An unsupervised algorithm for online review comprehension: An assessment based on online product reviews

Gowtham Yedapadi Rajaram, Jagpreet Sethi, Ramya Vangari, Surya Vikas Arun Alampally

Abstract

As ecommerce has grown, so has the importance of online reviews. However, reading these reviews can be time-consuming. Existing APIs such as twinword, sentiwordnet etc. have limitations in flexibility and are also unable to segregate ratings for specific product feature. We present an unsupervised learning method to automatically summarize the information in online product reviews and benchmark its performance to supervised learning models. The method can help consumers searching for new products identify what features are important in a new product, and also use these reviews to assign weights to these features.

1. Introduction

The wide spread adoption of the Internet has led to tremendous growth of e-commerce. According to data released by the U.S. Commerce Department, more than half of total retail sales growth in 2015($ 60 billion from $4.63 trillion in 2014 to $4.69 trillion in 2015) is accounted to online sales ($43.4 billion from $298.3 billion in 2014 to $341.7 billion in 2015) (Zaroban 2016). Drivers of this growth include 24X7 access from any location, large variety and product diversity, and price competitiveness.

However, the anonymous and remote nature of online purchases also makes it risky because customers cannot see or experience a product before purchase and may not know the seller or its selling policies (e.g., warranties, returns). It is difficult for customers to choose a product from the available variety of options just by looking at the images and description provided by the seller. In traditional storefronts, trained salespeople were able to educate customers on the importance of difference product features and explain why a product in a category was more expensive in another. This is especially important in new product categories, where customers cannot use prior experience to judge a product's fit with their needs.

Customer reviews serve this role in online shopping. Sellers as well as retailers (e.g. Amazon), encourage buyers to write reviews about purchased products to benefit other buyers. Customer reviews have therefore become very important in online commerce are play an increasingly important role in customers' purchase decisions.

Unfortunately though, the dramatic growth of e-commerce has also resulted in a growing volume of online reviews. For example, popular products on Amazon.com may have hundreds of reviews, of which very few are actually useful in helping customers' purchase decisions. A shopper in the market for a specific product may need to evaluate 15 – 20 different products. A typical Amazon review has about 250 words. If a set of candidate products has a total of 700 reviews, it can take a customer 12.5 hours to go through all the reviews to get a reasonable idea about a product and narrow down his purchase decision to a couple of candidate products. With time at a premium today, providing a summary of all the reviews in a product category, especially organized along key product features of interest to the customer, can save customers a considerable amount of time in manually reading all the reviews, and help them make more efficient and informed product purchase decisions.

Currently, reviews are generally listed on online websites in reverse chronological order of their entry, without attention to their content or usefulness. Some websites such as Amazon attempt to organize reviews as "most detailed reviews" or "most critical reviews." Typical websites also allow customers to enter an overall rating for a product (e.g., on a five-star scale) and provide an average rating across all reviews. Some websites (e.g., Expedia.com) also allow customers to rate specific attributes of a product or service (e.g., location, comfort, and amenities of hotels). However, to the best of our knowledge, reviews are not yet organized by product features (e.g., image quality, video quality, and processing speed of a smartphone), or used to rate products by feature.

In this paper, we present an unsupervised learning method for extracting specific sentences about product features from all online reviews, using sentiment analysis to determine whether each sentence is positive or negative, and generating a product review summary based on the number of sentences that are positive or negative. For evaluation purposes, we benchmark our algorithm to a Naïve Bayes supervised learning algorithm. These methods combine existing methods from text mining, natural language processing, Naïve Bayes learning, and semantic analysis to carry out these tasks.

In our supervised learning model, users can specify the specific product features they are interested in and use the algorithm to extract all positive and negative reviews for those features. This improves the accuracy of the algorithm, but not only does it place the burden of product knowledge upon the user, it also takes up

consumer time in specifying these features. In the unsupervised learning model on the other hand, users do not specify any product features. Instead, they can depend upon the algorithm to intelligently identify the most pertinent product features, along with the positive, and negative reviews by feature.

Our work is expected to have several contributions for research and practice. For research, the method presented in this paper can be the basis for continuous improvements in unsupervised learning methods for summarizing online product reviews. For practice, our API can be used to identify product features most desired by customers of a product or product category, summarize product reviews by features, and subsequently, using customer information, target specific product features for strategic positioning and improvement.

2. RELATED WORK:

2.1. Supervised Approach:

Fast and accurate sentiment classification using an enhanced Naive Bayes model (Narayanan et al. 2013):

The traditional supervised Naïve Bayes learning approaches in this domain are called Bernoulli Naïve Bayes algorithms. They use a combination of methods like effective negation handling, word n-grams and feature selection by mutual information to achieve a significant improvement in accuracy. Duplicate words are removed from the document since they don't add any additional information. Including just the presence of a word instead of its count has been found to improve performance marginally, especially when the training sample is large.

Laplacian Smoothing is used as a mechanism to deal with words not seen in the training set. If the classifier encounters a word that has not been seen in the training set, the probability of both the classes would become zero as there would not be a reference probability to compare with. This problem can be solved by Laplacian smoothing using the expression below to create a low probability for assigning the item to class j. Usually, k is chosen as 1.

$$P(x_i|c_j) = \frac{Count(x_i) + k}{(k+1) * (No\ of\ words\ in\ class\ c_j)}$$

Negations are handled by the algorithms using a state variable to store the negation state. It transforms a word followed by a not or n't into "not_" + word. Whenever the negation state variable is set, the words read are treated as "not_" + word. The state variable is reset when a punctuation mark is encountered or when there is double negation.

n – grams are words such as "very" or "definitely," which don't provide much sentiment information on their own, but in specific phrases like "very bad" or "definitely recommended" increase the probability of a document being negatively or positively biased. This information about adjectives and adverbs is captured by including bigrams and trigrams.

However, the use of higher dimensional features like bigrams and trigrams presents a new problem of increasing the number of features in the corpus. Most features are redundant and noisy in nature. To deal with this situation, a basic filtering step is used to remove features/terms which only occur once. The features are then further filtered on the basis of mutual information.

A Naïve Bayes Strategy for Sentiment Analysis on English Tweets (Gamallo and Garcia 2014):

Two different Naive Bayes classifiers, based on two different strategies were presented:

Baseline: This is a Naive Bayes classifier that learns how to classify the three sentiment categories found in the corpus (positive, negative, and neutral), from the original training corpus.

Binary: This was trained on a simplified training corpus and makes use of a polarity lexicon. The corpus was simplified since only positive and negative tweets were considered.

In this implementation of the algorithm, the main preprocessing tasks considered are:

• removing urls, references to usernames, and hashtags.

• Reduction of replicated characters (e.g. looooveeee→love).

• identifying emoticons and interjections and replacing them with polarity or sentiment expressions (e.g. :-)→good).

These cleaning measures improved the quality of the data used for the analysis.

They also identified the common features in the text, including:

- Lemmas (UL): To characterize the main features underlying the classifier, they made use of unigrams of lemmas instead of tokens to minimize the problems derived from the sparse distribution of words. Moreover, only lemmas belonging to lexical categories were selected as features, namely nouns, verbs, adjectives, and adverbs. So, grammatical words, such as determiners, conjunctions, and prepositions were removed from the model.
- Multi-words (MW): In addition to unigrams of lemmas, they also considered multi-words extracted by an algorithm based on patterns of part-of-speech (PoS) tags. In particular, they used the following set of patterns: NOUN-ADJ, NOUN-

NOUN, ADJ-NOUN, NOUN-PRP-NOUN, VERB-NOUN, VERB-PRP-NOUN. The instances of bigrams and trigrams extracted with these patterns were added to the unigrams to build the language model.

- Polarity Lexicon (LEX): A polarity lexicon was built with both Positive and Negative entries from different sources.
- Valence Shifters (VS): Negative words were taken into account that can shift the polarity of specific lemmas in a tweet.

3. Unsupervised method

Opinion mining or sentiment analysis is the task of mining polarity of opinions using natural language processing, data mining and related techniques. Our specific interest was in identifying product features. For example, important features of a camera include battery life, zoom etc. An unsupervised model for analysis of product reviews at feature level is obtained using the following sequence of procedures: Review Collection, Pre-Processing, Feature Extraction, Co-reference Resolution, Subjectivity/Objectivity Classification, Opinion words identification, Orientation Detection, and finally feature based summary.

Algorithm for orientation detection: In orientation detection, we determined the positive or negative score of every feature in the reviews. Reviews contain one or more sentences. We first calculated the sentence level score of the opinion by analyzing each sentence in the reviews. By combining all the sentences in a review, we calculated the overall score of features in reviews. We then calculated the overall score of each feature by combining all reviews. For calculating the opinion score of each feature, an algorithm for determining the score of adverb adjective and adverb verb combinations in sentences is used.

Feature Based Summary: Feature based summary calculates the overall score of each feature in all reviews and produce a feature based summary. Positive and negative scores of aspects are calculated separately. This gives us total positive and negative scores of the different features. By using these results, a sentiment profile of the product (such as a camera) can be created. The scores of opinions on each feature in all reviews can be calculated by aggregating the opinions across features. For each feature j of the product we have,

$$\text{Total\_Pos\_Score}[j] = \sum_i \text{PositiveScore}_{i,j}$$

$$\text{Total\_Neg\_Score}[j] = \sum_i \text{NegativeScore}_{i,j}$$

For Normalization, we have

$$\text{Norm\_Pos\_Score}[j] = \text{Total\_Pos\_Score}[j] \div \sum_i$$

$$\text{Norm\_Neg\_Score}[j] = \text{Total\_Neg\_Score}[j] \div \sum_i$$

3.1 Benchmark: Supervised Approach (Naïve Bayes)

As described earlier, in supervised approaches, the user has to specify a set of product features (PFs) for which a summary has to be generated. First, all the sentences that contain at least one of the PFs are extracted from the reviews. Then we identify the sentiment of each sentence as positive or negative and also assign a score in the range 0-1 for the sentence. This sentiment score measures how positive or negative the sentence is, using a Naïve Bayes classifier.

Naïve Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the product features. Naive Bayes classifiers work well on text classification and take less time to train compared to other models like support vector machines (Narayanan et al. 2013). An advantage of the Naïve Bayes approach is that it can be extended to any product or service by specifying the PFs related to that product or service. Each component in the system is explained below:

Feature (NF) Set[1]. We used the natural language toolkit (NLTK) for classifier implementation. All of the NLTK classifiers work with NF sets, which can be simple dictionaries mapping a NF name to a NF value. For text, a simplified bag of words model where every word is NF name with a value of True can be used. Generally an opinion is conveyed using an adjective or a combination of adjective with other parts of speech. The combination of adjective with other parts of speech can be captured as a NF in Naïve Bayes classification by using Bigrams. A bigram is a sequence of two adjacent elements from a string of tokens, which are typically letters, syllables, or words. As reported in (Narayanan et al. 2013), the algorithm ignores words like "very" or "definitely" as they don't convey any sentiment on their own, but phrases like "very bad" or "definitely recommended" increase the probability of a sentence being negatively or positively biased. Hence, by including bigrams, this information about adjectives and adverbs can be captured. So, our NF set contains single word NFs as well as some bigrams to improve the performance of the classifier and obtain higher accuracy

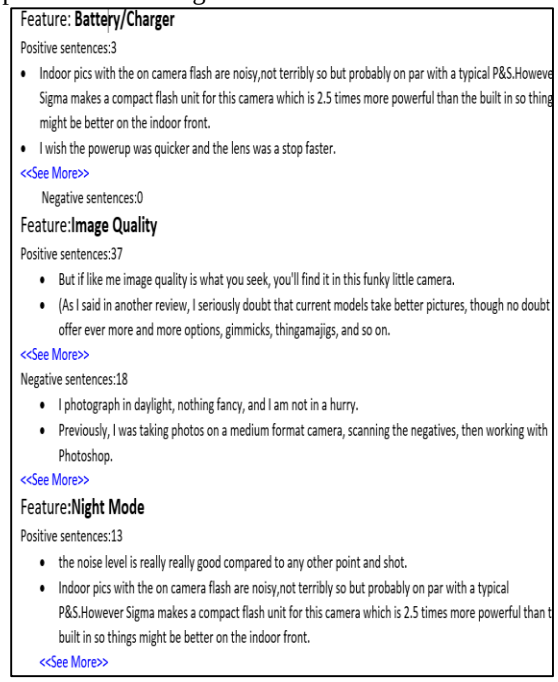
[1] We differentiate the features used in the feature set representation of sentences from the product features by representing the former as NF and the later as PF in the paper

Reviews. The reviews database contains the reviews for which PF based summary is to be generated. These reviews can passed by placing them in a simple text file.
Feature (PF) Sentences. In this approach the user has to provide a set of frequent PFs. For example, in the case of the digital camera, we used a set that includes PFs like Battery life, night mode, noise, etc.

As we are producing a PF based summary, we identify sentiments of only those sentences that are about frequent PFs. Hence, we extract all sentences that contain at least one of the PFs specified in the predefined list. These sentences are used to produce the summary.

Naïve Bayes classifier. We trained the Naïve Bayes classifier using the trained list of reviews in the training dataset. We first represent each review as a NF set and then pass these NF sets to train the Naïve Bayes classifier. Once the Naive Bayes classifier is trained, sentences that are identified in the "Feature (PF) Sentences" step can be passed to the classifier to obtain the polarity of the sentence as positive or negative. We also generate a score using this classifier that indicates the extent to which the sentence is positive or negative.

Summarize Sentiments. The final summary consists of the following details for each of the PF identified. A sample is shown in Figure 1:


Feature: **Battery/Charger**

Positive sentences:3

- Indoor pics with the on camera flash are noisy,not terribly so but probably on par with a typical P&S.Howeve Sigma makes a compact flash unit for this camera which is 2.5 times more powerful than the built in so thing might be better on the indoor front.
- I wish the powerup was quicker and the lens was a stop faster.

<<See More>>

Negative sentences:0

Feature:**Image Quality**

Positive sentences:37

- But if like me image quality is what you seek, you'll find it in this funky little camera.
- (As I said in another review, I seriously doubt that current models take better pictures, though no doubt offer ever more and more options, gimmicks, thingamajigs, and so on.

<<See More>>

Negative sentences:18

- I photograph in daylight, nothing fancy, and I am not in a hurry.
- Previously, I was taking photos on a medium format camera, scanning the negatives, then working with Photoshop.

<<See More>>

Feature:**Night Mode**

Positive sentences:13

- the noise level is really really good compared to any other point and shot.
- Indoor pics with the on camera flash are noisy,not terribly so but probably on par with a typical P&S.However Sigma makes a compact flash unit for this camera which is 2.5 times more powerful than t built in so things might be better on the indoor front.

<<See More>>

Figure 1: Final output generated from each method

The output is configured in the following format:

Feature: <PF Name>

Positive Sentences: <Number of positive sentences about the PF in reviews>

<Up to two positive sentences about the PF in reviews>

<<See More>>

Negative Sentences: <Number of negative sentences about the PF in reviews>

<Up to two negative sentences about the PF in reviews>

<<See More>>

As expected, the <<See More>> option shows more positive/ negative sentences if there are any, about the PF in reviews.

3.2. Unsupervised Approach

In this approach the frequent PFs are identified by the method and the user does not have to pre-specify a set of PFs related to the product. The first step in this approach is to identify the frequent PFs in the reviews. This is done by applying Part of Speech (POS) tagging to each sentence in the review. Then we find out the frequent PFs by identifying the noun phrases that occur most frequently. We do this since most PFs will be noun phrases. Once the frequent PFs are identified, the next step is to extract all the sentences from the reviews which contain at least one of the frequent PFs. Then we identify the sentiment for each extracted sentence as positive or negative. Finally, a summary of the reviews is produced by listing the number of positive and negative sentences about each of the PF and also some of the positive and negative sentences identified for each PF.

Each component of the system for the unsupervised approach is explained below:

Review Database. The review Database contains the input reviews for which a PF based summary has to be generated. This is a simple text file which contains reviews about a product.

POS Tagging. We identify the PFs that are frequently mentioned in the reviews. Most of the PFs are noun phrases. Hence, Part-of-speech is an initial step in identifying the frequent PFs. We used the NLTK package to parse each review and split it into separate sentences. We then performed part-of-speech tagging for each sentence and identified the noun phrases in the sentences.

Feature (PF) identification. Once all noun phrases in the reviews are identified, we identify the noun phrases that are mentioned frequently. These frequently occurring noun phrases are the PFs that are mentioned most frequently in the reviews. The final summary will relate to these PFs.

Frequent Feature (PF) Sentences. Once we identify the frequent PFs, we extract all sentences that contain at least one frequent PF. These are sentences that are used to produce the summary.

Opinion Word Extraction. Opinion words are the words that are used to express subjective opinions about a PF. Generally, adjectives and adverbs are used to express opinions, so we extract these from each sentence. We use POS tagging to identify adjectives and adverbs. Through observation during this research, we observed by examining numerous sentences that an opinion word occurs immediately after the PF or within two words before the PF in the sentence. So we looked for adjectives and adverbs that occurred after the PF or with in two words before the PF in the sentence and considered this as an opinion word.

Orientation Identification. In this step we identify the orientation of the opinion words identified in the "opinion word extraction" step as positive or negative. The orientation of the opinion word is considered as the orientation of the sentence. To find out the orientation of a word, we initially started with a bag of words containing 2000 positive and 2000 negative words. We looked for each opinion word identified whose orientation was to be determined in the bag of words. If the word was not found, then synonyms of all the words in the bag of words was searched to find a match with the opinion word. This process was repeated for four cycles. If a match was found, then the orientation of the root word was identified as the orientation of the opinion word, and the opinion word was added to the bag of words with the identified polarity.

Summary Generation. Once the sentiment is identified for all the sentences that contain opinions for the identified PFs, we produced a summary similar to that produced in supervised approach.

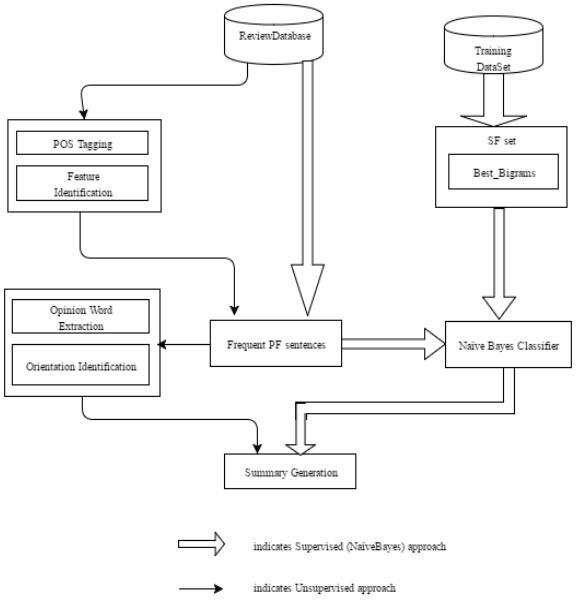


Figure 2: System overview of supervised and unsupervised approaches

We implemented our methods using the Python language. Python was preferred because it is widely used in the community and standard libraries are available to perform most calculations, thereby minimizing implementation errors. Figure 2 shows an overview of both these methods.

4. Data

We tested the two algorithms using multiple samples of digital camera reviews from Amazon. The Naive Bayes classifier requires a dataset that has text which is classified as either positive or negative in order to train the classifier. The training dataset that we used is publicly available dataset of movie reviews from the Internet Movie Database (IMDb). This dataset is used by Narayanan, Arora and Bhatia (Narayanan et al. 2013) and contains 12,500 positive and 12,500 negative movie reviews. Movie reviews cover a wide range of human emotions and cover a most the adjectives related to sentiment classification and hence is the best choice to train a sentiment classifier.

For testing, we intended to choose reviews which were comprehensive and our primary goal was to obtain the best source of reviews available. We found that a dataset of all reviews spanning from May 1996 to July 2014 was already available (McAuley and Leskovec 2013). Since this was an authoritative database of reviews, this was used for this paper. Amazon is a good source of reviews for many reasons. It is one of the most popular e commerce websites, offering the widest array of products. Also, it has some of the best reviews compared to peers such as Walmart, Target, etc.

The research team determined that it was best to start with a specific product to estimate the utility of the method. After looking through various products, we decided to work with cameras. Cameras are popular products, and the features are easily identified.

We had reviews of 450 cameras in our data set, with an equal number of positive and negative reviews, spanning from May 1996 to July 2014.The dataset is a 1.9 GB file and the data set included reviews (ratings, text, helpfulness votes), product metadata (descriptions, category information, price, brand, and image features), and links (also viewed/also bought graphs).

The data was initially in JSON format. We extracted the data in the format required for our programs, which used python data frames.

5. Data Analysis method

In the unsupervised method, we started by identifying the nouns or noun phrases which occurred most frequently (termed frequent feature sentences), using the Parts of Speech Tagger. Then we extracted the

opinion words and identified the orientations of these words from the feature. In the supervised method, we manually identified the set of features that were most frequently mentioned in the reviews of digital cameras. As can be seen, the supervised approach works only for products where such manual identification is possible whereas the unsupervised approach can be applied for reviews of any product.

6. Results

The unsupervised model achieved an accuracy of 61.5%, i.e., the model predicted 61.5% of the reviews correctly. To evaluate this algorithm, we used manual identification of the sentiment. Manual results were compared to those obtained from the model. The failure percentage of 38.5% can be attributed to reasons such as "unable to identify sentiment", "users talking about a different product", "no sentiment (neutral)" etc.

The supervised learning method had an accuracy of 72.1%, comparable to Textblob, the best commercial algorithm available for this purpose, and better than alternative algorithms such as SentiWordNet and Twinword API. The evaluation method was the same as for the unsupervised approach, i.e., the reviews were coded manually as positive or negative and the results compared to the results obtained from the algorithm.

7. Discussion

While the unsupervised algorithm currently has an accuracy of 61.5%, which is lower than the accuracy of the supervised approach, this algorithm has many advantages. The primary advantage is the lack of any training needs. The algorithm is also scalable since it identifies the product features internally. In particular, though our work started with evaluating online reviews of physical products such as digital cameras, we expect these algorithms to work equally well for digital products such as online movie downloads and services such as cellular service providers and online travel sites.

However, one disadvantage with an unsupervised model is the lack of an opinion word extraction technique.

While the supervised approach was more accurate, a limitation of the approach is the need to train the model. A related disadvantage is that the supervised approach is not scalable.

In future approaches, we will look at methods to automatically update the POS dictionary. The POS function in the unsupervised algorithm uses an English dictionary. If the noun phrase is found in this dictionary then it is treated as correct, otherwise the noun phrase can be misinterpreted, as an adjective for instance. Hence, approaches to integrate new words into the POS dictionary would be useful.

Another avenue for improvement is the treatment of bigrams and n-grams. In the current research, bigrams, which take a set of two words as a feature, are used to improve the accuracy of the Naïve Bayes algorithm. This has not yet been implemented in the unsupervised approach.

As part of the research, we also used a few APIs to identify the orientation of sentences. Sentiwordnet is a sentiment lexicon associating the sentiment information to each wordnet synset. WordNet is a lexical database for the English language. It groups English words into sets of synonyms called synsets, provides short definitions and usage examples, and records a number of relations among these synonym sets or their members. Hence, Sentiwordnet is the combination of Wordnet and Sentiment Information. For each wordnet synset, the following information is available in Sentiwordnet: Positive score - Pos(s), Negative Score - Neg(s), Objective Score - Obj(s). These three scores sum upto 1 i.e., Pos(s) + Neg(s) + Obj(s) = 1. We used Sentiwordnet in the unsupervised method to find out the orientation of the opinion word and obtained an accuracy of 44.1% which is very low.

Twinword provides a sentiment analysis API which returns sentiment analysis results with a score for a given text. It returns the sentiment of a text along with a score. We passed each sentence to Twinword's sentiment analysis API and the API returned whether the sentence is positive or negative. We observed that 67.3% of sentences are correctly classified by this API.

Figure 3: Product concept for the proposed website

Our future plans include testing our supervised and unsupervised algorithms with other types of online reviews, such as movie reviews, hotel reviews, and service provider reviews to further refine the approaches. We are also exploring the development of a "confidence score" to help users assess the trustworthiness of our reviews. For example, a product feature with more reviews may have a higher confidence

score than one with fewer reviews. We also plan to deploy our algorithm on the web and implement a Feature based customer reviews search engine as shown in Figure 3. We also plan to offer an application programming interface for users (researchers and online firms) to test our algorithms using their own data. The web site would accept the name of a product as user input and gather the reviews about the product from various e commerce web sites like Amazon.com, Flipkart.com etc. and produce a Feature based summary from the reviews gathered. The concept is shown in Figure 3. When a user enters the product name as canon EOS 5d, the system finds 7012 reviews from websites including Flipkart.com, amazon.com, mouthshut.com, reddit.com and bestbuy.com. The summary is produced in the form of a pie chart which shows the total number of review sentences as well as the number of positive sentences and negative sentences among them.

8. References:

Gamallo, P., and Garcia, M. 2014 "Citius: A Naive-Bayes Strategy for Sentiment Analysis on English Tweets," in: *Proceedings of the 8th International Workshop on Semantic Evaluation (SemEval 2014)*, Dublin, Ireland

McAuley, J., and Leskovec, J. 2013,"Hidden factors and hidden topics: understanding rating dimensions with review text," RecSys.

Narayanan, V., Arora, I., and Bhatia, A. 2013 "Fast and accurate sentiment classification using an enhanced Naive Bayes model," in: *Intelligent Data Engineering and Automated Learning–IDEAL*, Springer, Berlin Heidelberg, pp. 194-201.

Zaroban, S. 2016 "U.S. e-commerce grows 14.6% in 2015."https://www.internetretailer.com/2016/02/17/us-e-commerce-grows-146-2015