RESTAURANT REVIEW

CLASSIFICATION

Phodkar Parth Prashant Priyanka

Department of Electronics Engineering Shah and Anchor Kutchhi Engineering College, Mumbai 400088, India parth.phodkar\_19@sakec.ac.in

Arakade Aditya Vijay Chhaya

Department of Electronics Engineering Shah and Anchor Kutchhi Engineering College, Mumbai 400088, India aditya.arakade\_19@sakec.ac.in

Patil Vedant Bhausaheb Jayashri

Department of Electronics Engineering Shah and Anchor Kutchhi Engineering College, Mumbai 400088, India vedant.patil\_19@sakec.ac.in

Sawant Sarvesh Shashikant Smruti

Department of Electronics Engineering Shah and Anchor Kutchhi Engineering College, Mumbai 400088, India

sarvesh.sawant\_19@sakec.ac.in

Prof. Rameshwari Mane

Department of Electronics Engineering Shah and Anchor Kutchhi Engineering College, Mumbai 400088, India

rameshwari.mane@sakec.ac.in

***Abstract****—* Machine Learning is an application of AI that provides systems the ability to automatically learn and improve from experience without being explicitly programmed. Machine Learning can be used to help solve AI problems and to improve NLP by automating processes and delivering accurate responses.

***Keywords—*** Machine learning, Natural language Processing, Classification, Linear Logistic Regression , Bag-of-words, ML Models.

**I. INTRODUCTION**

In our project we have used an algorithm to classify a restaurant review whether it is positive or negative using sentiment analysis in NLP (Natural Language Processing) which is a part of machine learning. We have implemented using Linear Logistic Regression model. There are various machine learning models like Naïve Bayes classifier, Support vector machine (SVM), decision tree, K-clustering, random forest etc. which we have performed and compared. We have selected the algorithm based on accuracy.

1. **SOFTWARE**
2. ***Google colab***

Colab notebooks allow you to combine executable code and rich text in a single document, along with images, HTML and more., we can perform the following using Google Colab.

* Write and execute code in Python
* Document your code that supports mathematical equations
* Create/Upload/Share notebooks
* Import/Save notebooks from/to Google Drive
* Import/Publish notebooks from GitHub
* Import external datasets e.g. from Kaggle
* Integrate PyTorch, TensorFlow, Keras, OpenCV
* Free Cloud service with free GPU

1. **LIBRARIES**

***A. Numpy***

NumPy is a general-purpose array-processing package library. It provides a high-performance multidimensional array object, and tools for working with these arrays. It is the fundamental package for scientific computing with Python. It contains various features including these important ones:

* A powerful N-dimensional array object
* Sophisticated (broadcasting) functions
* Tools for integrating C/C++ and Fortran code
* Useful linear algebra, Fourier transform, and random number capabilities

Besides its obvious scientific uses, NumPy can also be used as an efficient multi-dimensional container of generic data. Arbitrary data-types can be defined using NumPy which allows NumPy to seamlessly and speedily integrate with a wide variety of databases.

In our project we have used the concatenate () function for printing the predicted values with actual values in an array format.

***B. Matplotlib***

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. We have used its pyplot module to plot the confusion matrix

***C. Pandas***

Pandas is an open source Python package that is most widely used for data science/data analysis and machine learning tasks. We have used it for importing our dataset file

***D. nltk***

The Natural Language Toolkit (NLTK) is a platform used for building Python programs that work with human language data for applying in statistical natural language processing (NLP). It contains text processing libraries for tokenization, parsing, classification, stemming, tagging and semantic reasoning

***E. sklearn***

Scikit-learn (sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms.

**IV. STEPS IN DEVELOPING ML ALGORITHM**

***A. Collect Data***

Data collection is the process of gathering and measuring information from countless different sources. Our dataset is taken from Kaggle website. There are numerous dataset repositories available like UCI, Datasetlist, Fastai datasets etc.

Dataset is available in tsv format which is imported using read\_csv () function from pandas library.

***B. Prepare Data***

For performing analysis on the reviews we need to prepare the data, we have used following methods to prepare the data.

1. Re.sub: - Sub function from re library has been used to replace punctuations with a space
2. Lower: - This function converts all the text to a lower case.
3. Split: - This function splits the text into array of words by splitting the words between each space in the text.
4. Stopwords: - A stop word is a commonly used word (such as “the”, “a”, “an”, “in”) that does not provide any meaningful information to the algorithm to decide the output. Nltk consists of corpus module which has a predefined list of common stopwords.
5. Porterstemmer: - Stemming is the process of producing morphological variants of a root/base word. Stemming programs are commonly referred to as stemming algorithms or stemmers. A stemming algorithm reduces the words “chocolates”, “chocolatey”, “choco” to the root word, “chocolate” and “retrieval”, “retrieved”, “retrieves” reduce to the stem “retrieve”.
6. Finally join the split words using join () function and append them using append () function to an array.

***C. Analyze the input Data***

Bag of words is a Natural Language Processing technique of text modelling. In technical terms, we can say that it is a method of feature extraction with text data. This approach is a simple and flexible way of extracting features from documents.

A bag of words is a representation of text that describes the occurrence of words within a document. We just keep track of word counts and disregard the grammatical details and the word order. It is called a “bag” of words because any information about the order or structure of words in the document is discarded.

CountVectorizer from sklearn is used to select max features based on frequency. Data is then stored into x and y variables using fit\_transform () and iloc [] functions.

Using the train\_test\_split function from sklearn the data is split into X\_train, X\_test, Y\_train, Y\_test and specify the test size in parameters.

***D. Training algorithm***

We have selected the Logistic Regression linear model for testing the dataset. This model is used from sklearn library

***E. Testing algorithm***

For testing we use the predict () function on X\_test and use the numpy concatenate function to print predicted and actual outcomes in 2d array.

***F. Result***

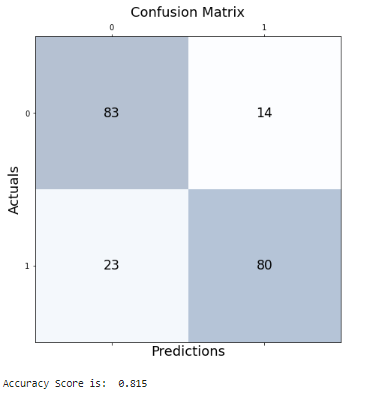
A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

Let's now define the most basic terms, which are whole numbers (not rates):

1. true positives (TP): These are cases in which we predicted yes (they have the disease), and they do have the disease.
2. true negatives (TN): We predicted no, and they don't have the disease.
3. false positives (FP): We predicted yes, but they don't actually have the disease. (Also known as a "Type I error.")
4. false negatives (FN): We predicted no, but they actually do have the disease. (Also known as a "Type II error.")

We have used confusion\_matrix () to get the confusion matrix and accuracy\_score function to get the accuracy score from sklearn library.

Using matplotlib.pyplot we have plotted the confusion matrix.



And we have also added a specific single review classification into positive and negative review.

Positive review: -

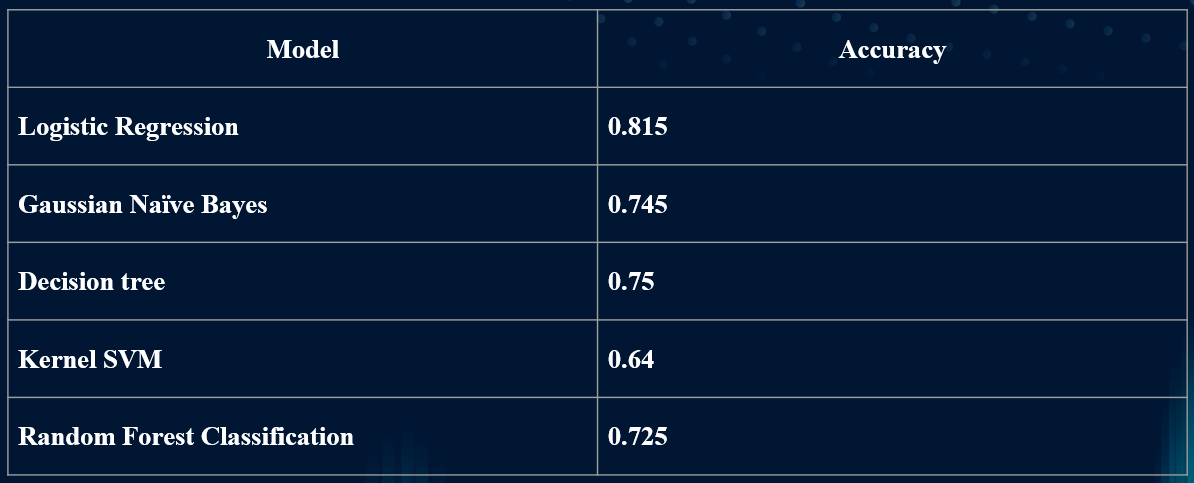


Negative review: -



**V. CHOOSING ALGORITHM**

We have tried many ML models by implementing them in the code and we have selected Logistic Regression linear model as it is giving the highest accuracy of 0.815 (81.5 %).



**VI. LEARNINGS**

* Applications of NLP in – Review prediction, Email classification, Article categorization, chat bot etc.
* Different machine learning Models.
* Need for Deep learning in NLP.
* Various functions of modules and their use cases.

**VII. REFERENCES**

1. Aleksandr Romanov, Konstantin Lomotin , Ekaterina Kozlova. Application of Natural Language Processing Algorithms to the Task of Automatic Classification of Russian Scientific Texts. [Online]. Available: <https://datascience.codata.org/articles/10.5334/dsj-2019-037/>

1. Shuohang Wang and Jing Jiang. Learning natural language inference with LSTM. In Proceedings of NAACL, 2016. [Online]. Available: <https://aclanthology.org/N16-1170.pdf>

1. Data set, Restaurant Review dataset <https://www.kaggle.com/datasets/jurk06/restaurant-dataset?resource=download>