Sentiment Analysis of Amazon Reviews

1st Roman Bellisari
Stevens Institute of Technology ECE
Computer Engineering
Hoboken, NJ
rbellisa@stevens.edu

2nd Matthew Lepis
Stevens Institute of Technology ECE
Computer Engineering
Brick, NJ
mlepis@stevens.edu

3rd Michael Delcid
Stevens Institute of Technology ECE
Electrical Engineering
West New York, NJ
mdelcid@stevens.edu

I. Introduction

The nature of human interaction is extremely complex with a wide range of variation from person to person. Written communication through the English language is such an important aspect of day to day life. With the rapid growth of e-commerce and the ever increasing reliance on the internet, the sheer volume of text available online is incomprehensible. One of the many unique value offerings of e-commerce giant Amazon is their utilization of user generated content in the form of product reviews. With a star rating system ranging on a scale from one (worse) to five (best), these reviews are essential for customers to better understand the product and to provide unique user feedback.

The mission of the teams analysis is to model the sentiment of millions of Amazon product reviews on various products to predict and summarize the reviews sentiment via the output of a five class probabilistic classifier. The team plans to implement both a multinomial logistic regression and also an artificial recurrent neural network to provide probabilistic sentiment classifications on the interval [1, 5]. After the analysis, the team will identify the trends within each algorithm and determine the optimal model.

Our dataset is derived from an Amazon review dataset originally from the Stanford Network Analysis Project (SNAP) (http://snap.stanford.edu/data/web-Amazon.html/). It contains 34,686,770 amazon.com reviews from the 18 year period, between June 1995 and March 2013. Reviews included product, user information, ratings, and a plaintext review. The derivation done for the dataset we are utilizing reduced the amount of data operated on and proportioned the data in a way that would represent the five different classifications of our projects goal in equal measures.

II. Related Work

[1] In the research paper Predicting the Semantic Orientation of Adjectives by Vasileios Hatzivassiloglou and Kathleen R. McKeown, the researchers identified and validated the positive or negative semantic orientation of conjoined adjectives within a large collection of written texts known as a corpus.

The semantic orientation or polarity of a word indicates the direction the word deviates from its meaning for its semantic group or lexical field. Within lexical fields, words are grouped together by topic and conveys a similar meaning to an idea. For instance, the lexical field for the word hiker may consist of the words backpacker, explorer, climber, alpinist, or adventurer.

Semantical groups are words that are grouped together by meaning. For example within the semantic field color the words red, blue, yellow, black all share a similar meaning in that they are used to describe color. Semantically similar words can be identified readability on the basis of linguistic cues.

Within their research, Hatzivassiloglou and McKeown presented a method that automatically retrieves semantic orientation information using indirect information collected from a large collection of text. The method extracts domain-dependent information from the corpus and adapts to a new domain when the corpus is changed. The method achieves high precision in classifying adjectives and the approach is as follows: Conjunctions of adjectives are extracted from corpus along with relevant contextual information A log-linear regression model combines information from different conjunctions to determine if two conjoined adjectives are of the same or different orientation. A clustering algorithm separates the adjectives into two subsets of different orientation and places as many words of the same orientation into the same subset. The average frequencies in each group are compared and the group with the higher frequency is labeled as positive.

[2] In analyzing opinion based product reviews on e-commerce websites, researchers working in the Department of Computer Science at the University of Illinois at Chicago mined opinion features on products reviewers have commented on. Their work re;ated to two areas of research, text summarization and terminology identification. A majority of existing work on text summarization focuses on a single document covering similar information with the purpose of summarizing similarities and differences in the information contents of the documents. Within the realm of terminology identification, statistical approaches are used to exploit the relationship by which terms are found within other words within a phrase. For example, in the English language the words brick and mortar might be found in close proximity or are correlated with one another in various documents across the internet. In the work by Minqing Hu and Bing Lui, they studied feature based opinion summarization of customer reviews in two steps:

By identifying the features of the product that customers have expressed opinions on (opinion features) and ranking the features according to their frequencies that they appear in the

reviews. For each feature, the researchers identified how many customer reviews had positive or negative opinions. Specific reviews that express these opinions are attached to the feature.

Additionally, in order to discover features not explicitly mentioned in the reviews, it is necessary to implement a semantic understanding of sentences into the model. For example in the sentences The pictures are very clear and Overall a fantastic very compact camera the product features are not easily discernible and will require more sophisticated techniques to pick out implicit features.

[3] In the paper Character-level Convolutional Networks for Text Classification researchers Ziang Zhang, Junbo Zhao, and Yann LeCun at New York University explored the use of character-level convolutional networks for text classification.

The researchers used a 1-D version of the temporal-max pooling used in computer vision to train convolutional networks deeper than 6 layers. A rectifier was introduced to add a non-linearity component in the model which made the convolutional layers similar to rectified units(ReLUs). The characters in the text had to be quantized using one-hot encoding before it was used as input for the model. Weights within the convolutional network were initialized by using a Gaussian distribution.

In order to control the generalization error for deep learning models, data augmentation using an English Thesaurus was introduced. Every synonym to a word or phrase was ranked by the semantic closeness to the most frequently seen meaning. The number of words that were replaced was determined by using a geometric distribution. After the model was trained, other traditional and deep learning methods were used in order to quantitatively evaluate the convolutional neural network model. Our approach is similar to Hatzivassiloglou and McKeown[1] in that we are retrieving semantic orientation of adjectives, but are not limiting ourselves to one word class. Our purpose in determining the semantic orientation of word classes is to accuratley tokenize words that will be used in the sentence analysis of product reviews.

Within the paper Mining Opinion Features in Customer Reviews Minqing Hu and Bing Liu focused on mining implicit or explicit product features left in the comments by reviewers. They then tabulated the frequency of positive and negative reviews on each feature in order to provide future customers with a concise way of evaluating a product. Our team will be mining for product features in reviews for the purpose of classifying it within a five star ranking system and then using these rankings to predict the actual rating of the product. This way lay the groundwork for distinguishing between real and fake reviews left by customers on a given product.

III. Our Solution

A. Description of Dataset

The derived dataset is constructed by randomly taking 600,000 training samples and 130,000 testing samples for each review score from 1 to 5. In total there are 3,000,000 training samples and 650,000 testing samples. For continuity, the data is split 82.2

Our dataset is formatted in .CSV format and as reported in the introduction, only 3 of the full datasets features were most relevant to our goal: class index (star rating) [1 to 5], review title, and review text, the rest of the features were trimmed.

Given that most machine learning algorithms require numerical vector input, the training and testing data must be converted from raw text data to vector form. The team filtered out symbols, stop words, punctuation and converted all characters to lowercase through a process called sentence tokenization. Next the team removed duplicate or non English reviews. Next, the Bag of Words tokenization process is utilized to maintain the sentiment of the text corpus. After tokenization pre-processing, the tokenized data is normalized through a lemmatization rather than stemming. This allows for the full word to be normalized rather than simplified to the root or stem word. Next, the tokenized and normalized data is transformed into a vector through the utilization of the NLTK library. This vectorized output is stored with each variable representing a value on [-1, 1].

For more information pertaining to the original dataset, please refer to the following paper: J. McAuley and J. Leskovec. Hidden factors and hidden topics: understanding rating dimensions with review text. RecSys, 2013..

B. Machine Learning Algorithms

Upon the vectorization of the raw text data, an optimal machine learning algorithm must be identified and selected. The team will analyze the effectiveness of both traditional machine learning alongside deep learning algorithms. Utilizing supervised learning through the implementation of a multinomial logistic regression model and a recurrent neural network (RNN), the team will determine the most optimal model. Since the vectorized data exists entirely on [-1, 1], there is no need for further data pre-processing or normalization for the RNN deep learning implementation.

One of the key preferences that impacted the algorithm selection process was the desire to output a probability distribution for each of the five classes. The teams need to output probability distributions for each class rather than just classification predictions comes from the nature of the dataset and the classification features. Due to the fact that the output sentiment ratings classes are not entirely independent of each other, it is favorable to include classification probabilities in contrast to an individual classification output. For example, a prediction output of a review with strong positive sentiment with higher percentages of ratings belonging to class 4 and class 5 is more useful than a single classification of class 4. Because some traditional machine learning classification algorithms such as k nearest neighbor and support vector machines yield deterministic classifications exclusively rather than probability distributions, these algorithms were omitted from the selection process.

Although standard logistic regressions only allow for binary classification, this model can support multiclass classification by extending the logistic regression into multiple binary classification problems. To receive multiclass probability distribu-

tions, the logistic regression cost function is changed to a cross entropy loss function. The multinomial logistic regression model was chosen as a traditional machine learning model because of its strengths with binary classification and ability to easily be converted into a multiclass probabilistic classifier. This algorithm was chosen for the sentiment analysis problem because of its simplicity, its capability to fight overfitting and its ability to produce multiclass probability distributions.

The default parameters of Scikit Learns logistic regression will be used accordingly with the optimization solver beginning with the default Limited-memory Broyden - Fletcher - Goldfarb - Shanno solver. The maximum number of iterations required for convergence will be reduced from the default 100 to 30 iterations due to the large training dataset. Depending on whether or not the model is overfit, the team may introduce hyper parameterization in the form of regularization penalties of either ridge, lasso or elastic net items.

Because the Amazon reviews dataset utilizes training data with millions of samples, the team wanted to also reap the performance benefits of deep learning. Additionally, with deep learning implementations, there is less emphasis on performance attributed to feature selection and more emphasis on direct model implementation. Recurrent neural networks (RNNs) are a type of deep learning network and are very useful within the field of linguistic processing because they account for the order and sequence of the input data. RNNs are often an optimal solution to NLP problems because of the natural approach to capturing the importance of sequential data found within raw text. This is directly applicable to the teams dataset because of the large amounts of raw text data required for analysis. One potential alternative to a RNN model was the convolutional neural network (CNN) model which the team decided against because of RNNs strengths when processing text data. The team will compare the results from the multinomial logistic regression and the RNN to identify the best algorithm.

C. Implementation Details

When evaluating the performance of the multinomial logistic regression model, several metrics will help understand model performance. As mentioned above, the cross entropy loss function will be utilized as the cost function for optimization. For performance metrics, the team will calculate precision, recall, and F-1 score for each of the classes. Precision will enable the team to determine the ratio of correct class predictions to total class predictions. This will help to identify false positive predictions. Recall enables the team to determine the ratio of correctly predicted positive classes to every observation in that class to detect false negative predictions. The F-1 score is used to summarize both precision and recall. Because precision and recall are natively implemented in binary classification problems, there will be some slight modification to implement a multiclass confusion matrix.

IV. References

[1] Vasileios Hatzivassiloglou and Kathleen R. McKeown, ”Predicting the Semantic Orientation of Adjectives” [2] Minqing Hu and Bing Liu, ”Mining Opinion Features in Customer Reviews” [3] Xiang Zhang and Junbo Zhao and Yann LeCun, ”Character-level Convolutional Networks for Text Classification”