TAXONOMY CLASSIFICATION

ACHAL RAJYAGURU
CHAROTAR UNIVERSITY OF SCIENCE AND TECHNOLOGY, GUJARAT

PROBLEM STATEMENT

- Taxonomy Classification - The task is to predict the tags (a.k.a. keywords, topics, summaries) of a question, given only the question text and its title. The dataset contains content from disparate stack exchange sites, containing a mix of both technical and non-technical questions.

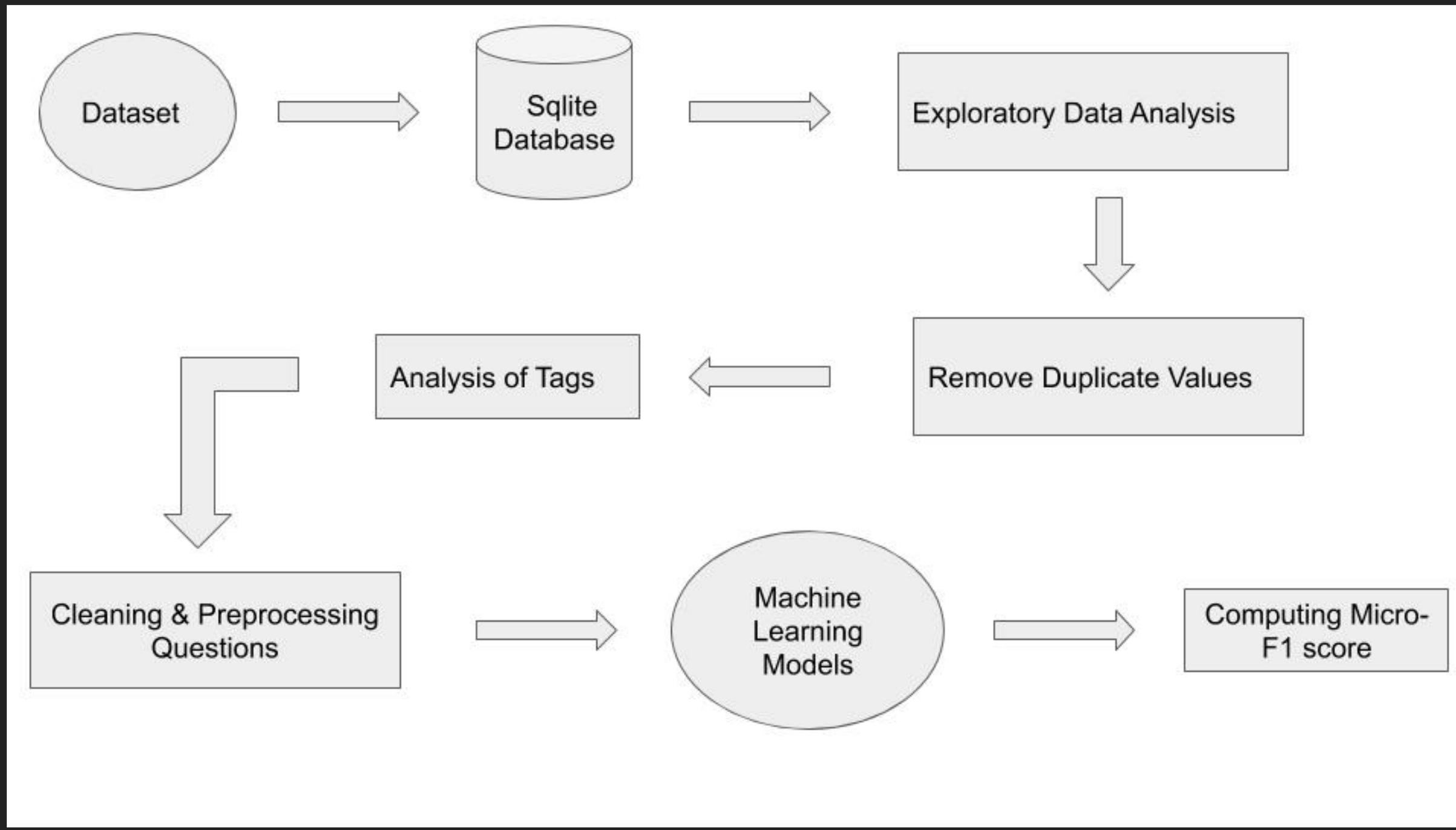
SOLUTION APPROACH

- The dataset consists of 6M entries (61,20,000) which consists of ‘Title’ , ‘Body’ and ‘Tags’ of the questions posted on StackOverflow.
- Here, ‘Title’ and ‘Body’ are the independent attributes whereas ‘Tag’ is the dependent variable.
- The dataset is quite messy and contains a lot of redundancy which needs to be removed using Preprocessing techniques. The dataset is also highly imbalanced out of a total 42,000 tags, the top 15 % (5,500) tags cover almost 99.04 % questions (explained in detail in later sections). Hence, we also need to try to reduce the bias nature of dataset

ACCURACY METRICS

- " Mean F1 score" (Micro F1 score) should be used to evaluate our model due to imbalanced dataset problem
- Micro F1 score takes into account the contribution of each tag, whereas Macro F1 score averages the result of the F1 score of every individual tag

BRIEF OVERVIEW



SOLUTION APPROACH （STEP 1)

- 1.0 : Exploratory Data Analysis
    - 1.1 : Data Loading and Cleaning
    - 1.1.2 : Using Pandas with ‘Sqlite’ to load data
    - 1.1.3 : Counting number of rows
    - 1.1.4 : Checking for duplicates
    - 1.1.5 : Creating a new dataset with no duplicates

SOLUTION APPROACH (STEP 2)

- 1.2 : Analysis of Tags
  - 1.2.1 : Total number of unique tags
  - 1.2.2 : Number of times a Tag appeared / Plotting Charts
  - 1.2.3 : Tags per question
  - 1.2.4 : Most Frequent Tags
  - 1.2.5 : Word Cloud, Top 20 Tags

SOLUTION APPROACH (STEP 3)

- 1.3: Cleaning and Pre-processing of Questions
  - (1) Sample 1M data points (due to limitation in computational power)
  - (2) Separate Code snippets from body
  - (3) Remove special characters from question title and description (not from code)
  - (4) Remove Stopwords (Except ‘C’)

CONTINUED..

- (5) Remove HTML Tags
- (6) Convert all characters to small letters
- (7) Use SnowballStemmer to stem words

SOLUTION APPROACH (STEP 4)

- 2.0 : Machine Learning Models
    - 2.1 : Converting Tags for Multi-label problems
    - 2.2 : Split the dataset
    - 2.3 : Featurizing data, Using n-gram model in range 1 to 3
    - 2.4 : Applying Logistic Regression with One vs Rest Classifier
    - 2.5 : Modelling with less data points (0.5M) and giving more weight to title and 500 Tags only

CONTINUED

- 2.5.1 : Pre-processing Questions (same as before) just giving more weightage to ‘Title’
- 2.5.2 : Featurizing data with TF-Idf Vectorizer
- 2.5.3 : Applying Logistic Regression with One vs Rest Classifier

ARRIVING AT FINAL SOLUTION

- Using Binary Relevance Method
- Applying Logistic Regression with One vs Rest Classifier

PROS AND CONS OF SOLUTION

- CONS :
  - (1) Can only sample 1M data point due to limitation in computational power
- PROS :
  - (1) Using Sqlite Database for faster analytics as compared to using only pandas
  - (2) Using the top 15% tags which cover almost 90% question
  - (3) Using Binary Relevance Method with One vs Rest Classifier and using only the top 15% tags hence making it more computationally efficient

THANK YOU!