Language Classification Problem

Libraries Used:

Pandas:

- For reading data from csv file.
- Removing rows with null values.

Scikit-Learn:

- Train Test Split for splitting the data.
- Pipeline for pipelining operations on data.
- CountVectorizer and Tf-idfTransformer to transform input to features.
- SVC for Model Building.
- Grid Search for Hyper Parameter tuning.
- F1 Score as metric since the data is imbalanced.

Joblib:

- To save the model as pickle file.
- To Load the model from pickle file.

Data Analysis:

Preprocessing:

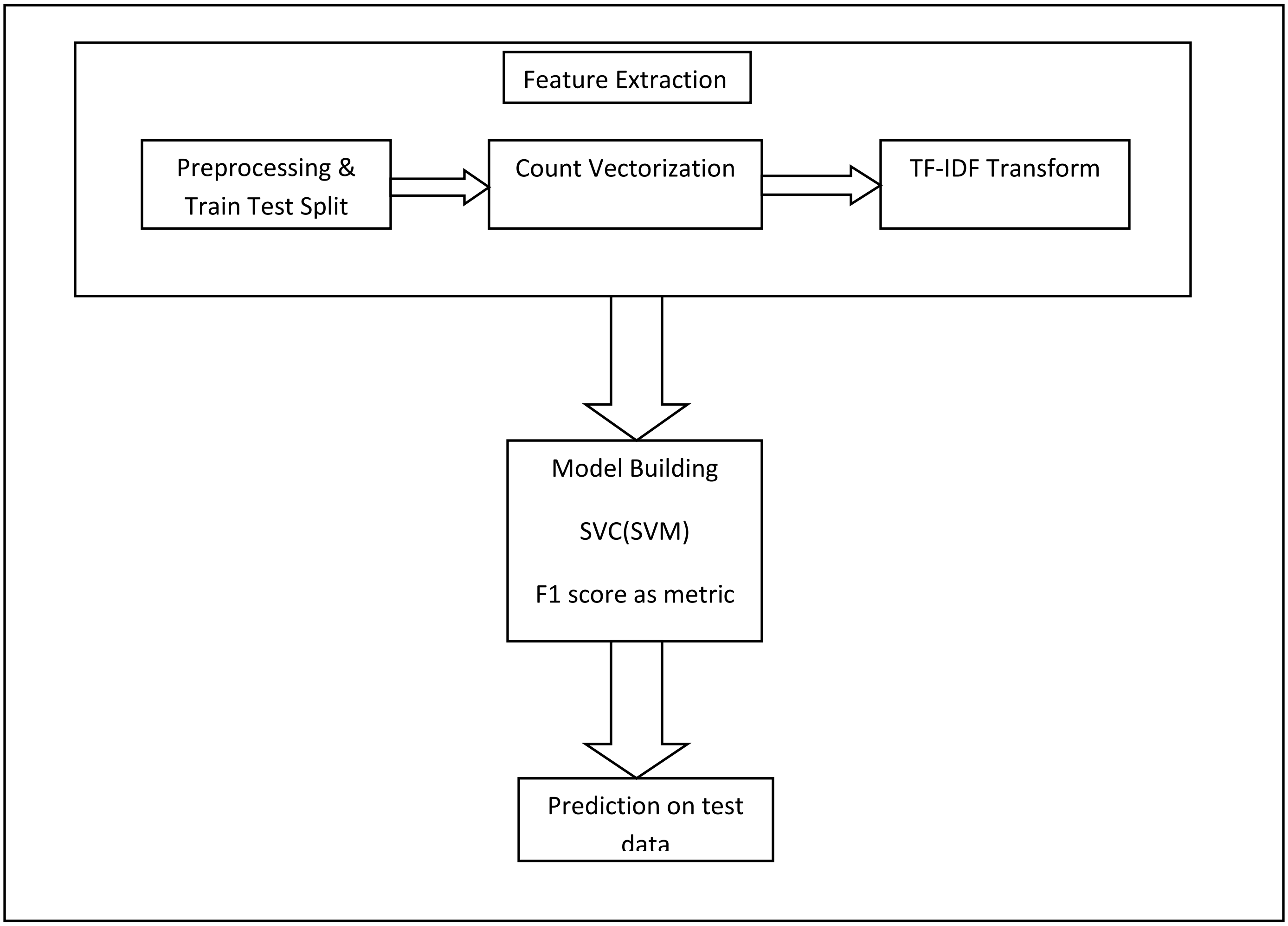
- Input Data Shape is 2839 X 2.
- Shape of Data after removing null values is 2761 X 2.
- Below table shows the information about each class.

Language/Count	Input Data (A)	Empty (B)	(A)-(B) after dropping
Afrikans	671	32	639
English	2077	22	2055
Nederalands	91	24	67
Total	2839	78	2761

Feature Extraction:

- Converting strings of data into CountVectorizer using ngrams.
- Sparse matrices from CountVectorizer are transformed to TF-IDF Matrices.

Model Architecture:



Feature Extraction

Preprocessing & Train Test Split

Count Vectorization

TF-IDF Transform

Model Building

SVC(SVM)

F1 score as metric

Prediction on test data

Training Process:

SVM is chosen as the algorithm to train the model. Grid Search is applied to tune hyper parameters.

Hyper Parameters:

Hyper Parameter	Range/Values
ngram_range	[(1, 1), (1, 2), (1,3)]
use_idf	(True, False)
C	[0.1,1, 10, 100]
gamma	[1,0.1,0.01,0.001]
kernel	["linear", "poly", "rbf", "sigmoid"]

Stratified 5 fold Cross Fold validation is applied. All CPU’s are used for parallel processing.

Best model’s hyper parameters are

{'C': 10, 'gamma': 1, 'kernel': 'sigmoid', ’use_idf': False, 'ngram_range': (1, 2)}

As we are using Cross Validation in Grid Search, Data is split into train and test.

The best Model is used to predict on test set.

F1 Score on each class is `[0.95297806 0.99128751 0.8]`

Confusion Matrix for test Data

Actual/Predicted

Actual/Predicted	Afrikaans	English	Nederland’s
Afrikaans	152	5	1
English	4	512	0
Nederland’s	5	0	12

Bonus Questions:

1. Discuss two machine learning approaches (other than the one you used in your language

Classification implementation) that would also be able to perform the task. Explain how these

Methods could be applied instead of your chosen method.

1. **Class Imbalance problem:**
    a. Using smote technique to up sample the data.
2. **Feature Extraction:**
    a. Use Feature Hashing for converting input to features.
    b. Use Pre-trained embeddings.
    c. Build a vocabulary for each language and apply rule based methods to classify.
3. **Machine Learning Algorithms:** use any of the above feature extraction technique and apply Naive Baye’s Algorithm, XGBoost (Boosting) Ensemble of SVM, Naive Baye’s, and XGBoost.

2. Explain the difference between supervised and unsupervised learning.

Supervised Learning: When there is labeled data, we try to map a function from input features to labeled data. Supervised Learning is divided into Regression and Classification problems depending upon the labeled data type whether it is numeric or categorical.

Unsupervised Learning: It is applied when there is unlabeled data. Algorithms use the input features to discover underlying structures in the data. Unsupervised learning is divided into Clustering and Association problems. Clustering is grouping data into different groups. Association problems are how two things associated are.

3. Explain the difference between classification and regression.

Classification means the target variable is categorical. We use Cross Entropy or logistic loss for solving classification problem.

Regression means the target variable is continuous. We use RMSE, MSE as loss for solving Regression problems.

4. In supervised classification tasks, we are often faced with datasets in which one of the classes is much more prevalent than any of the other classes. This condition is called *class imbalance.* Explain the consequences of class imbalance in the context of machine learning.

Suppose we have a cancer dataset in which most of the cases are negative. We have to predict whether a patient contracts cancer or not. Most of the target variables will be negative. Machine Learning Algorithms learn from data. So what most of the data says, the algorithm maps a function from input values to output since we are trying to reduce the loss. As most of the data is negative there is a function maps every data point to negative class. This becomes a serious issue, if someone gets false negative (contracts cancer but model predicted negative).If patient is predicted positive he can be treated early. In this type of problems accuracy should not be considered as a metric.

5. Explain how any negative consequences of the class imbalance problem (explained in question 4) can be mitigated.

Choose a metrics like Recall, Precision, F1 Score as metric to know the quality of the model. Metric selection depends upon the problem. Tune the hyper parameters so that a good score is achieved.

Smoting is a technique by which we can reduce the negative consequences of class imbalance i.e. Up-sampling the data, so that there is no class imbalance.

6. Provide a short overview of the key differences between deep learning and any non-deep learning method.

Deep Learning: Deep Learning is used to map more complex data. We add more Hidden layers to the model and train the data on problems like Object Detection, Semantic Segmentation, Instance Segmentation, and Deep Reinforcement Learning. Automatic Feature Engineering happens in Deep Learning. Huge amount of data is required when compared to non-deep learning method. High Compute power is required. On the merits side we have high performance when compared to non-deep learning method

Non-deep learning method: Feature Engineer should be done manually. Requires less compute. Performance is limited even the data available is huge. These methods are useful when data size is good, compute power is good. Performance doesn’t go up even data is high.

On Fun Side :D Deep Learning means