Sentiment Analysis of Amazon Reviews

Project Report COMP 755 - Machine Learning

presented by
Nils Richter (PID 730291493)
Nick Wang (PID 730190031)

submitted to
Prof. Jorge Silva
University of North Carolina at Chapel Hill

November 2018

Chapter 1

Introduction

1 Problem Definition and Goals

Sentiment analysis forms an important part of web content mining. Online product reviews can be exploited as one of the main raw data sources to examine the aforementioned task. In this regard, our project applies different ML models on Amazon product reviews to determine if their sentiment is positive or negative.

Analyzing the polarity of online reviews bears a tremendous business value. Not only do reviews reveal the satisfaction of a customer but they also give hints towards possible product improvements, marketing needs or innovations. Certainly one of the biggest online resources is Amazon where numerous reviews are posted every day. After having purchased a product, a customer can post his/her opinion with a text and associated number of stars for the product.

In todays age, the availability of data is often not the main problem. The question for firms using sentiment analysis to boost their business is rather which data out of all the available data to use and if we can make use of more targeted data. For our project in particular, we aim to explore whether it is better to use mass data across multiple product-categories or if we can take advantage of the fact that data can be separated by category. We examine if performance of sentiment analysis models improves when they are trained and applied on the same product category or if it is better to use as much data as possible for training, disregarding the category.

The sentiment of a review is categorized as positive or negative depending on the star rating. We identify four product categories for our analysis since emotions for different domains probably vary in their intensity. These are namely ”camera”, ”grocery”, ”videogames”, and ”watches”.

Our approach for this project follows the process for knowledge discovery of

Feyyad [2]. Therefore, the remainder of this report is organized as follows: First, the structure and size of the data set is examined by applying exploratory data analysis. Second, the data is preprocessed by sampling and transforming the texts into machine-readable representations. Moreover, baselines are established. Third, binary classification algorithms are trained on the preprocessed data and evaluated. In the last part, the results of the ML algorithms are assessed with regards to their respective performance metrics on held-out datasets.

2 Datasets

As a data source, we use pre-crawled amazon review datasets[1] that contain at least the product-category, the review text and the rating. As neutral ratings are problematic in terms of sentiment classification and are not as informative as distinct positive/negative reviews, we exclude 3-star-ratings.

Exploratory data analysis gives us an overview and helps us to identify pitfalls and preprocessing needs. Overall, 5,501,521 reviews are originally collected. The distribution for the four product categories and their average star ratings are displayed in Table 1.1. The rating spans from one to five with one being the worst and five the best. Noteworthy, most reviews are rather positive with an overall mean of 4.11 stars. However, there are slight variations in the average for the categories, especially seen when looking at videogames being rated much lower on average than the other categories. This aligns with our assumption that ratings are probably domain dependent.

Category	**Observations**	**Avg. Stars**
Camera	1,800,845	4.12
Grocery	960,204	4.13
Videogames	1,780,268	4.05
Watches	960,204	4.13
Total	**5,501,521**	**4.11**

Table 1.1: Summary statistics for product categories

The high ratings also imply a skewness towards the positive label. Plotting the frequency (Figure 1.1) of the given stars, we can confirm this assumption. This imbalance has to be taken into account as many algorithms cannot easily handle skewness or perform better with balanced data. It is important to note that all

[1] https://s3.amazonaws.com/amazon-reviews-pds/readme.html

balancing attempts can only be applied on the training data. Balancing the test data would lead to an unfair representation of the real-word data.

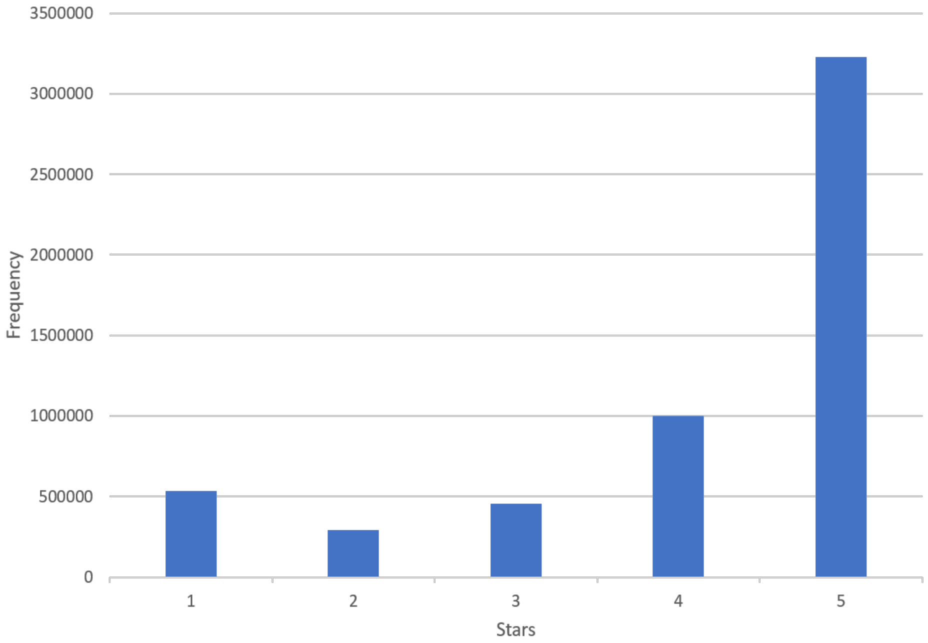

Figure 1.1: Star frequencies across all datasets

Chapter 2

Problem Solving

1 Initial Preprocessing

First of all, we removed the neutral examples, meaning the ones with 3 stars for the already mentioned reasons. Then, the validation/test sets were taken out (around 20,000 examples as we we decided to use 100,000 for each of the 4 categories for training. This ensures reasonable computation times).

In a next step, 50,000 positive and 50,000 negative examples were taken out from each category. This means we decided to balance while sampling. If we randomly sampled 100,000 examples from each category and then balanced later, we would have to upsample the negative class. But since we have such a large amount of data, we were able to get 50,000 individual examples from each class and thus ensure higher data quality.

The individual examples have to then be preprocessed to match the word vectorization standard. This involves normalizing emoticons, converting to lowercase all words, removing stopwords, external links, HTML tags, separating negations, and eventually stemming the text.

With the preprocessed text data, the next step is feature extraction. To achieve this, all the text data were first vectorized using the same count vectorizer, and then applied with Tf-idf transformation to convert to word frequency vectors. And up to this point, the features are ready for model training.

Finally, the label was created based on the star ratings (4 and 5 stars are positive, 1 and 2 stars are negative), and for training purposes, negative and positive labels are represented with 0 or 1.

2 Baseline

It is important to establish a baseline to which the ML algorithms can be compared with. For classification in sentiment analysis, a sentiment lexicon can serve as a baseline. A sentiment lexicon is a collection of words that are often used to express positive or negative sentiments, e.g. ”good, nice, great” and ”bad, horrible, aweful” [4, p. 5]. In this regard, we use two different lexica.

The first sentiment lexicon we use is a popular list of English positive and negative opinion words (around 6800 words) which was collected by Bing Liu over many years starting in 2004. It also includes misspellings, morphological variants and slang to some extent. We use this list by counting the positive (+1) and negative words (-1) occurring in a review and calculate the sum for each review to determine the overall sentiment:

- Positive if sum > 0
- Neutral if sum = 0
- Negative if sum < 0

In our case of binary predictions, for both lexica, the neutral reviews are excluded and the neutral predictions are changed to the major class (positive) before evaluation.

The models throughout this report are assessed w.r.t. to the overall Accuracy (ACC), the Macro-Precision (P), Macro-Recall (R) and Macro-F1-measure (F1). In our setting, Micro-Precision, Micro-Recall and Micro-F1-measure correspond to ACC [1] and are thus not reported explicitly. Macro scores weigh each class equally by taking the unweighted average. We focus on the macro measures since businesses are particularly interested in the underrepresented class of unsatisfied customers.

The first lexicon produces the following performance measures:

Lexicon	**ACC**	**P**	**R**	**F1**
Bing Liu	0.71	0.71	0.71	0.71

Table 2.1: Results of the baseline using Bing Liu’s sentiment lexicon in binary classification

The second sentiment lexicon we considered is NLTK’s ”VADER”. It represents a list of lexical features and their associated sentiment intensity measures. To obtain these intensity measures, ”Amazon Mechanical Turk”, a micro-labor platform, was used. In addition, the lexical features are combined with five grammatical and syntactical rules that emphasize sentiment intensity:

1. Exclamation marks increase intensity, e.g. ”The food is good!!!” is more intense than ”The food is good.”
2. Using ALL-CAPS emphasizes a sentiment-relevant word, e.g. ”The food is GREAT!” is more intense than saying ”The food is great!”.
3. ”Degree modifiers” or ”booster words” can increase or decrease the intensity, e.g. ”The food is extremely good” is more intense than ”The food is good.”, whereas ”The food is marginally good.” is less intense.
4. The word ”but” signals a shift in sentiment polarity, meaning that the sentiment of the text following the ”but” is dominant. E.g. ”The food is great, but the service is horrible” is mixed in sentiment while the second part dictates the overall rating.
5. Cases where a negation flips the polarity of a text are identified by looking at the tri-gram preceding a lexical feature [3].

Hence, Vader is not only a list of opinion words, but also has a rule-based component. Surprisingly, the performance is comparable with the much less sophisticated lexicon of Bing Liu. Both tend to predict the positive class too often.

Lexicon	**ACC**	**P**	**R**	**F1**
VADER	0.71	0.77	0.71	0.69

Table 2.2: Results of the baseline using VADER sentiment lexicon in binary classification

3 Classification

For the sentiment analysis classification, two models were used in this project. Naive Bayes and Logistic Regression.

The classification process is the main portion of this project. The training and testing set consists of features and labels for four different categories of products. The goal of this project is to discover if a larger non-specific dataset or a smaller specific dataset results in a better sentiment classification accuracy.

To find out about this, five Multinomial Naive Bayes models and five Logistic Regerssion models are trained with full dataset, camera set, grocery set, watches set and videogames set separately. Afterwards, the two models trained with full dataset are used to predict values in all five testing sets to see the result. The other eight models are used to test their own category testing sets separately and plot the result.

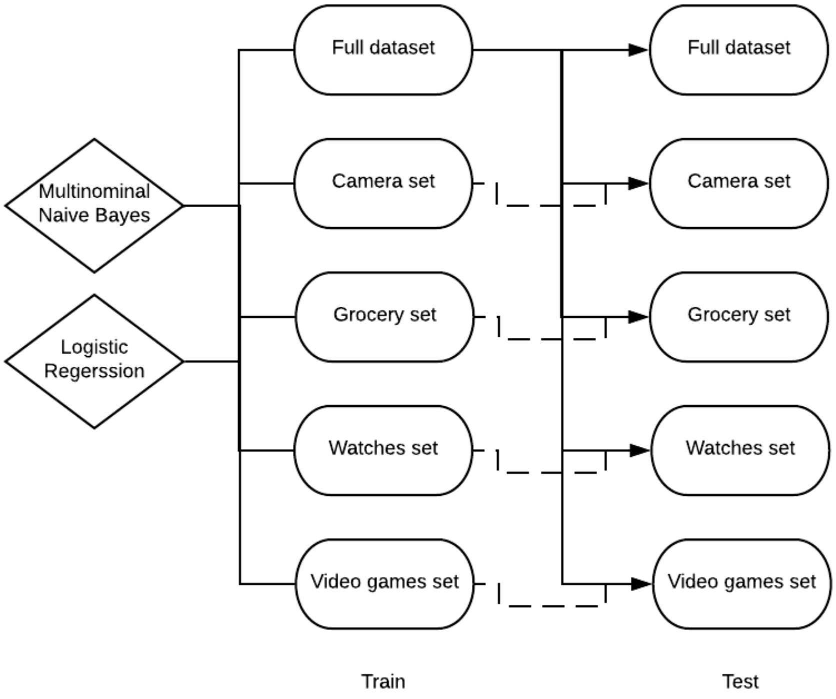

Figure 2.1: Training and testing process

Chapter 3

Evaluation of Results

Table 3.1: Classification F1 score result

	Fullset NB	Subset NB	Fullset LR	Subset LR
Fullset Test	0.769886	/	0.832783	/
Camera Test	0.752214	**0.760436**	0.841892	**0.844153**
Grocery Test	**0.755148**	0.732736	0.790949	**0.794246**
Watches Test	**0.814971**	0.796264	**0.858764**	0.856779
Videogames Test	**0.756104**	0.728851	0.837709	**0.838754**

Testing on the same classes with model trained with the much larger but less specific dataset (referred as fullset) or a smaller but more specific dataset (referred as subset) using Naive Bayes and Logistic Regression model had different results. As presented in the table, three out of four classes had better result in the Naive Bayes model trained with fullset. The difference is around 2 percent, which is rather significant in a dataset consisting of merely 400k review text data.

On the other hand, in the Logistic Regression model, three out of four classes had slightly better result in the subset-trained model. However, the difference is relatively small, only around 0.2 percent. The results are very similar no matter the training set. Thus the difference in training data, in this case, did not pose a substantial influence on the result.

Additionally, all the Logistic Regression F1 scores are substantially higher than Naive Bayes, no matter trained with fullset or subset. As suggested by the result, training with a smaller dataset using Logistic Regression can yield better prediction accuracy than the Naive Bayes classifier trained with larger dataset.

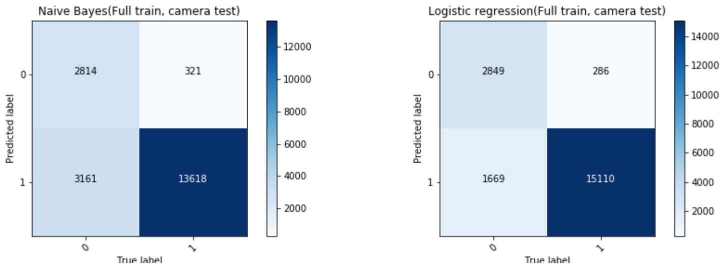

Figure 3.1: Confusion Matrix Fullset training on camera testing

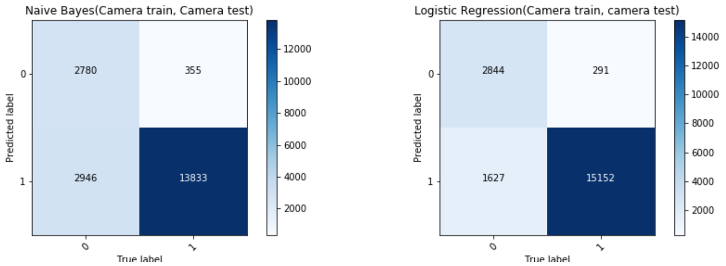

Figure 3.2: Confusion Matrix Camera training on camera testing

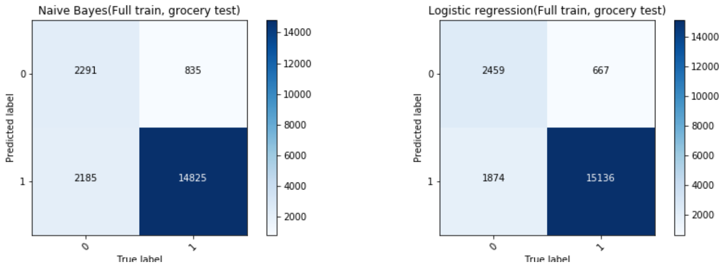

Figure 3.3: Confusion Matrix Fullset training on grocery testing

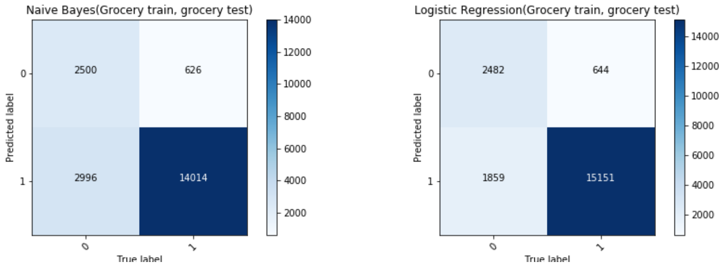

Figure 3.4: Confusion Matrix Grocery training on grocery testing

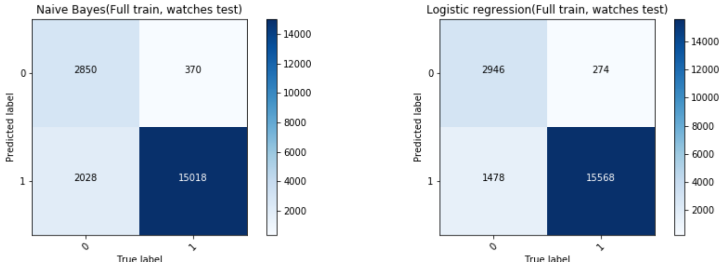

Figure 3.5: Confusion Matrix Fullset training on watches testing

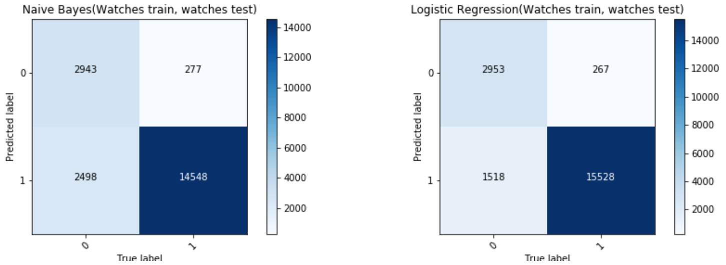

Figure 3.6: Confusion Matrix Watches training on watches testing

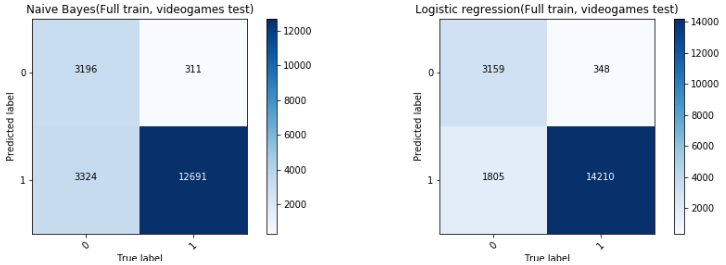

Figure 3.7: Confusion Matrix Fullset training on videogames testing

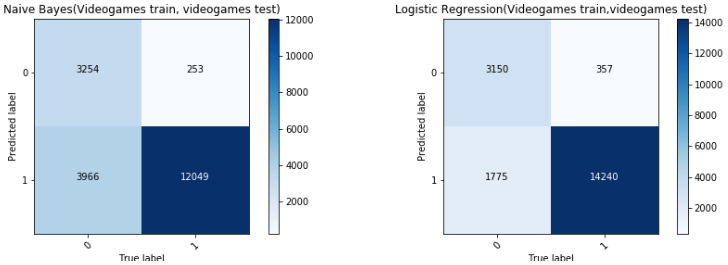

Figure 3.8: Confusion Matrix Videogames training on videogames testing

Finally, our verdict on the research question - whether a larger non specific dataset or a smaller specific dataset could yield better binary sentimental analysis classification results in Naive Bayes and Logistic Regression model, from our dataset, the result showed that using a training set that is not as specific but contains more data resulted in better F1 score and accuracy in the classification with Naive Bayes model. With Logistic Regression model, the difference in training sets did not have a significant impact on the classification performance. And overall, the Logistic Regression model outperformed Naive Bayes on the exact training and testing set in all of our results. The result supports the hypothesis that larger datasets might yield better performance in binary sentimental analysis classification with Naive Bayes model, but not with Logistic Regression. As observed in our result, the Logistic Regression model, no matter small or large training set, performed significantly better than Naive Bayes model. In this case, we suggest researchers/companies use Logistic Regression over Naive Bayes for sentiment analysis. Due to smaller amounts of data, this results in less computation time and data storage for people performing sentiment analysis with product reviews.

Bibliography

[1] 3.3. Model evaluation: quantifying the quality of predictions scikit-learn 0.19.1 documentation, 2017.

[2] U. M. Feyyad. Data mining and knowledge discovery: making sense out of data. *IEEE Expert*, 11(5):20–25, October 1996.

[3] C.J. Hutto and Eric Gilbert. VADER: A Parsimonious Rule-based Model for Sentiment Analysis of Social Media Text. In *Proceedings of the 8th International Conference on Weblogs and Social Media, ICWSM 2014*, January 2015.

[4] Bing Liu. *Sentiment analysis and opinion mining*. Synthesis lectures on human language technologies; 16. Morgan & Claypool, San Rafael, Calif.], 2012.