News Article Sorting System

Detailed Project Report

Ashwani Devi

INTRODUCTION

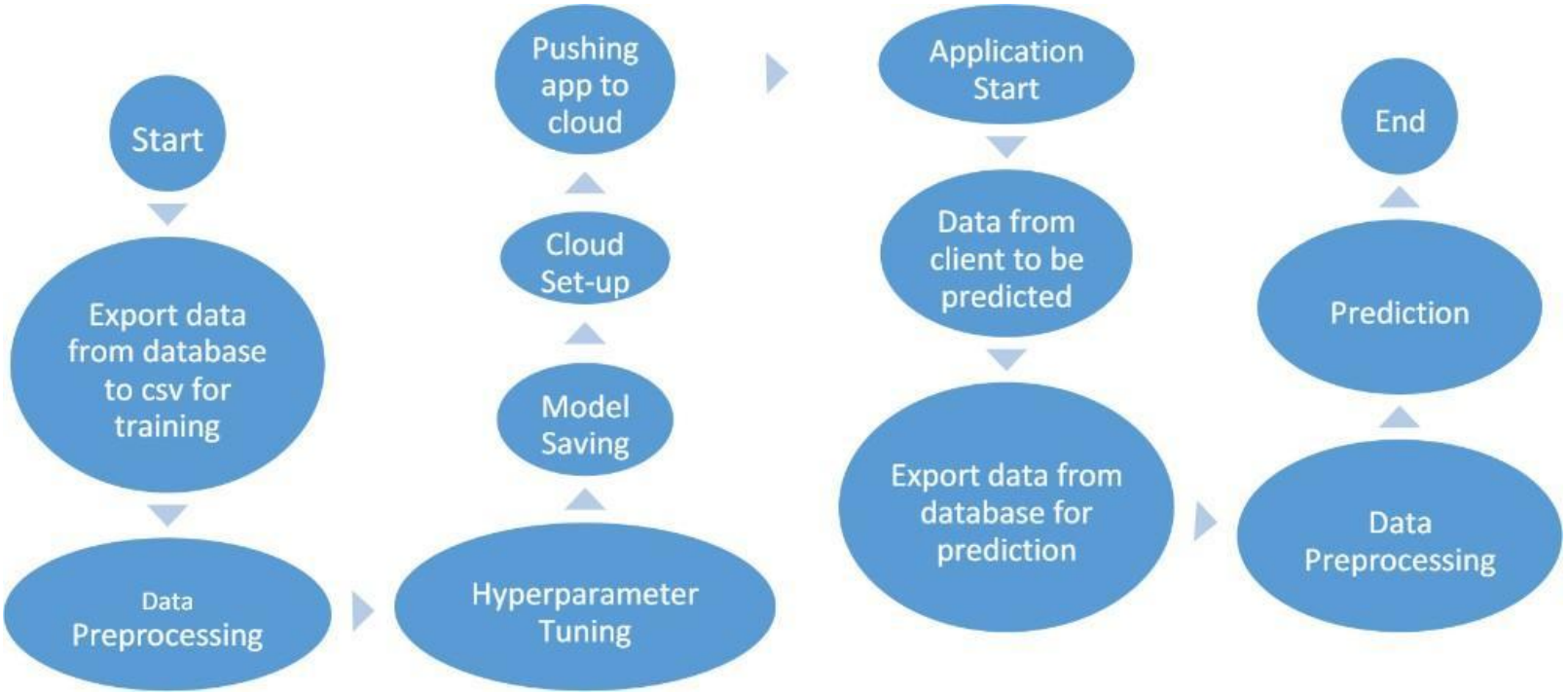
Introduction:
Our **news article sorting system** is a cutting-edge solution designed to revolutionize the way readers consume news content. In the era of information overload, it has become increasingly challenging for users to find and access news articles that align with their specific interests. Our system addresses this issue by introducing a sophisticated category-based sorting mechanism, which **automatically classifies news articles into relevant categories** such as sports, entertainment, politics, technology, and more.

By leveraging advanced algorithms and machine learning techniques, our system streamlines the news consumption experience, allowing users to effortlessly navigate through a personalized news feed tailored to their preferences. This innovative approach not only saves valuable time but also ensures that readers receive the most relevant and engaging content, ultimately enhancing their overall satisfaction and keeping them informed about the topics they care about the most.

OBJECTIVE

The objective of our news article sorting system is to provide users with a streamlined and personalized news consumption experience by implementing a category-based sorting mechanism, enabling effortless navigation and access to relevant news articles based on their individual interests and preferences.

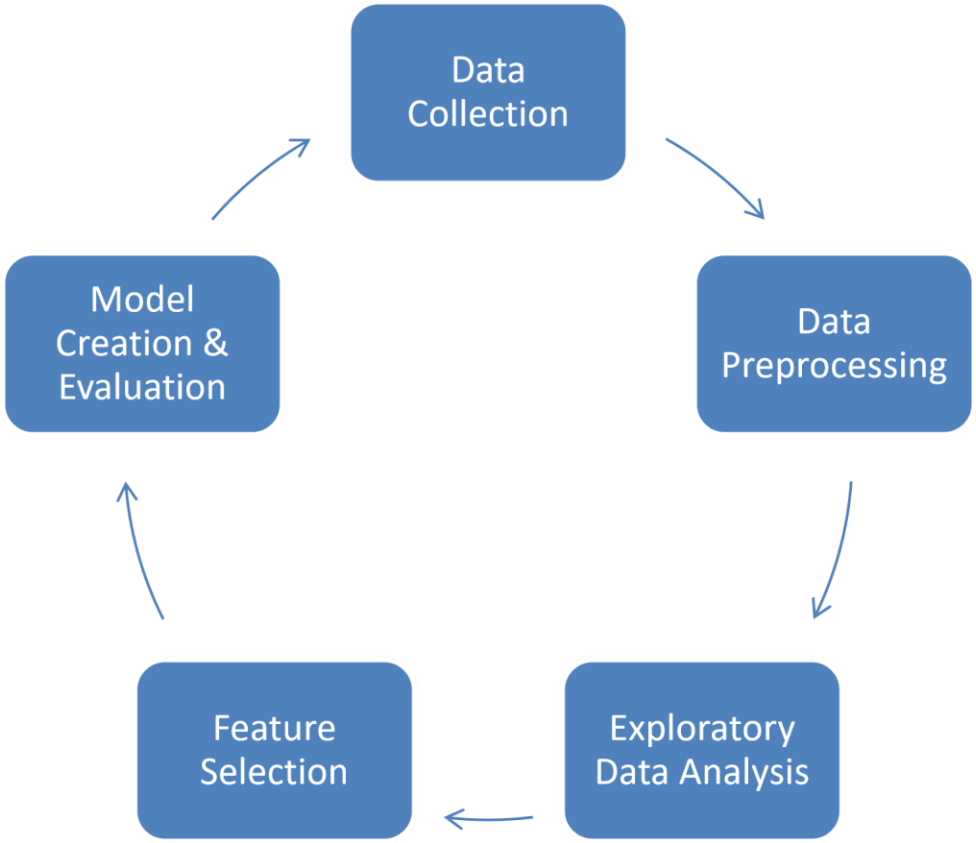
ARCHITECTURE



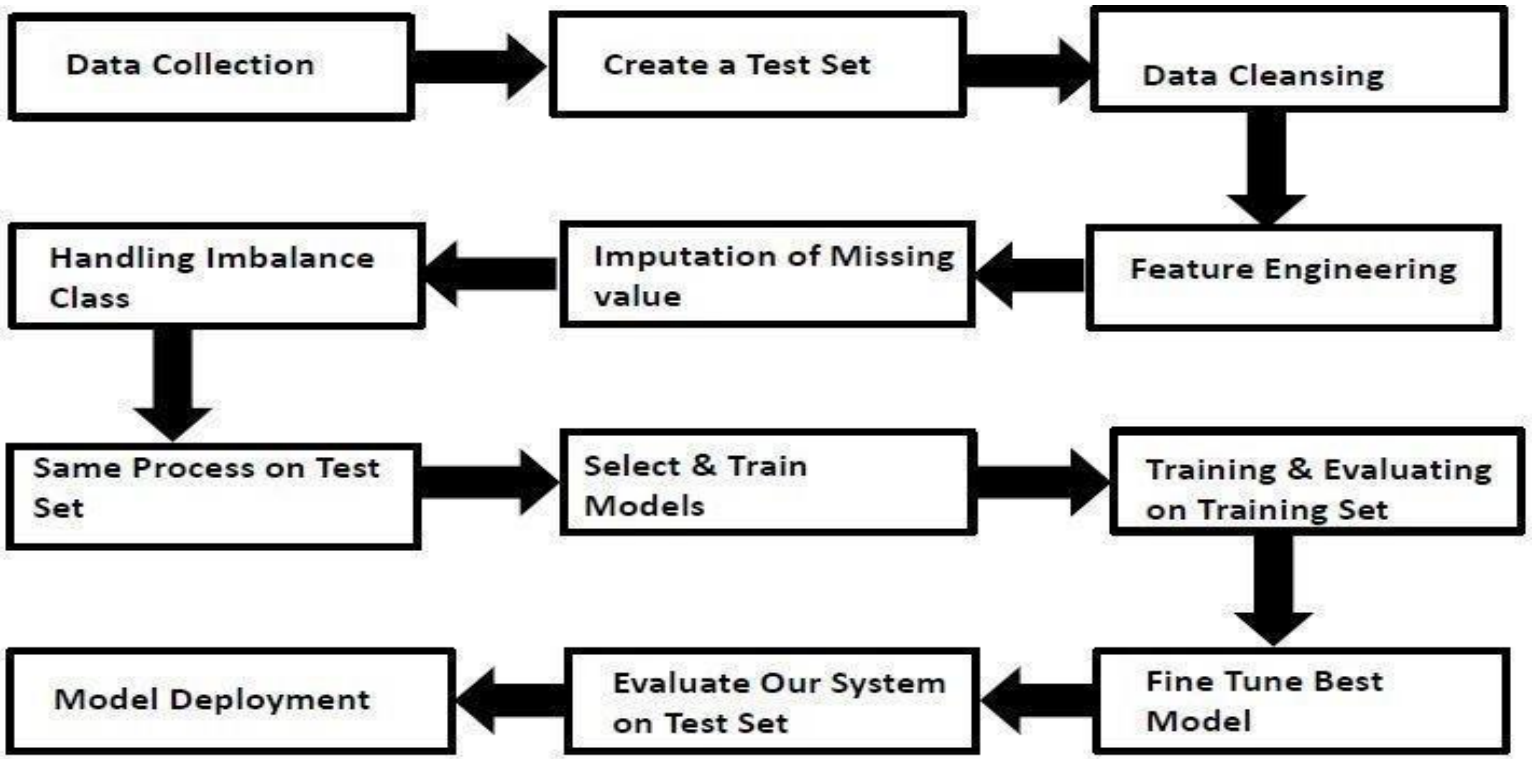
DATASET

- **Dataset includes News Articles for prediction.**
- **Output will be the category of articles respectively.**

Data Analysis Steps



MODEL TRAINING AND VALIDATION WORKFLOW



MODEL TRAINING AND VALIDATION WORKFLOW

Data Collection

- From Kaggle's BBC News Classification Dataset.
- For Data Set: https://www.kaggle.com/c/learn-ai-bbc/data

Data Pre-Processing

- Missing values handling by Simple imputation (Used KNN Imputer)
- Outliers' detection and removal by boxplot and percentile methods
- Categorical features handling by ordinal encoding and label encoding
- Feature scaling done by Standard Scalar method
- Imbalanced dataset handled by SMOTE -Over sampling
- Drop unnecessary columns

MODEL TRAINING AND VALIDATION WORKFLOW

Model Creation and Evaluation

- MultiNomial Naive BAyes was tested and gives good results.
- Hyper parameter tuning was performed.
- Model performance evaluated based on accuracy, confusion matrix, classification report.

MultiNomial Naive Bayes Model

The Multinomial Naive Bayes model is trained by estimating the probabilities of each feature's occurrence for each class in the training dataset. The model assumes that the features are conditionally independent given the class. During training, the frequencies or counts of each feature in each class are calculated. Then, Laplace smoothing or other smoothing techniques can be applied to handle zero probabilities.

The training process involves:

1. Preparing the training dataset with labeled examples, where each example consists of features and their corresponding class labels.
2. Counting the occurrences of each feature for each class in the training dataset.
3. Calculating the probabilities of each feature given each class, usually using the term frequency or term frequency-inverse document frequency (TF-IDF) approach.
4. Applying smoothing techniques to handle zero probabilities and avoid overfitting.
5. Estimating the class priors, which are the probabilities of each class occurring in the training dataset.

After training, the Multinomial Naive Bayes model can be used to classify new instances by calculating the conditional probabilities of each class given the observed features. The class with the highest probability is selected as the predicted class for the new instance.

Model Deployment

- The final model is deployed on Render using the Flask framework.

FREQUENTLY ASKED QUESTIONS

Q1) What is the source of data?

The data for training is obtained from Kaggle.

UCI Machine Learning Repository: https://www.kaggle.com/c/learn-ai-bbc/data

Q2) What was the type of data?

The data was the combination of numerical and Categorical values.

Q3) What's the complete flow you followed in this Project?

Refer slide 7th, 8th and 9th for better understanding.

Q4) After the File validation what do you do with incompatible files which didn't pass the validation?

Files like these are moved to the Archive Folder and a list of these files has been shared with the client and we removed the bad data folder.

Q 6) What techniques were you using for data pre-processing?

- Removing unwanted attributes

- Visualizing relation of independent variables with each other and output variables
- Checking and changing Distribution of continuous values
- Removing outliers
- Cleaning data and imputing if null values are present.
- Converting categorical data into numeric values.

Q 7) How training was done or what models were used?

- First Data validation done on raw data and then good data insertion happens.
- Then Data preprocessing done on the final CSV file received from DB.
- MultiNomial Naive Bayes model is used for predictions, the model is saved.

Q 8) How Prediction was done?

- The client fills the required inputs which is visible on the homepage of API.
- After filling all the required inputs, prediction was made and the client sees the desired result.

Q 9) What are the different stages of deployment?

- After model training and finalizing all models. We created required files for deployment.
- Finally deployed our model over a cloud platform named Render.

Q 10) How is the User Interface present for this project?

- For this project we have made only one type of UI.
- It is for one user input prediction.
- It is very user friendly and easy to use.

THANK YOU