vs recall. Hence we finally settled on the ROC curve and AUC score which give a very accurate picture of the performance of a discriminative model.



* A Receiver Operating Characteristic is a curve which plots True Positive Rate vs True Negative Rate.

**COMPARISON TABLE:**



1. **CONCLUSION:**

* We compared the performance of the model based on the mean AOC ROC scores.
* We also found out that the classifier chain method performed slightly better than binary relevance in this task.
* We observed that the classical models Logistic Regression failed to achieve high AUC ROC scores.
* The tree based ensembling models performed significantly better than the classical models.