PROJECT DESCRIPTION

INTRODUCTION

“A small matchstick can cause a large forest fire.”

Similarly, a small typing error can cause a lot of nuisance which can lead to controversies that’ll be harmful to lot of people. To find the these typing errors in a huge amount of Data is nearly an impossible task to do manually. Also, in today’s world technology is increasing rapidly, our future is changing into automation, for example, speech to text apps or automatic translators needs enormous data to increase its work efficiently which requires data like most common words used by people, how frequently those words are used, which word is used for a specific type of emotion.

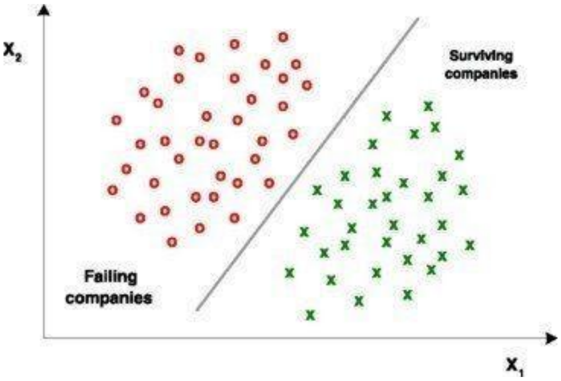
Hence, we have built a machine learning model to find typing errors and find most commonly used words for the given dataset. By using three machine learning models we predict their accuracy scores and compare them to see which suits the best.

LOGISTIC REGRESSION

Logistic regression is a machine learning statistical technique used to predict probability of binary responses based on one or more independent variable.

This is a supervised classification algorithm in which predictors are independent of each other.

It assumes that the data is linearly separable as seen below.



This technique is best suitable when the target variable is binary in nature.

It uses sigmoid function to normalize the values between 0 and 1 for predicting the probability.

$$\mathbf{Y = 1 / 1+} \boldsymbol{e^{-z}}$$

Logistic regression is used in case of predictive analysis it helps to give the surety of occurrence for example –

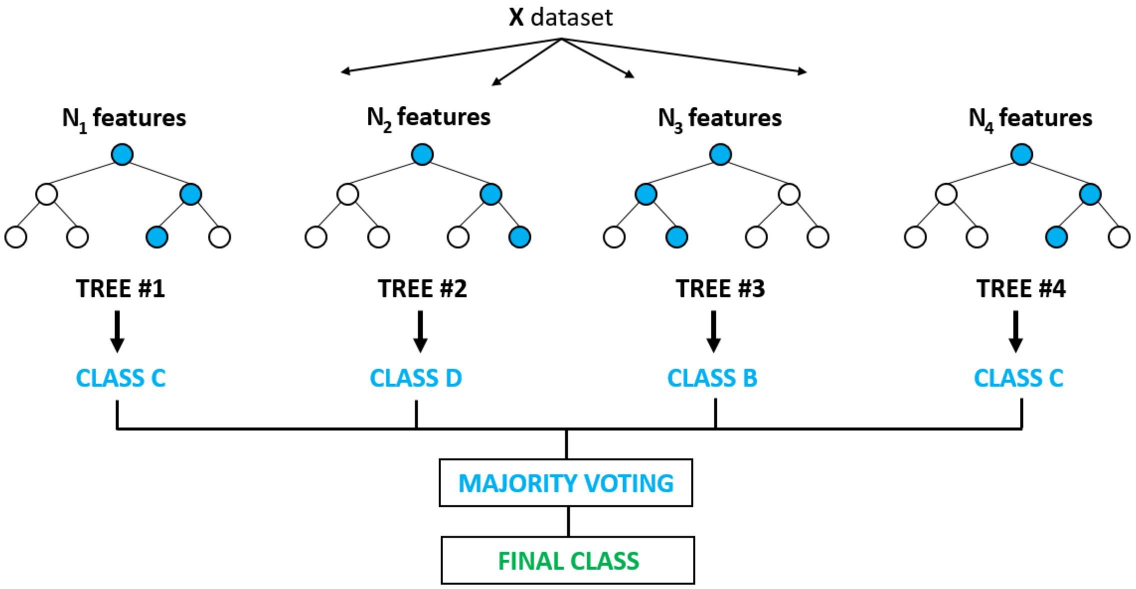
Churn prediction in a company.

It can help in predicting risk of developing a disease.

It can predict the probability of failure of a process under given conditions

RANDOM FOREST

Random forest is a supervised learning algorithm. The “forest” it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the bagging method is that a combination of learning models increases the overall result. Here the data frame used uses the list of common words as the independent variable. This algorithm trains and tests the words with the dependent variable of gender. Here the “bagging” is done using the common words from both text and description. The variables that are considered as dependent variables help predict the outcomes for the test.



According to the training and testing done in this project the training accuracy score is 98% approximately, whereas the testing score is 63% which is very less compared to the train score. The reason for this might be that the predicting of words is very much tougher than predicting a statistical value.

NLP using Naïve Bayes Classifier

This is a classification technique based on Bayes' theorem with an assumption of independence between predictors. In simple terms, a Naive Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

For example, a fruit may be considered to be an apple if it is red, round, and about 3 inches in diameter. Even if these features depend on each other or upon the existence of the other features, a naive Bayes classifier would consider all of these properties to independently contribute to the probability that this fruit is an apple.

Naive Bayesian model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods.

Bayes theorem provides a way of calculating posterior probability P(c|x) from P(c), P(x) and P(x|c). Look at the equation below:

$$\underbrace{P(c\mid x)}_{\text{Posterior Probability}}=\frac{\overbrace{P(x\mid c)}^{\text{Likelihood}}\,\overbrace{P(c)}^{\text{Class Prior Probability}}}{\underbrace{P(x)}_{\text{Predictor Prior Probability}}}$$

$$P(c\mid \mathrm{X}) = P(x_1\mid c)\times P(x_2\mid c)\times\cdots\times P(x_n\mid c)\times P(c)$$

Here,

P(c|x) is the posterior probability of class (target) given predictor (attribute).

P(c) is the prior probability of class.

P(x|c) is the likelihood which is the probability of predictor given class.

P(x) is the prior probability of predictor.

QUESTIONS

Q1) What are the most common emotions/words used by Males and Females?

We first need to remove stop words. Stop words usually refers to the most common words in a language. Some Examples of stop words are:”a”, ”and”, ”but”, ”how”.

Stop words can be removed by using NLTK. NLTK is a leading platform for building Python programs to work with human language data.

After removing stop words. We can count the frequency of each word separately for male and female and the words with highest frequency both for male and female are :-

<u>Most common words used by Male:-</u>

- I - 1567
- The - 631
- like - 316

<u>Most common words used by female:-</u>

- I - 2358
- The - 546
- I'm - 481

Q2) Which gender makes more typos in their tweets?

We will spell check each word with the dictionary. This can be done using enchant.

Enchant is a module in python which is used to check the spelling of a word, gives suggestions to correct words.

If it is found to be incorrect we will add it to the number of typos .This has to be done separately for both male and female.

- Typos made by female are:-

18588

- Typos made by male are:-

18480

Thus, we conclude that female makes more typos in their tweets.

SUMMARY

Machine learning models can do pretty amazing things, even when it comes to something as subjective as the human language.

We have successfully performed ensemble modelling on the given dataset. For this we have used 3 classification algorithms:

1. Logistic Regression: uses a logistic function to model a binary dependent variable.

2. Random Forest: usually trained with “bagging” method.

3. Natural Language Processing (NLP): it programs the computer to process and analyse large amount of natural data.

For this dataset we used ‘gender’ as dependent variable and text, description as independent variables.

Our models have performed relatively well with an accuracy score of 63% using Random Forest followed by NLP with an accuracy of 62% and Logistic Regression with an accuracy of 55%.

The algorithm which suited the best for the given problem is Random Forest with an accuracy of 63%.