CSC 215-01 Artificial Intelligence (Fall 2018)

Mini-Project 1: Yelp Business Rating Prediction using Pandas and Sklearn

Megha Mathpal - 219695880

John Cyrus Kandikatla - 219720697

Abstract:

In this project, we will use regression and classification models to predict the business star rating of yelp by using the review count and all reviews of that business. We implement Linear Regression for regression problem and Logistic Regression, Nearest Neighbor, Support Vector Machine, Multinomial Naive Bayes for classification problem in scikit learn.

Introduction:

Yelp is a crowd sourced local business review site where a user can submit a review regarding that business's product or service. Based on these user reviews and the count of reviews a business has got our system and predict the overall star rating for that specific business. To obtain this outcome we train our models by applying different ML models that helps us predict the star rating of that business.

Data source :

We got the data from: https://www.yelp.com/dataset/download This set includes information about local businesses in 10 metropolitan areas across 2 countries with millions of reviews. The dataset contains several json files from which we are using business and review json files for the prediction of star rating.

MACHINE LEARNING MODELS IN USE

The algorithms that have been used in our project are:

1) **Linear Regression**

Linear Regression is a machine learning algorithm based on supervised learning where the target prediction is based on independent variables or a predictor variable X. Based on that X we predict the quantitative value of Y i.e. the dependent variable. We use scikit learn to import the linear regression model. We fit the model on the training data and predict the values for the testing data. We use R2 score and mean square error to measure the accuracy of our model.

Analysis:
Mean squared root error - 0.55
variance score(r2)- 0.60

2) **Logistic Regression**

Logistic regression is a binary classification problem in supervised learning. It is used to predict binary outcomes for given set of independent variables and the dependent variable's outcome is discrete. The outcome can be either 0 or 1.

Analysis:
F1 score – 0.37
recall -0.44
precision -0.40

3) KNN(***k*-nearest neighbors**)

k-NN is a type of instance-based learning, or lazy learning, where the function is only approximated locally and all computation is deferred until classification. In *k-NN classification*, the output is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common among its *k* nearest neighbors (*k* is a positive integer, typically small). If $k = 1$, then the object is simply assigned to the class of that single nearest neighbor.

Analysis:

F1 score – 0.47
recall -0.48
precision -0.49

4) Support Vector Machine

Given a set of training examples, each marked as belonging to one or the other of two categories, an SVM training algorithm builds a model that assigns new examples to one category or the other, making it a non-probabilistic binary linear classifier.

Analysis:

F1 score – 0.70
recall -0.71
precision -0.70

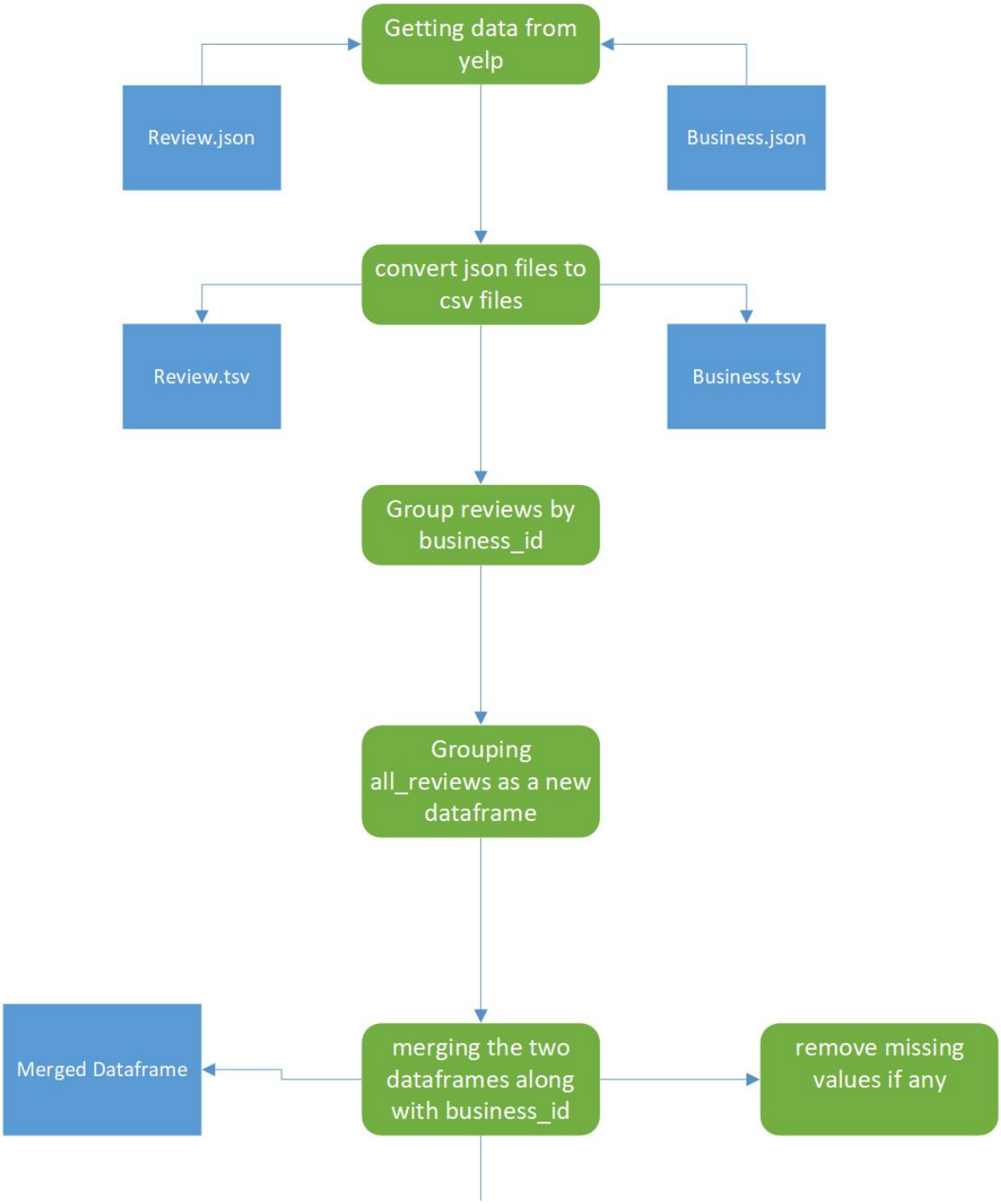
5) Multinomial Naive bayes

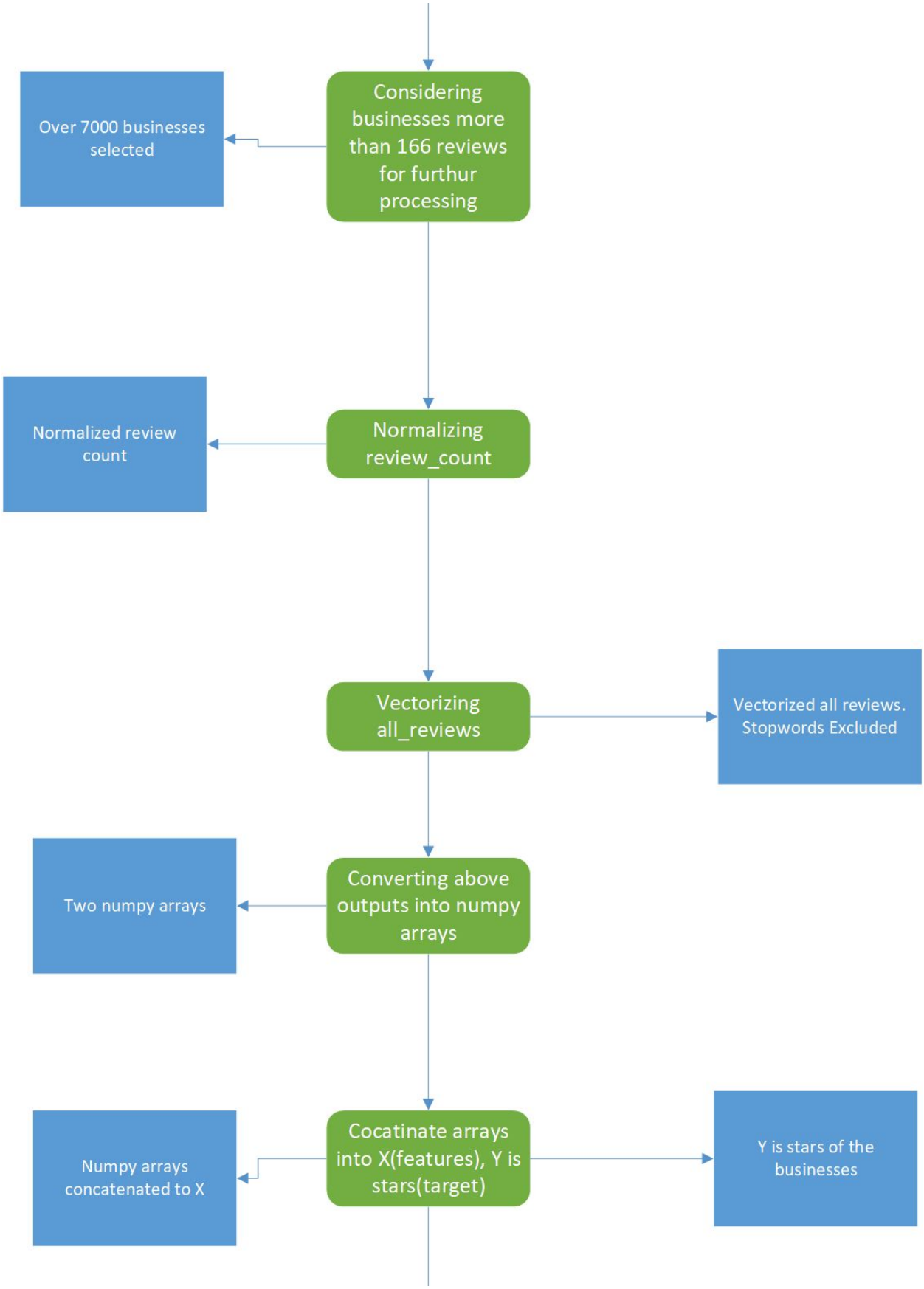
MultinomialNB implements the naive Bayes algorithm for multinomially distributed data, and is one of the two classic naive Bayes variants used in text classification where the data are typically represented as word vector counts, although tf-idf vectors are also known to work well in practice.

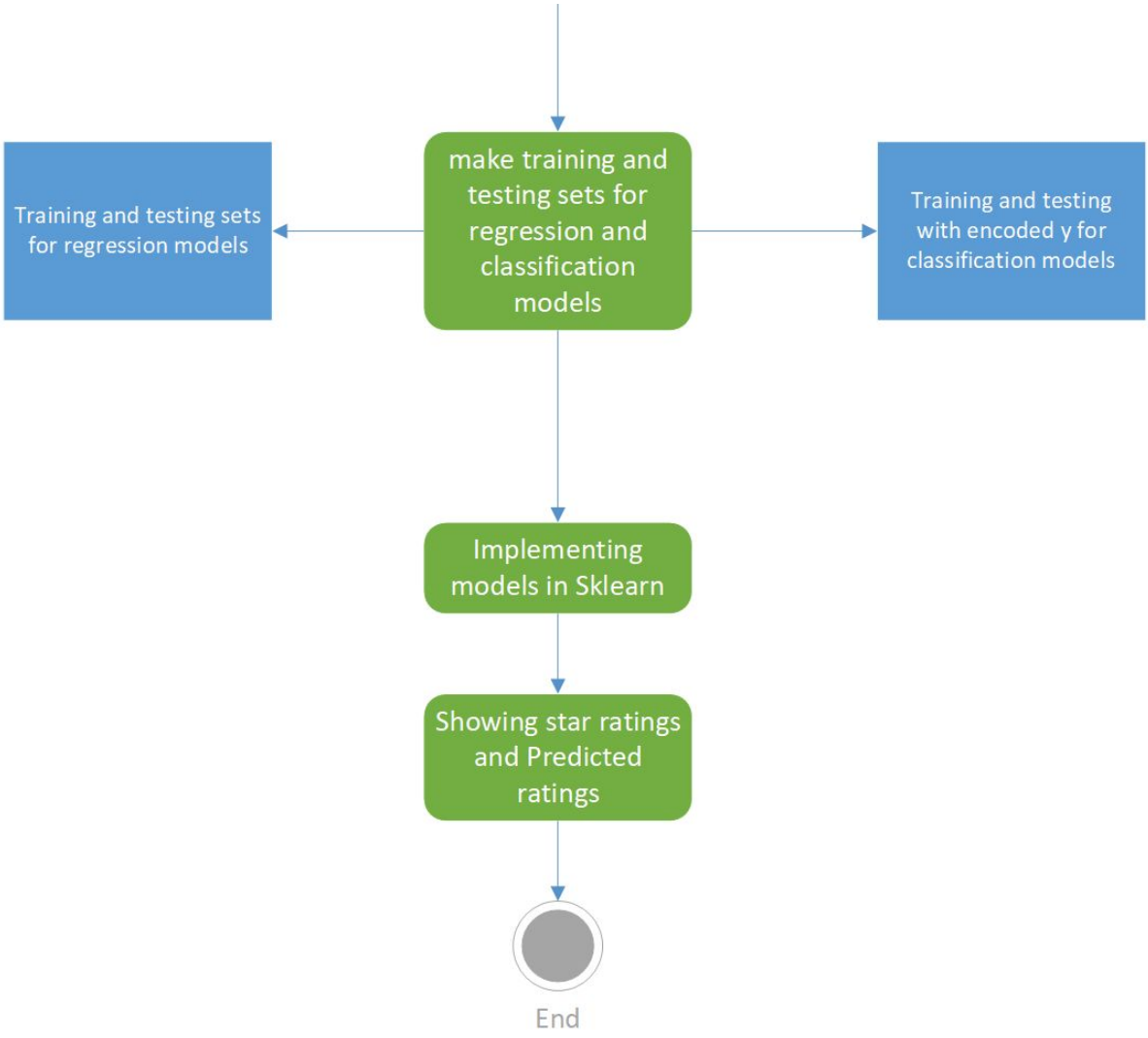
Analysis:

F1 score – 0.37
recall -0.44
precision -0.40

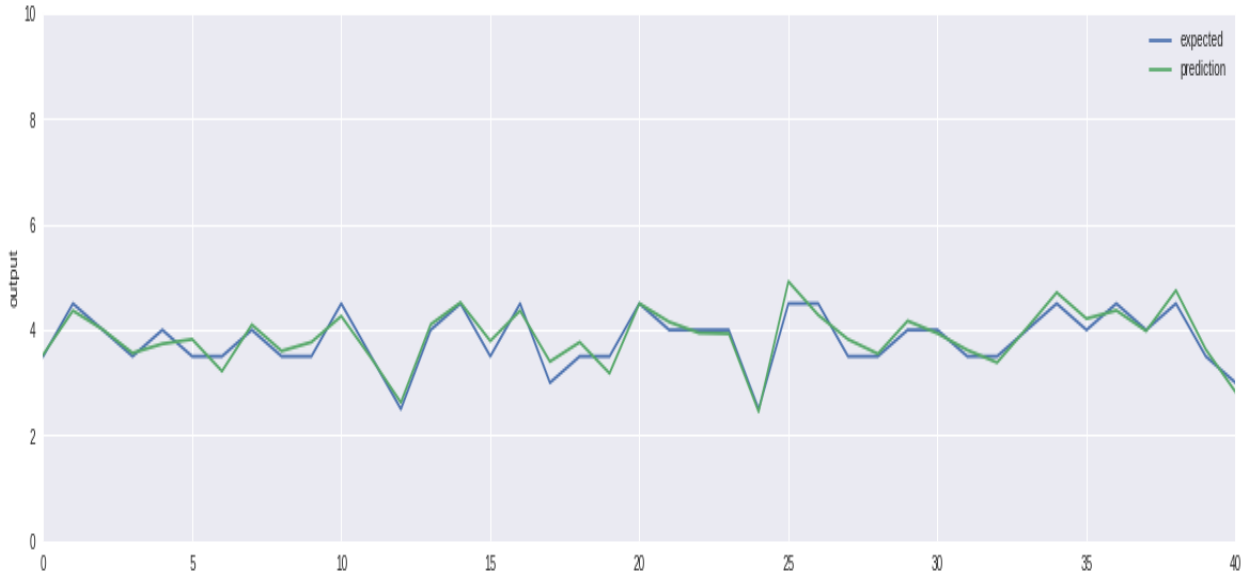
Model Flow Chart/ Methodology:




Considering businesses more than 166 reviews for furthur processing
Over 7000 businesses selected
Normalizing review_count
Normalized review count
Vectorizing all_reviews
Vectorized all reviews. Stopwords Excluded
Converting above outputs into numpy arrays
Two numpy arrays
Cocatinate arrays into X(features), Y is stars(target)
Numpy arrays concatenated to X
Y is stars of the businesses


make training and testing sets for regression and classification models
Training and testing sets for regression models
Training and testing with encoded y for classification models
Implementing models in Sklearn
Showing star ratings and Predicted ratings
End

Additional features(Data visualization)
Linear regression chart regression:



Logistic regression chart regression:

TASK DIVISION AND PROJECT REFLECTION

Task Division :

Megha Mathpal :

1. Data cleaning
2. Data pre-processing
3. Implementation of linear regression
4. Implementation of KNN
5. Vectorizing
6. Analyzing data and Creating datasets

John Cyrus Kandikatla:

1. Analyzing data and attributes
2. Implementation of SVM
3. Implementation of logistic regression
4. Implementation of MNB models
5. Normalization
6. Visualization

Project Reflection :

We learned a lot about how to process and clean data with such large data set. As this is our first time with python we did a lot of trial and error to get the best output. Our machine gave us challenges. We were facing problem in reading the huge json files. We managed it by using google colab. We learned to use Scikit Learn library and Numpy. As a team, it was a good learning journey to get a good understanding of the topics taught in class.