Dan Burke

Dburke01@syr.edu

Homework 7

MNB SVM

06/18/2023

|  |  |
| --- | --- |
| Homework 7 MNG, SVM | |
| **Introduction** | Sentiment analysis has a multitude of applications from analyzing classical literature, to Twitter Tweets, to restaurant and movie reviews. This report attempts to utilize sentiment prediction within the context of movie reviews. There are two datasets, positive reviews and negative reviews, which are combined and then ingested into a Support Vector Machine (SVM) and Naïve Bayes model.  This report intends to apply the Naïve Bayes and SVM approach to movie review text data provided within class resources. This data possesses file name and movie review “text” features. Two separate datasets have been provided, one of positive reviews and one of negative reviews. Sentiment labels have been generated for each document relative to the original dataset it was sourced from.  Naïve Bayes is a probabilistic machine learning process utilizing the Bayes’ theorem. This process assumes that all features (columns) of the data set are conditionally independent. Within the scope of this report the features refer to induvial words; all individual words contained within the dataset.  Support Vector Machines (SVMs) are binary classifiers with the goal to find an optimal “hyperplane” within a multidimensional space. |
| **Analysis** | data preparation The original data provided consisted of two folders (positive and negative), containing individual text files of movie reviews. It was first necessary to load/open both directories, then iterate through each file parsing its data and storing that data in a form which could then construct a data frame.  A custom method was created to accomplish this task, which returned a dictionary of keys corresponding to the file name and values corresponding to the movie review text. Each key-value pair represents a single document, movie review. Each document of the populated positive and negative data frames was then given sentiment labels corresponding to the original source of the data.    Figure 1 File Parsing and Aggregation      Figure 2 Transformation from Dictionary to Data Frame    Figure 3 Label Assignment    Figure 4 Resulting Data Frame  Once the data had been labeled it was then necessary to concatenate both data frames. Naïve Bayes and Support Vector Machines require all features to be numerical, because of this it was necessary to change the “label” feature to a categorical data type.    Figure 5 Data Frame Concatenation, "Label" feature type reassignment   models and methods AND Parameters **Naïve Bayes**  This effort’s first focus is to generate a classifying model via the Naïve Bayes Classifier (leveraging SkLearn). At this point, the data had been aggregated, appropriately labeled and is ready for the removal of stop words, count vectorization and fitting to a model.  Several custom methods were written to accomplish these tasks allowing the most flexible and reusable codebase.    Figure 6 Custom Method for Count Vectorization, Stop Word Removal, Generation of "x" variable    Figure 7 Custom Method to Output results of a Naive Bayes Classifier with a Test Train Split of 70/30  **Support Vector Machine (SVM)**  Data preparation for ingestion into a Support Vector Machine model took a similar path to that of Naïve Bayes except for tokenization and lemmatization.  For this effort it was necessary to iterate throughout each document within the aggregate dataset, tokenizing the text, lemmatizing and removing default stop words (immediate figure below).    Figure 8 Tokenization, Lemmatization, Stop Word Removal  Upon completion it was then prepared for count vectorization, segmented into test and training subsets.    Figure 9 SVM Count Vectorization, Test Train Split    Figure 10 Fitting To SVM Model    Figure 11 Generating Prediction |
| **results** | technical results . Resulting Multinomial Naïve Bayes Model Prediction Accuracy:    Figure 12 No reduction in Data Set Size    Figure 13 No reduction in Data Set Size    Figure 14 After Limiting Positive and Negative Reviews to 5,000 Documents Each    Figure 15 Confusion Matrix After Limiting Positive and Negative Reviews to 5,000 Documents Each    Figure 16 SVM Model Accuracy Score |
| **conclusions** | The Multinomial Naïve Bayes approached proved to produce results of exceptional accuracy ~86.2%. With expansion of stop words combined with tactful removal of punctuation, this may be increased. Due to the size of the dataset, running the project code locally and then through a virtual machine caused repeated crashes. It became necessary to reduce the size of each data set (positive, negative) to n=5000. This reduction in size resulted in an increased Naïve Bayes accuracy of ~88%.  The Support Vector Machine (SVM) implementation performed unsatisfactorily ~50% accuracy score. This can be attributed to multiple aspects of how the data has been prepared and cleaned. SVM’s are ideal for sentiment analysis as they are well suited to handle high-dimensional feature spaces; textual sentiment analysis is just this; which suggests that there is a flaw within the data cleaning and preparation.  SVM models also are not best suited for large datasets, initially this dataset was comprised of approximately 25,000 documents (roughly 50/50 Positive/Negative labels). This proved to be an issue with performance as the development environment both locally and virtually (leveraging Google Colab) consistently crashed. Even within a virtual development environment with more than 25 GB of RAM; it proved to be consistently unstable.  Naïve Bayes proved to perform adequately while SVM requires further action to optimize the data for SVM. This effort cannot conclusively determine if SVM is not satisfactory for sentiment analysis until further efforts in the way of data preparation and data cleaning have been made. |