IDS 575 Statistical Models and Methods

Term project- Final Report

Project Title: Automated product score generation based on sentiment analysis on customer reviews

Contents:

Introduction to Dataset

Amazon product data is a collection of customer reviews for products belonging to specific categories carried by the e-commerce site. The collection has 24 product groups with a total of 142.8 million reviews collected over the timespan of May 1996 - July 2014.

Data Source:	Julian McAuley, UCSD
Data Description:	The data contains reviewerID, productID(ASIN), reviewerName, helpfulness, reviewText, overallScore, summary, unixReviewTime and reviewTime

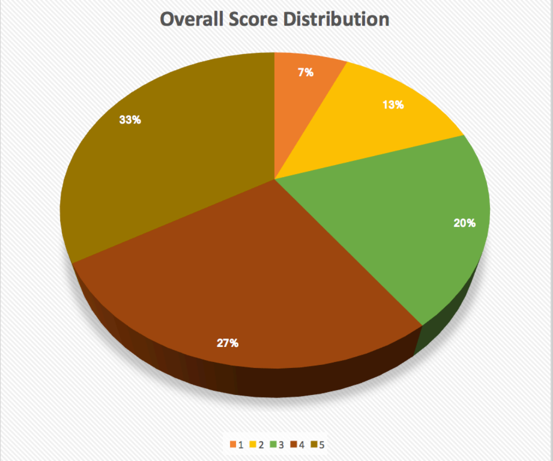
The product group in context is Clothing, Shoes and Jewelry which has 278,677 reviews in total. For computational ease, a smaller sample set will be used for feature generation and model building.

Feature Generation From Raw Text

In order to learn the sentiment associated to each of the overall score, I chose to sample my data such that each score category has equal number of records. The original distribution of scores in the 278,677 reviews is given by the chart. The sample set contains 5000 reviews equally split for each score in order to learn features without bias.

The first step to generate features from raw text is to pre-process user generated text. Using the ***TfidfVectorizer*** method from python's inbuilt package for Natural Language Processing (NLP)- ***sklearn.feature_extraction.text,*** a bag-of-words termed as vocabulary set was generated. The ***TfidfVectorizer*** method performed the following steps on the raw review text.

- Tokenization
- Stemming
- Removing Stop words
- Removing Punctuations
- N-grams



With this set of processed bag of words, the features for a model to learn are generated by calculating the Term Frequency – Inverse Document Frequency values for each token in the vocabulary set. This value is a measure of how important is the word within a document as well as among a corpus of documents (collection of 5000 reviews is a corpus where each individual review is a document).

The sub-sample of 5000 reviews are shuffled and split into training (70%) and test (30%) data. The feature set generated by the ***TfidfVectorizer*** is a matrix of values having each token in columns and each individual review as rows.

REGRESSION MODEL TO PREDICT SENTIMENT

The central idea of this analysis is to standardize and automate product score generation which will increase the reliability of the overall score value. One way of approaching this idea is to analyze the sentiment of each customer review and generate a score that best represents the user generated overall score.

Hypothesis: Sentiment analysis of customer reviews is a direct measure of the overall product score.

Method: Build a logistic regression model that identifies sentiment associated to each review and represents the results on a scale of 1-5 (similar to overall score). The sentiment score is compared against the original overall score to measure the model performance and parameter tuning.

The critical assumption for the central idea to hold good is that the overall score in training data is a standardized representation of the scores with very minimal subjectivity.

Though there are many ways like SVM, Naïve Bayes methods to predict sentiment scores based on text features, Logistic Regression is the simplest and elegant method for its easy interpretability and coefficient learning. Since our analysis is basically a classification problem with 5 classes, Logistic Regression works best.

The ***LogisticRegression*** method from python's inbuilt package ***sklearn.linear_model*** is used to fit a model using the training features and labels. The training data consisted of a collection of 3500 reviews and their corresponding overall score. This model is then run with 1500 review features of the test data to predict the sentiment score that is considered as the standardized equivalent of overall score. A classification report is built using the actual scores and predicted scores of the test data to interpret the model performance.

PERFORMANCE – SENTIMENT ACCURACY

Model evaluation for sentiment analysis differs from other general methods. Looking at the overall accuracy of model will not give us the right insights in this case. The default choice of metrics while measuring the performance of any model that analyses sentiment is PRECISION which is the measure of how often is the sentiment score correct. Precision is also termed as model accuracy as its measurement is straightforward. But there is RECALL which is equally important to find how many records have been classified with their appropriate sentiment out of all the records in the corpus. The metrics that are focused in this analysis are:

Precision: True Positives / (True Positives + False Positives)

How many selected items are relevant? [w.r.t sentiment score for a review]

Recall: True Positives / (True Positives + False Negatives)

How many relevant items are selected?

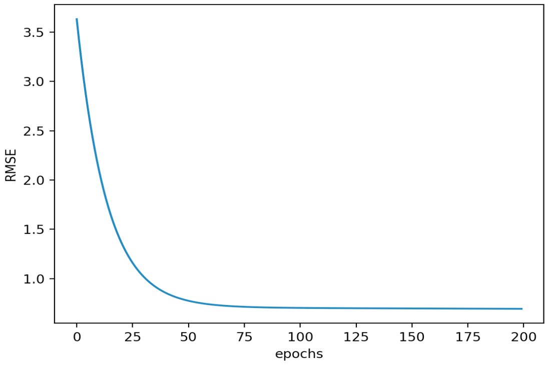
F1 Score: A combination of both Precision & Recall. This score is between 0 & 1 where 1 represents a perfect model.

The classification report generated using the original overall scores and predicted overall scores of the test data is given below.

SCORE	PRECISION	RECALL	F1 SCORE	SUPPORT
1	0.46	0.58	0.51	286
2	0.42	0.33	0.37	315
3	0.41	0.34	0.37	306
4	0.38	0.38	0.38	290
5	0.59	0.70	0.64	303
AVG/TOTAL	**0.45**	**0.46**	**0.45**	**1500**

The F1 scores for scores '1' &'5' are good in comparison to scores '2', '3' and '4'. The direct reason for this variation is due to the small test data set with only 1500 reviews. The original distribution of scores in our data is very much skewed towards the score '5' which makes it very important for our model to learn classifying score '5' more accurately. The recall value of 0.70 gives us confidence that the model performance is acceptable for this particular score. Support is the number of occurrences in each score category (labels).

Linear Regression: In an attempt to capture the variation in model error (RMSE) through few iterations and look at the coefficients that are learnt at each step, the TF-IDF matrix is used to fit a linear regression model. The graph to the right is a result of 200 iterations and the corresponding RMSE.



CONCLUSION

With a bigger data set, the F1 scores for all the 5 scores can be improved using a simple logistic regression model. The next steps to validate the hypothesis will be taking two groups of customers of Amazon and do testing similar to AB testing. Recording the change in purchase behavior of group that is exposed to this standardized overall score and the group that does not have this feature will help in quantifying the impact of this analysis. In product categories like Apparel, Shoes & Jewelry where there are many sellers on Amazon platform are not likely to have all the required product information to make a customer decide on purchase. Automating the overall score to depict a standardized numeric value is one of the many ways to address the above mentioned business problem.

REFERENCES

- R. He, J. McAuley. Modeling the visual evolution of fashion trends with one-class collaborative filtering. WWW, 2016
- J. McAuley, C. Targett, J. Shi, A. van den Hengel. Image-based recommendations on styles and substitutes. SIGIR, 2015
- Sentiment analysis: Measuring opinions
- Understanding Sentiment Analysis & Sentiment Accuracy