TEXT CLASSIFICATION-
XYZ Health Services

PRATHAM NAWAL
https://github.com/PrathamNawal

CONTENTS

XYZ Health Services - TEXT CLASSIFICATION

1. PROBLEM STATEMENT

Build a **classification model to identify Categories and Sub Categories**. Given data set includes fileid, Summary of the call data, Call Data, Previous Appointment and target variables – Categories(5) and Sub Categories (20).

2. INTRODUCTION TO PROJECT

A. CASE STUDY EXPLANATION

This report is an analysis of Text data from XYZ Health services. The data contains call data from various telephonic conversations occurred between patient and the receiver at the XYZ Health Services. We have to classify the telephonic conversations into **5 major categories and 20 sub categories**. The aim of this project is to build two **Text classification models** to classify call data to 5 categories and sub categories **to understand the true purpose of the call**.

B. PAIN AND GAIN ANALYSIS

The Pain and Gain analysis of this project needs very **subtlety in Perception and Understanding of the communication** usually varies from **Tone, Body language, Vocabulary and Absurdity levels** while communicating.

In **real world scenarios,** a person needs to understand the subtlety of this usage, requires **second-order interpretation** of the speaker's or writer's intentions; different parts of the brain must work together to understand purpose.

Using Analytics to identify the purpose of the call will be beneficial. This approach will reduce **Time Consumption of text classification**. Applications are such as **analyzing healthcare reviews** to know if a positive review is really positive or how many people are affected by a health issue.

Examples: **Optum Labs,** an US research collaborative, has collected **EHR**s of over 30 million patients to create a database for predictive analytics tools that will improve the delivery of care.

C. DOMAIN KNOWLEDGE

XYZ Health Services is a top ranked Health care provider in USA with stellar credentials and provides high quality-care with focus on end-to-end Health care services. The Heath Care Services range from **basic medical diagnostics to critical emergency services.** The provider follows a ticketing system for all the telephonic calls received across all the departments. Calls to the provider can be for **New Appointment, Cancellation, Lab Queries, Medical Refills, Insurance**

Related, General Doctor Advise etc. The Tickets have the details of **Summary of the call** and **description of the calls** written by various staff members with no standard text guidelines.

The challenge is, based on the Text in the Summary and Description of the call, the ticket is to be classified to **Appropriate Category (out of 5 Categories) and Subcategories (Out of 20 Sub Categories).**

D. EXPLORATORY DATA ANALYSIS

The Given dataset contains **57280 Observations and 7 Variables**- fileid, summary, data, previous appointment, categories, sub categories and ID The Independent variables needed for modelling should be structured, but our only predictor variable is unstructured. So, our **first goal is to convert the unstructured call data to structured** data . We use nltk package for this text mining operations involved.

1. UNDERSTANDING THE DATA

Let's look at Summary and head of the data.

h

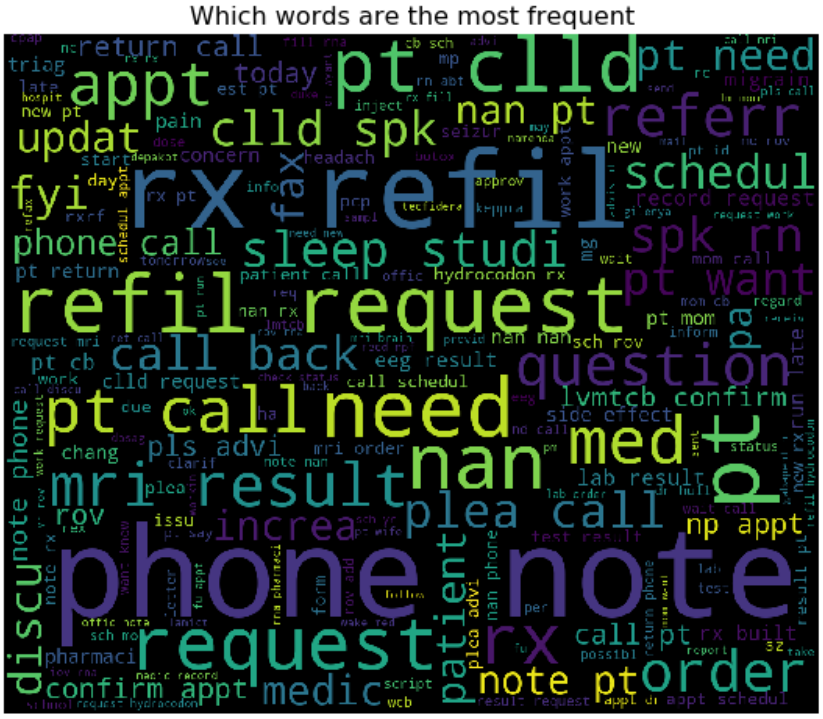
By observing above head of the data, we can infer that call data is in Rich Text Format. By researching, it is found that we have to convert it into a proper text format first. So, while preprocessing, I have introduced three functions to clean the call data. Data contain many meaningless characters such as Punctuation marks, Numbers, unwanted spaces, single alphabets. We have to clean all these irrelevant characters from the sumamry and call data. This is taken care in Pre Processing step.

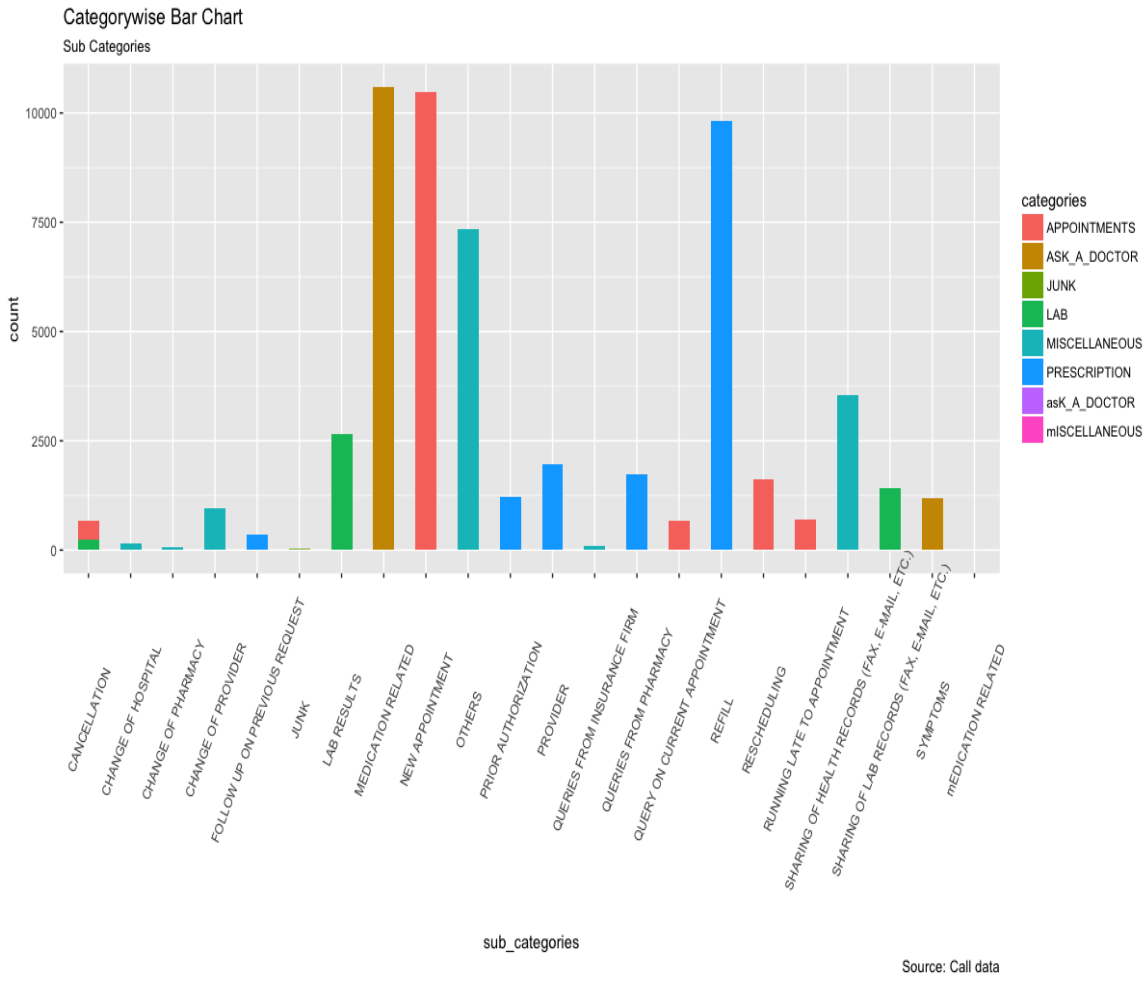
The **label feature specifying the categories and sub categories** are in a character form by default. We have converted it to factor while analysis.

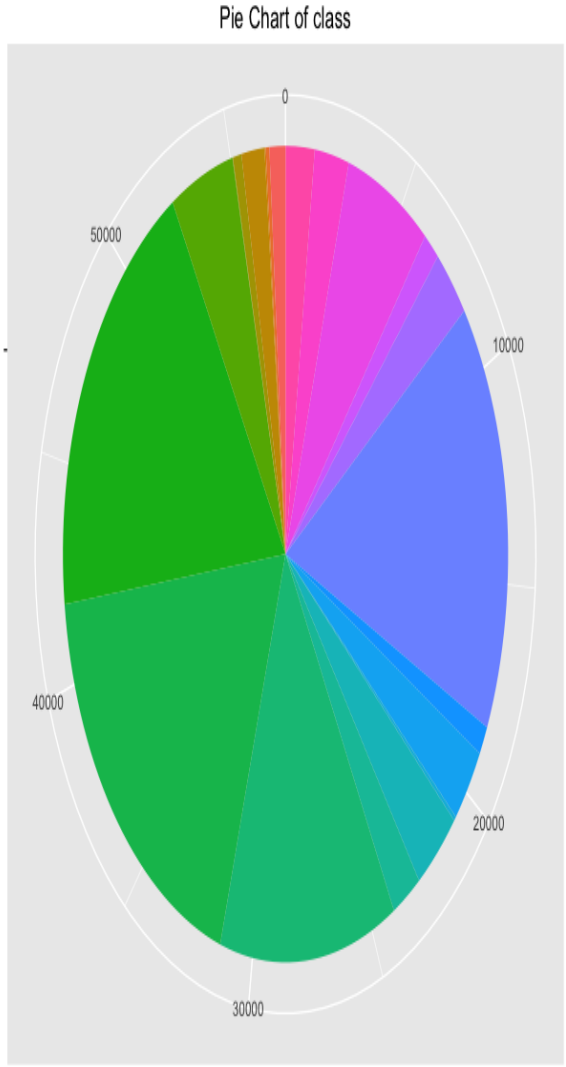
B. UNDERSTANDING VISUALIZATIONS

For getting Our First level of Insights, we are going to discuss Visualizations here in this module. We have constructed two Word clouds- Most frequent words in the summary and most frequent terms in the call data. And a Bar plot is constructed to represent most frequent terms of the whole corpus which have **frequency more than** .

We are going to analysis these four visualizations and extract most frequent words which used in both Categories and Sub Categories. It is very important to know that if a word occurs most frequently in both of the data set, then it is evident that those words doesn't contribute to our model in classifying call data. We are going to eliminate all those frequent words which appear in both data sets for fair distribution of the data.




Categorywise Bar Chart
Sub Categories
count
10000
7500
5000
2500
0
CANCELLATION
CHANGE OF HOSPITAL
CHANGE OF PHARMACY
CHANGE OF PROVIDER
FOLLOW UP ON PREVIOUS REQUEST
JUNK
LAB RESULTS
MEDICATION RELATED
NEW APPOINTMENT
OTHERS
PRIOR AUTHORIZATION
PROVIDER
QUERIES FROM INSURANCE FIRM
QUERIES FROM PHARMACY
QUERY ON CURRENT APPOINTMENT
REFILL
RESCHEDULING
RUNNING LATE TO APPOINTMENT
SHARING OF HEALTH RECORDS (FAX, E-MAIL, ETC.)
SHARING OF LAB RECORDS (FAX, E-MAIL, ETC.)
SYMPTOMS
mEDICATION RELATED
sub_categories
categories
APPOINTMENTS
ASK_A_DOCTOR
JUNK
LAB
MISCELLANEOUS
PRESCRIPTION
asK_A_DOCTOR
mISCELLANEOUS
Source: Call data


Pie Chart of class
0
10000
20000
30000
40000
50000
class
CANCELLATION
CHANGE OF HOSPITAL
CHANGE OF PHARMACY
CHANGE OF PROVIDER
FOLLOW UP ON PREVIOUS REQUEST
JUNK
LAB RESULTS
MEDICATION RELATED
NEW APPOINTMENT
OTHERS
PRIOR AUTHORIZATION
PROVIDER
QUERIES FROM INSURANCE FIRM
QUERIES FROM PHARMACY
QUERY ON CURRENT APPOINTMENT
REFILL
RESCHEDULING
RUNNING LATE TO APPOINTMENT
SHARING OF HEALTH RECORDS (FAX, E-MAIL, ETC.)
SHARING OF LAB RECORDS (FAX, E-MAIL, ETC.)
SYMPTOMS
mEDICATION RELATED

Source: mpg

By observing two word clouds of Summary and Call Data, we have enough information to produce our first level of Insights.

FIRST LEVEL OF INSIGHTS OR KNOWN INSIGHTS

1. Different purposes of the call have distinct frequency proportions in the data.
2. Words like **'request', 'referr', 'note', 'please', 'pls'(please), 'phone'** are most common frequent words in all kinds.
3. There are few words which are frequent yet clearer in their appearance which are very helpful for the model in classifying.
4. There is a certain amount of noise the both the categories and subcategories, It might be due the improper data entry operations…
5. Significant use of short forms is used for words while writing the call data which makes it a messier text. To draw insights we need to dig deeper into the short forms and extract their true meanings.

4. PRE PROCESSING AND CLEANING- TEXT MINING

Our Unstructured Data should be converted to Structured Data in order to do modelling.
We are using Text mining's '**nltk**' package to do this. A corpus is prepared to **preprocess and clean the data and do tokenization** on it.
A user defined function ***nltk*** is created to do preprocessing and cleaning of the corpus. This function takes in a vector with all the call conversations in it and convert it to a corpus and cleans it. The following steps are done on the corpus:

A. CASE FOLDING

The first preprocessing step is Case folding. Here, we are converting all the letters in the **Corpus to lowercase** using python's base function *tolower*.

B. REMOVING NUMBERS

In this step, we are **freeing corpus from numbers**. Here, we define our own functions.

C. REMOVE STOPWORDS

This is very interesting preprocessing step. This step is about eliminating words that doesn't make any meaning. Stopwords of English would be enough, but since the dataset contains several short words in the form of short forms which are of no meaning to use.

Coming to user defined stopwords, we have few frequent words which can be considered. We are going to deal this in feature engineering with more analysis on them.

D. USER DEFINED FUNCTIONS

While checking data for initial insights, there are **few call data conversations which is not written in the standard rich text format. It posseses several missing words and regular expressions required for the rtf format.**

To avoid this situation, I have defined several cleaning and manipulating functions to make sure that every conversations is converted into rtf standard text format and further cleaned to make it available for insights. It contained several uses of "xxx.." terms etc.

E. REMOVING PUNCTUATION MARKS

We have use nltk's ***removePunctuation*** function to **remove all punctuation marks** such as comma, full stop, parenthesis, various brackets etc.., from the corpus.

F. STEMMING

For grammatical reasons, document contains different **inflectional forms like tense forms and derivational forms,** we are performing **stemming to reduce** all those words **to their root word.** We are using nltk's ***stemWord*** function to do this. Stemming greatly help in reducing total number of terms and increase weighting.

G. STRIPING WHITESPACES

The above performed preprocessing steps **left our corpus with many leading and trailing whitespaces** within documents. We are cleaning all of them in one go using nltk's ***stripWhiteSpace*** function. With this step our basic preprocessing is completed.

CORPUS TO DOCUMENT TERM MATRIX:

All the above discussed preprocessing and cleaning steps are **wrapped in a user defined function** for simplicity.

A **Document Term Matrix** (DTM) is created from the corpus. **Term Frequency** is considered as weighting to create Document term matrix to keep DTM simple. DTM has **9** documents and **30555** terms.

This Document term matrix is then converted to Data frame for Feature engineering.

Data frame has **52780** observations and **6** features (including target class) which has **Document frequency** of varying from **74 to 8976**. We have successfully converted the raw **Unstructured Data to Structured Data**.

FINAL DATA SET

After doing Feature Engineering on the Data, we have taken **two sets of data**. One containing **Categories** as the target variable and the other one has **Sub Categories** as the target variable. Overall both the dataset has the same dimensions of 40996 observations and 6 variables.

Upon Observing, **fileid and ID** variables don't contribute much when it comes to extracting insights from the datasets. We have kept only **summary of call data, call data, previous appointments** as the significant feature.

An important point to note here is that when choosing **categories** as the target variable, we have used **sub categories** as one of the features and while choosing **sub categories** as the target variable we have chosen **categories** as one of the features.

We are using **stratified sampling** to **preserve this ratio throughout sampling and splitting**.
We have kept the ratio of train/test in proportions of 70:30.

7. PREDICTIVE MODELLING

We are considering **Naïve Bayes as our Base model** which is very significant in Text classification because of its **assumption of considering all variables equally important and independent**. **Random forest model** is our **ensemble model** in this analysis. As Random forest can handle numeric and factor data, we are providing both for a given sample and validate the performance.

A. NAÏVE BAYES MODELLING

Using '**naivebayes**' package, we are training our **Naïve Bayes classifier with factor data**. Being a Bayesian classifier with an assumption of having equally important and independent features, we are expecting a very good accuracy with Naïve Bayes.

Naïve Bayes model with default parameters trained on **6 features** and **78.59 as highest accuracy**. Being a base model, Naïve Bayes gave very good accuracy with this huge data.

B. RANDOM FOREST MODELLING

Being an ensemble model, random forest is popular in providing very good model. We have **trained our**
Random forest model with Text data and Factor data. Random forest model also gave good accuracies.

Our ensemble model provided a **highest accuracy of 76.69**.

It is very interesting to know that **text data is more informative in our Classification than factor data**.

8. CHOOSING MODEL

We have chosen our Naïve Bayes as our Model for this analysis. We are **tuning our Naïve Bayes** to get more reliable and robust model for Text Classification.

Being a classifier based on **conditional probabilities,** Naïve Bayes has a parameter called **Laplace estimator**. Generally while modelling, there are features who have **zero probability to a class which may cause disturbance while evaluating**. We have set this parameter to 1 (Laplace = 1). **Laplace estimator will make sure there are no zero probabilities throughout the model and set a minimum of 1.**

This **Laplace estimator tuning has increased accuracy** further and **made our model more appropriate and robust** for our problem. Naïve Bayes model with Laplace estimator has evaluated with a **median of 77.89** and
a **maximum accuracy of 79.28**.
"Naïve Bayes model with Laplace estimator as 1 resulted us a more Robust model."

Considering error metrics for this problem "Text Classification" is easy to put on. As there are situations **when a patient calling for emergency situations might be misclassified as the one with least priority value** and a situation where the patient is **calling for general advices** is put on the top of the priority order for **immediate attention** from the Doctor. So, we **consider both False Positive and False Negative** and take them as whole as **Misclassification error and minimize it**.

We are **aiming to freeze the model which is giving least misclassification error**. So, the model and seed with highest accuracy is considered to be our model of Deployment.

Hence, **model with is an appropriate model for this Text classification problem.**

10. CONCLUSION

Text classification problem combined **with both Text mining and Data mining** is really an **interesting problem to work on**. Being a subtlest phenomenon, categorizing a call data can be a very time consuming and mentally exhaustive process to be carried out.

In this case study, we have converted **Unstructured Raw Data to clean preprocessed and Structured Data** with a lot of information to work on. We have **compared most frequent terms** between the target classes to help our model with more clearly classified information and finally built models and **tuned them to perfection**