Low Level Document

News Articles Sorting

LLD History :-

Date	Version	Author
20/02/2022	V1.2	Umang Tank

Approval Status:

Version	Review Date	Reviewed By	Approved By	Comments

Contents

Abstract

Abstract

- Nowadays on the Internet there are a lot of sources that generate immense amounts of daily news. In addition, the demand for information by users has been growing continuously, so it is crucial that the news is classified to allow users to access the information of interest quickly and effectively.

- This way, the machine learning model for automated news classification could be used to identify topics of untracked news and/or make individual suggestions based on the user's prior interests.

1. Introduction

1.1. What is Low-Level design document?

The goal of LLD or a low-level design document (LLDD) is to give the internal logical design of the actual program code for Back Order Prediction Model. LLD describes the class diagrams with the methods and relations between classes and program specs. It describes the modules so that the programmer can directly code the program from the document.

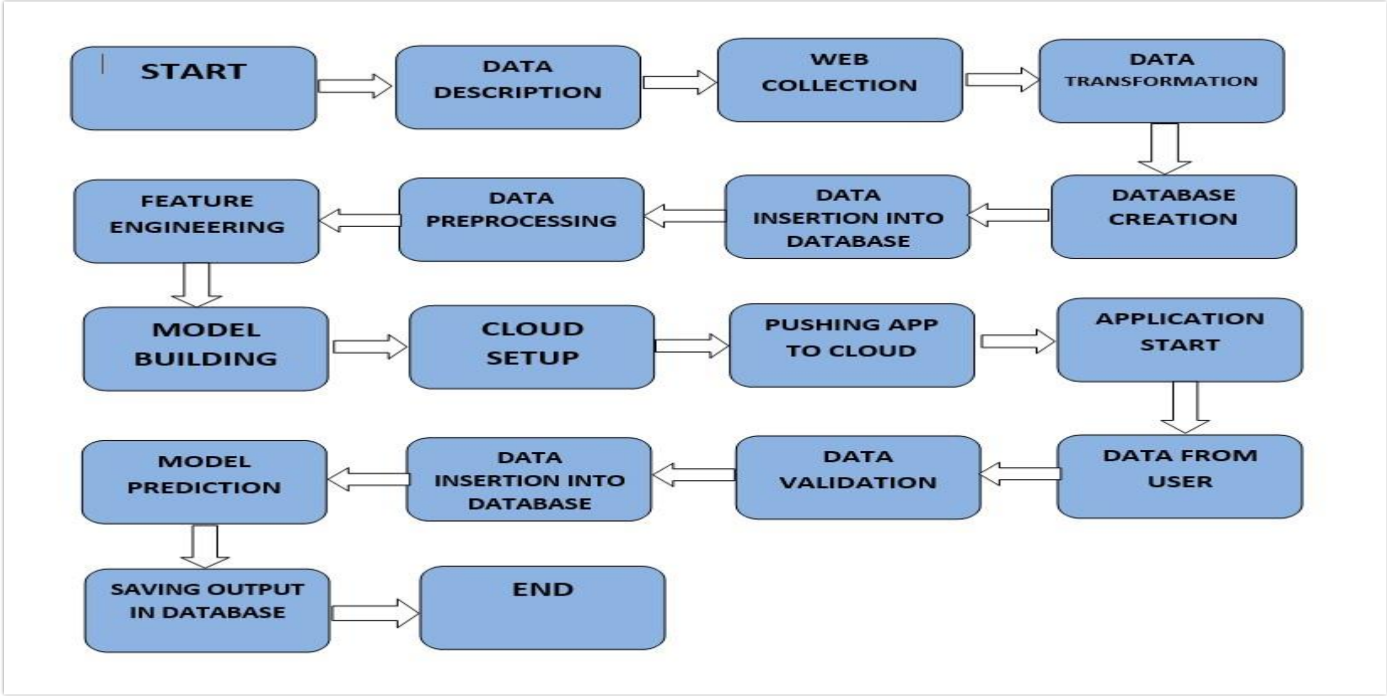
1.2. Scope

Low-level design (LLD) is a component-level design process that follows a step-by step refinement process. This process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the data organization may be defined during requirement analysis and then refined during data design work.

1.3. Constraints

Internet connection is a constraint for the application. Since the application fetched the data from the database, it is crucial that there is an Internet connection for the application to function. Since the model can make multiple requests at same time, it may be forced to queue incoming requests and therefore increase the time it takes to provide the response.

2.Architecture



3.Architecture Description

3.1. Data Description

The dataset provided to us contains many rows, and 2 independent features. We aim to predict category of a news. So this clearly is a classification problem, and we will train the classification models to predict the desired outputs based on input News.

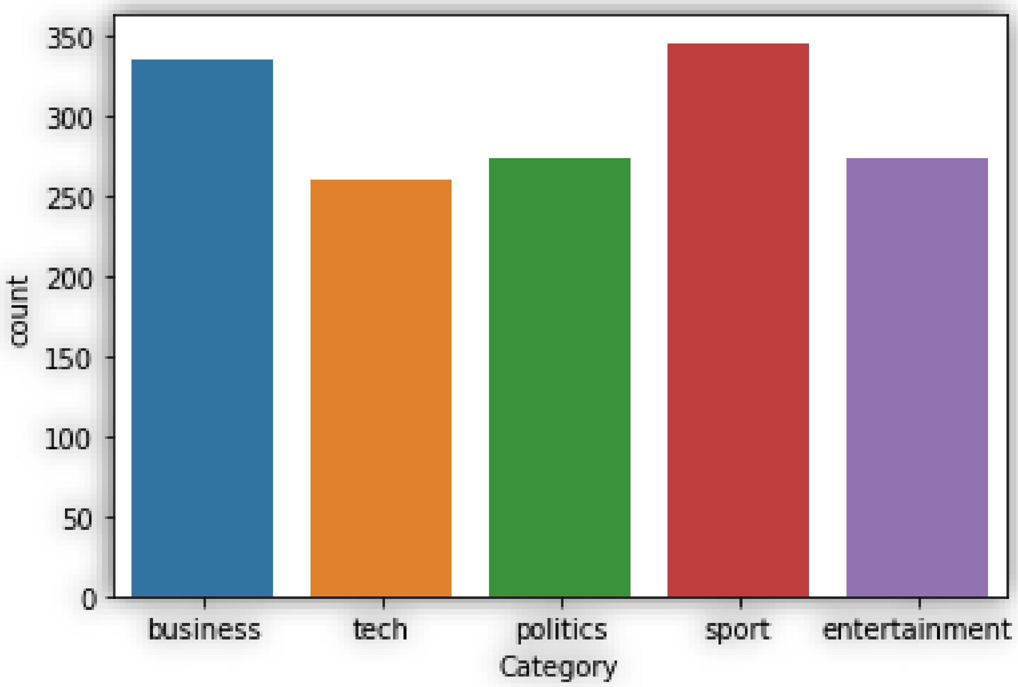
- **Article Id** – Article id unique given to the record
- **Article** – Text of the header and article
- **Category** – Category of the article (tech, business, sport, entertainment, politics)

3.2. Data Cleaning / Data Transformation

Data preprocessing is the process of transforming raw data into an understandable format. It is also an important step in data mining as we cannot work with raw data. The quality of the data should be checked before applying machine learning or data mining algorithms.

3.3. Exploratory Data Analysis

In EDA we have seen various insights from the data so we have selected which column is most important and dropped some of the columns by observing them plotting their heatmap from seaborn library also we done null value managed in an efficient manner and also implemented categorical to numerical transfer of column method .
Here we use nltk library converting paragraph in structured data and removing stopword and converting it into stem word.

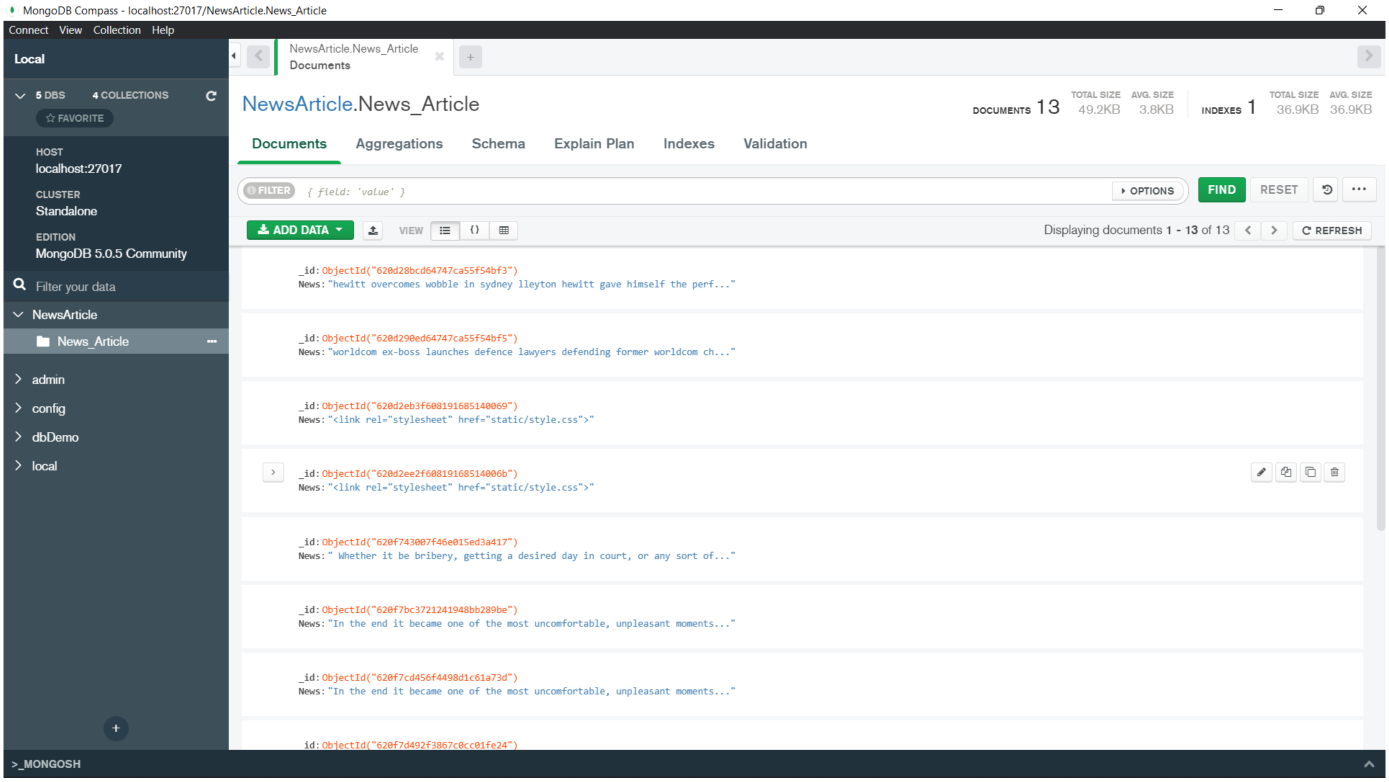


3.4. Data Insertion into Database

Database Creation and connection - Create a database with name passed. If the database is already created, open the connection to the database.

Collection creation in the MongoDB database.

Insertion of data in the table



```
_id: ObjectId("620d28bcd64747ca55f54bf3")
News: "hewitt overcomes wobble in sydney lleyton hewitt gave himself the perf..."

_id: ObjectId("620d290ed64747ca55f54bf5")
News: "worldcom ex-boss launches defence lawyers defending former worldcom ch..."

_id: ObjectId("620d2eb3f608191685140069")
News: "<link rel="stylesheet" href="static/style.css">"

_id: ObjectId("620d2ee2f60819168514006b")
News: "<link rel="stylesheet" href="static/style.css">"

_id: ObjectId("620f743007f46e015ed3a417")
News: " Whether it be bribery, getting a desired day in court, or any sort of..."

_id: ObjectId("620f7bc3721241948bb289be")
News: "In the end it became one of the most uncomfortable, unpleasant moments..."

_id: ObjectId("620f7cd456f4498d1c61a73d")
News: "In the end it became one of the most uncomfortable, unpleasant moments..."

_id: ObjectId("620f7d492f3867c0cc01fe24")
```

3.5. Export Data from Database

Data Export from Database - The data in a stored database to be used for Data Pre-processing and Model Training.

3.6. Data Pre-processing

Data Pre-processing steps could use are Splitting Data into Dependent and Independent Features , Remove those columns which are does not participate in model building Processes , Imbalanced data set handling, Tokenization of sentences, Removing Stopwords in data, Convert whole sentences into TFIDF vector.

3.7 Model Creation / Model Building

After cleaning the data and completing the feature Engineering/ data Per processing. we have done splitted data in the train data and test data using method build in pre-processing file and implemented various Classification Algorithm like RandomForestClassifier and naïve-bias, AdaBoost also calculated their accuracies on test data and train data.

3.8 Hyperparameter Tuning

In hyperparameter tuning we have implemented grid search cv and from that we also implemented cross validation techniques for that.

3.9 Model Dump

After comparing all accuracies and checked all ROC, AUC curve accuracy we have choose multiNomialNB as our best model by their results so we have dumped this model in a pickle file format with the python module.

3.10 Data from User

Here we will collect user’s NewsArticle to predict Category of news.

3.11 Data Validation

Here Data Validation will be done, given by the user.

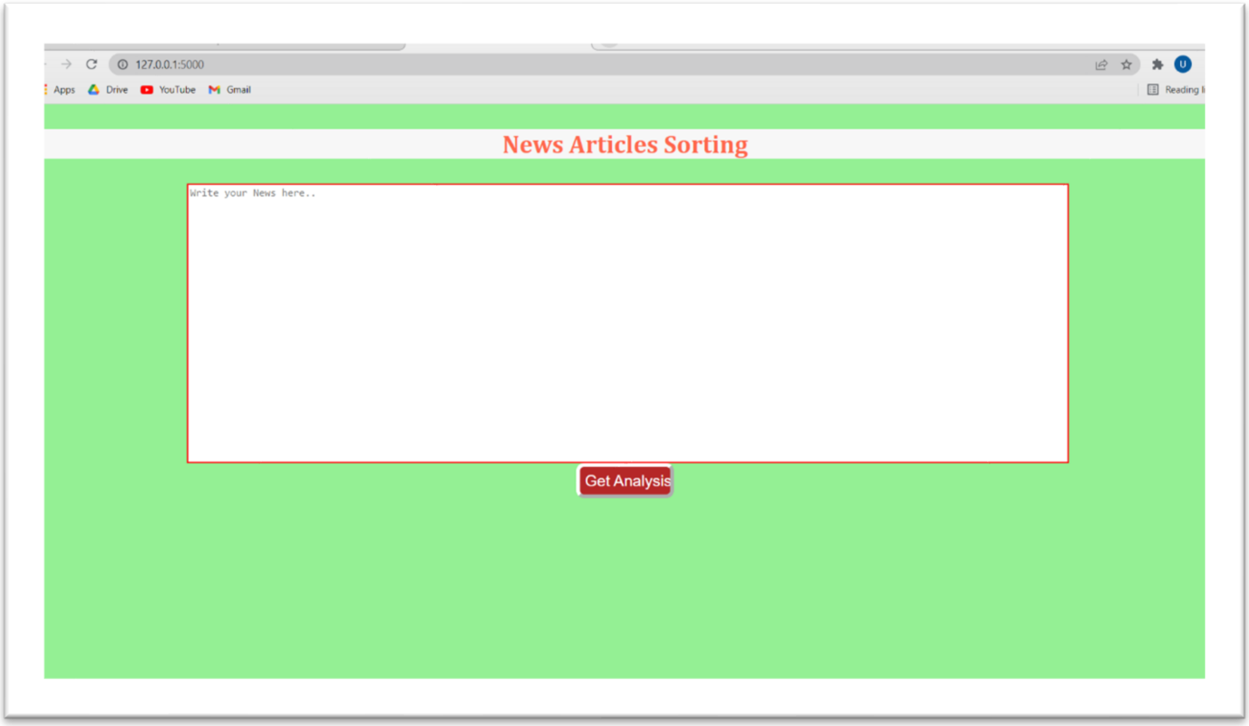
3.12 Model Call for specific input

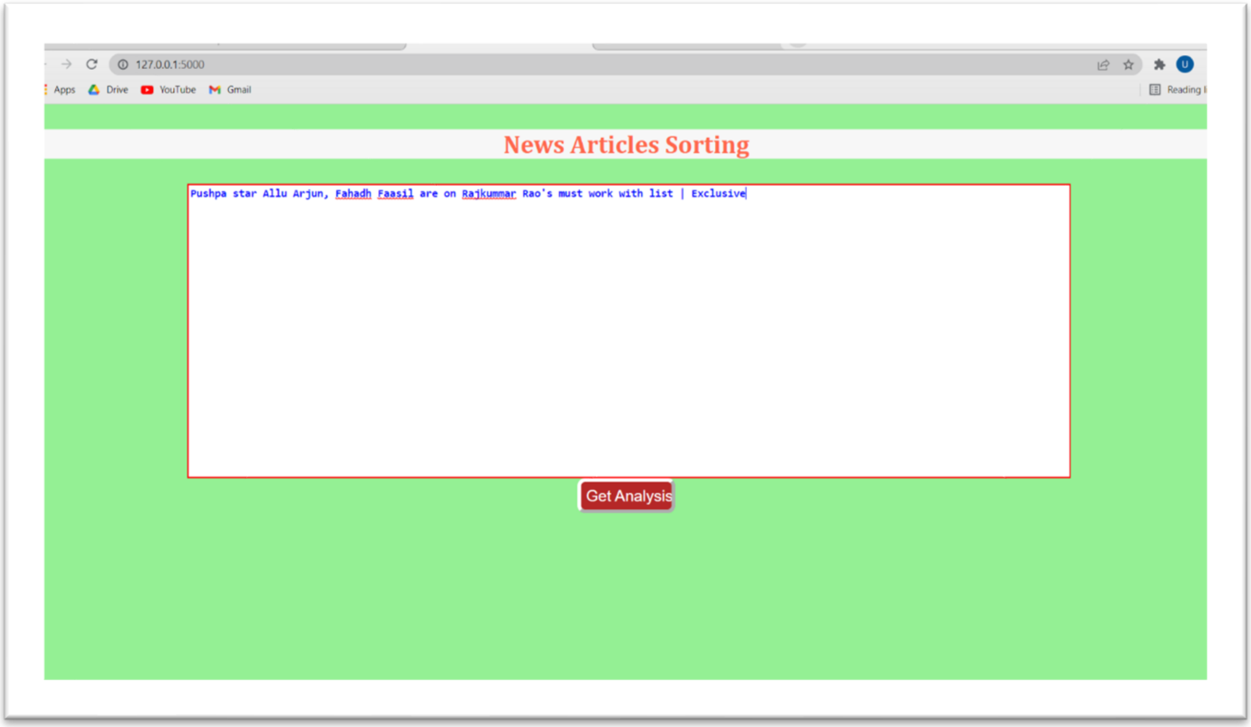
Based on the User input will be throwing to the backend in the variable format then it converted into pandas data frame then we are loading our pickle file in the backend and predicting Category of news as an output and sending to our html page.

3.13 User Interface

In Frontend creation we have made a user interactive page where user can enter their input values to our application. In these frontend page we have made a form which has beautiful styling with CSS and bootstrap. These HTML user input data is transferred in variable format to backend. Made these html fully in a decoupled format.

Input page :-

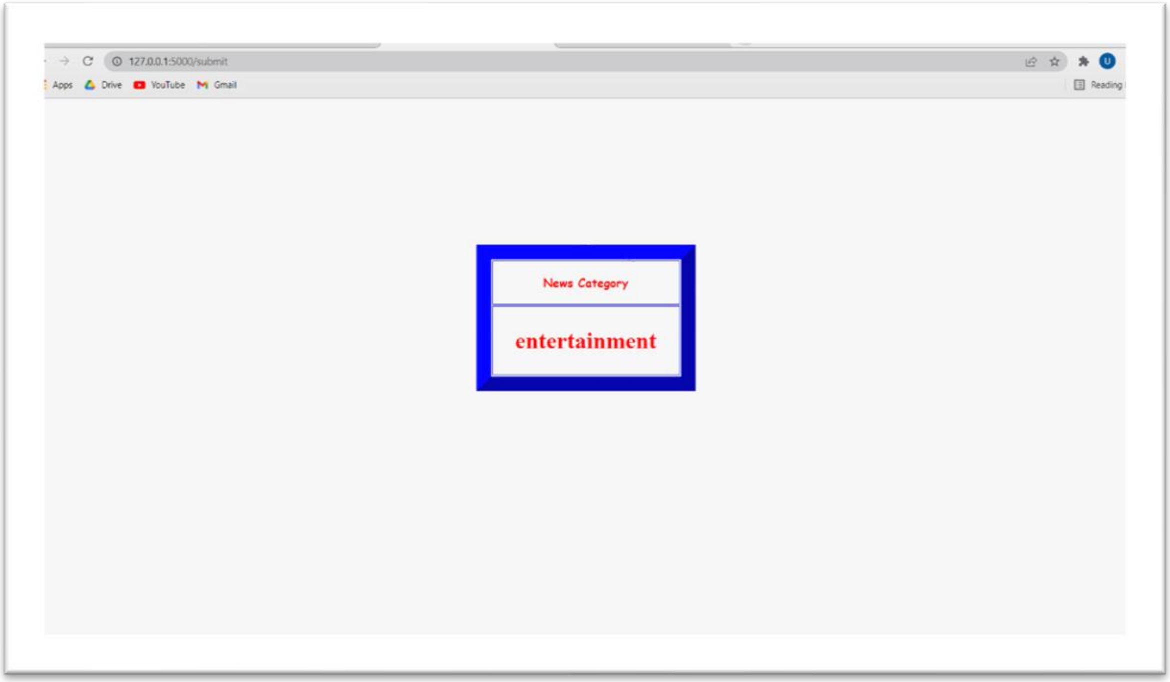




News Articles Sorting

Pushpa star Allu Arjun, Fahadh Faasil are on Rajkummar Rao's must work with list | Exclusive

Output page :-



News Category

entertainment

3.14 Deployment

We will be deploying the model with the help of Azure cloud platforms.

This is a workflow diagram for the News Article Sorting Application ………….

4.Technology Stack

Front End	HTML, CSS, Bootstrap File
Back End	Flask, Pandas, NLP, scikit-learn etc
Database	mongoDB
Deployment	Azure

6.Unit Test Case

Test Case Description	Pre – Requisite	Expected Resulted
Verify whether the Application URL is accessible to the user	1. Application URL should be defined	Application URL should be accessible to the user
Verify whether the Application loads completely for the user when the URL is accessed	1. Application URL is accessible 2. Application is deployed	The Application should load completely for the user when the URL is accessed
Verify whether the User is able to Enter the data	1. Application is accessible	The User should be able to enter the data
Verify whether user is able to successfully Enter all the data.	1. Application is accessible 2. User is Enter all the data	User should be able to successfully Enter all the data.
Verify whether user is giving standard input.	1. Handled test cases at backends.	User should be able to see successfully valid results.
Verify whether user is able to edit all input fields	1. Application is accessible 2. User is logged in to the application	User should be able to edit all input fields

Verify whether the model giving the predicted output.	1.Application is accessible 2. User is able to see predicted output.	User is able to see predicted output.
Verify whether user is presented with recommended results on clicking submit	1.Application is accessible 2. user is presented with recommended results on clicking submit	User should be presented with recommended results on clicking submit