Text and Sentiment Classification of Clothing Reviews

Krishna Nambi
University of Maryland
College Park, MD, USA
knambi@umd.edu

Abhimanyu Sachdeva
University of Maryland
College Park, MD, USA
asachde8@umd.edu

Vyjayanthi Kamath
University of Maryland
College Park, MD, USA
vkamath@umd.edu

ABSTRACT
Product reviews are highly important avenues of information for a business to improve their sales. Reviews help identify the products that are highly in demand, products that should be restocked etc. Our project will aim to describe one other insightful application of product reviews, specifically, clothing reviews. This paper will explore review sentiments and classification. The analysis conducted generated a 42% accuracy on using 23000 reviews.

Author Keywords
Sentiment Analysis, Classification, VADER Sentiment, Topic Modelling, Lexicons, Clothing Reviews

INTRODUCTION
With the advent of technology revolutionizing business process and models, online shopping and e-commerce portals have opened several doors for all types of enterprises. The next logical step in boosting their online presence seemed to be integrating consumer reviews. From previous research(1), electronic word of mouth has heavily influenced consumers in purchasing decisions leading to them preferring online ratings and reviews more than other avenues of information. Reviews are important for all product categories, but they are highly important and relevant for clothing since there are many aspects that one needs to verify before proceeding with purchase like fabric quality, color accuracy and fit. We will conduct sentiment analysis on women clothing reviews and classify them based on four aspects of clothing – fit, color, quality, and price.

DATA
The data is sourced from Kaggle. This data is provided under the CC0 Public Domain which states that one is free to copy, modify, distribute, and perform the work, even for commercial purposes, all without asking for permission.

The data includes 23486 observations and 10 feature variables. Each row corresponds to a customer review, and includes the variables:

- Clothing ID: Integer Categorical variable that refers to the specific piece being reviewed.
- Age: Positive Integer variable of the reviewers age.
- Title: String variable for the title of the review.
- Review Text: String variable for the review body.
- Rating: Positive Ordinal Integer variable for the product score granted by the customer from 1 Worst, to 5 Best.
- Recommended IND: Binary variable stating where the customer recommends the product where 1 is recommended, 0 is not recommended.
- Positive Feedback Count: Positive Integer documenting the number of other customers who found this review positive.
- Division Name: Categorical name of the product high level division.
- Department Name: Categorical name of the product department name.
- Class Name: Categorical name of the product class name.

PROBLEM STATEMENT
With consumer review-based system being implemented on various retail business platforms, business organizations have been able to evaluate the performance metrics based on the feedback from their market sales. This platform can be further utilized to identify key components from the sentiments expressed in these reviews to make smarter and informed business decisions.

The number of online consumers who read and trust online reviews is increasing. According to a survey by BrightLocal, 88 percent of consumers trust online reviews as much as a personal recommendation—which is astounding, considering most online reviews are posted by total strangers. The same survey found that only 12 percent of the population did not regularly read reviews for consumer products.

We intend to analyze the customer reviews received on women's clothing retail products and classify them on four different labels such as fit, color, quality and cost. This will help the organization to evaluate the customer feedback on the products and identify components and departments they need to improve. The type of study we are conducting is observational.

Our research question can be defined as -

Can we evaluate consumer reviews successfully by using sentiment classification?

Our study involves two unique contributions:

BRAND
With an increase in the number of small businesses selling clothes, a classifier that classifies hundreds of comments on each product would be very useful. Small businesses can

rarely afford a team to scan through all reviews to identify the features of a product that are doing well and the ones that need improvement. A classifier that tags each review with the appropriate sentiment along with the reasons for the sentiment is a good starting point for a brand, especially to focus on products that need improvement.

USER

Other reviews improve shopping experience for customers. For any customer who purchases a product, it is highly likely that they will scan reviews for the product online before making a purchase. With reviews classified as positive/negative, it is easy for customers to identify what the overall view regarding a product is, and what are the reasons for the view. For example, if a review is classified as negative because the user writing the review did not like the style, it may not be relevant for another user who is more focused on the fit, not the style. Labelling each review with the reason is therefore helpful for users to identify relevance of a positive/negative rating.

METHODOLOGY AND MODEL EVALUATION

We plan on using Python to implement sentiment analysis, topic modeling and classification.

The dataset we have obtained is not annotated with the labels we require. We plan to annotate the database as follows:

● For each review we will identify whether it is positive, negative or neutral.

● Each review will be labeled with the following tags that define the reasons for the sentiment.

○ Fit

○ Quality

○ Color

○ Cost

For example, if a review is talking about size, color as well as quality, it will be tagged with all the three labels. This will help us identify and categorize the reviews into different labels that can help the business make informed decisions on their product category and ways in which they can improve customer experience.

PREPROCESSING

Before using the training data, we have done some pre-processing on the data to improve data quality. The pre-processing steps used are same for both baseline approach and final classifier approach.
The following pre-processing steps have been performed on training as well as test data

1. **Conversion to lowercase**
   We have converted all words to lowercase before processing.

2. **Removal of numbers and some punctuations**
   We have removed all characters except alphanumeric characters, exclamation marks and dollar signs.

3. **Use of emoticons lexicon**
   Clothing reviews tend to have emoticons used to convey sentiment. It becomes difficult to train the classifier on both emoticons and sentiment words used in the review. To improve accuracy of the classifier, we have used a lexicon of emoticons, wherein we replace each emoticon by one of the terms ‘happy, sad, neutral’.

4. **Converting short word forms to correct forms**
   Reviews contain short forms of words commonly found in twitter or text messaging such as ‘gr8’ which denotes ‘great’. We have used a lexicon to convert such words to their correct spelling.

5. **Removal of stop words**
   We have removed specific stop words from the given data using a lexicon. Stop words such as ‘not’ ,’nor’ are not being removed. We need such stop words to identify sentiment.

6. **British to American English conversion**
   In clothing reviews, words like ‘color’ may be spelled with either the British spelling ‘colour’ or the American spelling ‘color’. We have standardized the spellings using a British to American English conversion lexicon.

7. **Stemming and lemmatization**
   We have used NLTK Porter Stemmer for stemming of words in training and testing data.

VADER SENTIMENT ANALYSIS

The sentiment analysis approach used was the Vader sentiment analysis technique (Valence Aware Dictionary and sentiment Reasoner). VADER is a lexicon and rule-based sentiment analysis tool that is specifically attuned to sentiments expressed in social media. Empirically validated by multiple independent human judges, VADER incorporates a "gold-standard" sentiment lexicon that is especially attuned to microblog-like contexts. VADER uses a combination of A sentiment lexicon is a list of lexical features (e.g., words) which are generally labelled according to their semantic orientation as either positive or negative. VADER has been found to be quite successful when dealing with social media texts, NY Times editorials, movie reviews, and product reviews. This is because VADER not only tells about the Positivity and Negativity score but also tells us about how positive or negative a sentiment is.

The compound score is computed by summing the valence scores of each word in the lexicon, adjusted according to the

rules, and then normalized to be between -1 (most extreme negative) and +1 (most extreme positive). This is the most useful metric if you want a single unidimensional measure of sentiment for a given sentence. Calling it a 'normalized, weighted composite score' is accurate.

It is also useful for researchers who would like to set standardized thresholds for classifying sentences as either positive, neutral, or negative. Typical threshold values (used in the literature cited on this page) are:

1. positive sentiment: compound score >= 0.05
2. neutral sentiment: (compound score > -0.05) and (compound score < 0.05)
3. negative sentiment: compound score <= -0.05

The pos, neu, and neg scores are ratios for proportions of text that fall in each category (so these should all add up to be 1 or close to it with float operation). Vader incorporates values and modifiers under various categories such as slangs, emojis, conjunctions, degree modifiers, punctuations and many more.

After performing Sentiment Mining for our reviews, we normalized the sentiment scores to 1 for positive and -1 for negative by rounding off the scores. The sentiment scores were then evaluated against the manually annotated reviews with sentiments to check for accuracy. This process helped us to identify whether a review was positive or negative and if positive and negative, we tried to identify the reason based on the 4 defined labels using our classification method

LEMMATIZATION

The goal of using lemmatization is to reduce inflectional forms and sometimes derivationally related forms of a word to a common base form.

In many languages, words appear in several inflected forms. For example, in English, the verb 'to walk' may appear as 'walk', 'walked', 'walks', 'walking'. The base form, 'walk', that one might look up in a dictionary, is called the lemma for the word. The association of the base form with a part of speech is often called a lexeme of the word.

STEMMING

Stemming is the process of reducing inflection in words to their root forms such as mapping a group of words to the same stem even if the stem itself is not a valid word in the Language.

```
In [27]: reviews_2 = lemmatization(tokenized_reviews)
         #print(reviews_2) # print lemmatized review

In [80]: #freq_words(reviews_2, 35)
         reviews_2

Out[80]: [['top',
           'stunning',
           'color',
           'cozy',
           'soft',
           'delicate',
           'everyday',
           'top',
           'material',
           'cotton',
           'candy',
           'appearance',
           'unique',
           'kind'],
          ['care',
           'label',
           'inch',
           'overall',
           'diameter',
```

Figure 1: The sentences from the reviews are lemmatized

TOPIC MODELING (LDA)

The Topic modeling approach used was the Latent Dirichlet Allocation (LDA). Topic Modeling is different from rule-based text mining approaches that use regular expressions or dictionary-based keyword searching techniques. It is an unsupervised approach used for finding and observing the bunch of words (called “topics”) in large clusters of texts.

LDA assumes the documents are produced from a mixture of topics. Those topics then generate words based on their probability distribution. Given a dataset of documents, LDA backtracks and tries to figure out what topics would create those documents in the first place.

The probabilistic topic model estimated by LDA consists of two tables (matrices). The first table describes the probability or chance of selecting a particular part when sampling a particular topic (category). The second table describes the chance of selecting a particular topic when sampling a particular document or composite. It Iterates through each word “w” for each document “d” and tries to adjust the current topic – word assignment with a new assignment. A new topic “k” is assigned to word “w” with a probability P which is a product of two probabilities p1 and p2.

The current topic – word assignment is updated with a new topic with the probability, product of p1 and p2. In this step, the model assumes that all the existing word – topic assignments except the current word are correct. This is essentially the probability that topic t generated word w, so

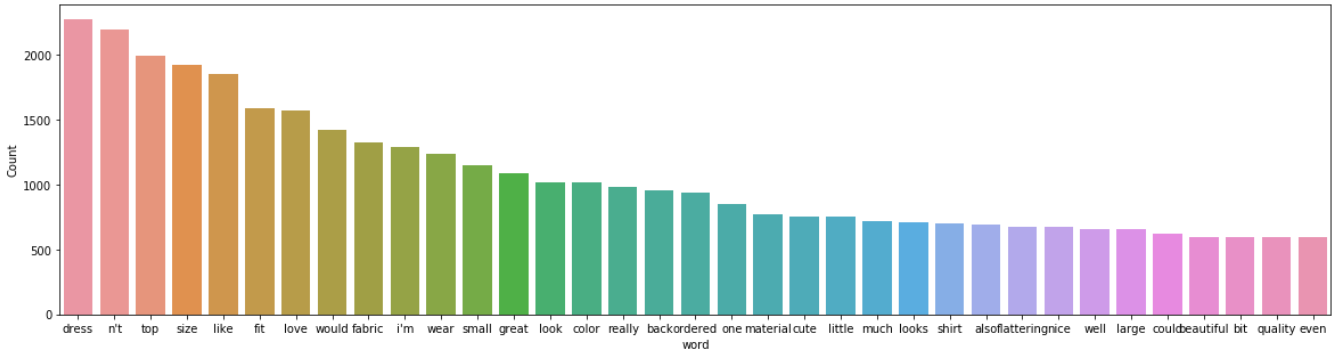

Figure 2: Top words derived from topic modelling. Words like dress, top, fabric, etc. are highly frequent

it makes sense to adjust the current word's topic with new probability.

After a number of iterations, a steady state is achieved where the document topic and topic term distributions are fairly good. This is the convergence point of LDA.

After removing stopwords and performing lemmatization we performed topic modeling to create 4 lexicons (color, cost, quality and fit).To create the lexicons we distributed the words and manually assigned each word for specific lexicon as no predefined dictionary was available for attributes like fit, quality cost and color.

COSINE SIMILARITY FOR TEXT CLASSIFICATION

Cosine similarity is a metric used to determine how similar the documents are irrespective of their size. Mathematically, it measures the cosine of the angle between two vectors projected in a multi-dimensional space. In this context, the two vectors I am talking about are arrays containing the word counts of two documents.

When plotted on a multi-dimensional space, where each dimension corresponds to a word in the document, the cosine similarity captures the orientation (the angle) of the documents and not the magnitude. If you want the magnitude, compute the Euclidean distance instead.

The cosine similarity is advantageous because even if the two similar documents are far apart by the Euclidean distance because of the size (like, the word 'cricket' appeared 50 times in one document and 10 times in another) they could still have a smaller angle between them. Smaller the angle, higher the similarity.

Once we converted our reviews into tokens and after the preprocessing procedure, we performed cosine similarity in python to check the weights of each token against the four lexicons created.

After assigning the weights for each token against the 4 defined labels, we then evaluated the average weights and identified the max weighted average amongst the 4 defined labels. This was then compared against manually classified and annotated labels to check for accuracy. The accuracy measure was reported as 43%

	Unnamed: 0	Review Text	Sentiment	Fit	Color	Cost	Quality	attribute	man attr
0	0	This top is stunning. the colors are vivid and it is extremely cozy and soft. however it seems delicate and would need to only be worn once in a while. it is not an everyday top. the material has ...	0	NaN	1.0	NaN	1	QUALITY	COLOR,QUALITY
1	1	Even following the care label exactly, it shrunk ~3 inches in the length and overall in the diameter. i had to return this since i am tall and it looked more like a work-inappropriate mini after t...	-1	1.0	NaN	NaN	1	FIT	QUALITY,FIT
2	2	I was excited to receive this top. it looked great online, vibrant colors with the beautiful detail on the sleeves. when i tried it on, the fabric looked and felt cheap. it is not a flowy top. do ...	-1	NaN	1.0	NaN	1	QUALITY	COLOR,QUALITY
3	3	Nice weight sweater that allows one to wear leggings or ultra skinny jeans without looking like i'm pregnant (not that there's anything wrong with that) very feminine and light weight enough to we...	1	NaN	NaN	NaN	1	QUALITY	QUALITY
4	4	Perfect trans top. skinnies or boyfriend and booties. love the mixed prints	1	NaN	NaN	NaN	1	QUALITY	QUALITY

Figure 3: Attribute stands for the aspect the algorithm detected and man_attr stands for aspect that was manually assigned

```
accuracy

42.02536510376633
```

Figure 4: The model generated an accuracy of 42%

REFERENCES

1. The Effect of Online Consumer Reviews on Consumer Purchasing Intention: The Moderating Role of Involvement : https://www.tandfonline.com/doi/abs/10.2753/JEC1086-4415110405
2. Proven Power of Ratings & Reviews: A Report https://www.powerreviews.com/insights/proven-power-of-ratings-and-reviews/\
3. Kaggle Dataset https://www.kaggle.com/nicapotato/womens-ecommerce-clothing-reviews
4. Do online reviews affect product sales? The role of reviewer characteristics and temporal effects: https://link.springer.com/article/10.1007/s10799-008-0041-2
5. How Online Product Reviews Affect Retail Sales: A Meta-analysis. Retrieved from https://www.sciencedirect.com/science/article/pii/S0022435914000293#bib0155

6. Agarap, Abien Fred & M Grafilon, Paul. (2018). Statistical Analysis on E-Commerce Reviews, with Sentiment Classification using Bidirectional Recurrent Neural Network. 10.13140/RG.2.2.16988.08321. Retrieved from https://www.researchgate.net/publication/323545316_Statistical_Analysis_on_E-Commerce_Reviews_with_Sentiment_Classification_using_Bidirectional_Recurrent_Neural_Network
7. BrightLocal- Local Consumer review survey https://www.brightlocal.com/learn/local-consumer-review-survey-2014/
8. David, Andrew Y. Ng& Michael I. Jordan (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research 3 (2003) 993-1022. Submitted 2/02; Published 1/03.
9. Aneesha Bakharia, September 2016. Topic Modeling with Scikit Learn. Retrieved from https://medium.com/mlreview/topic-modeling-with-scikit-learn-e80d33668730
10. Susan Li, May 2018. Topic Modeling and Latent Dirichlet Allocation (LDA) in Python. Retrieved from https://towardsdatascience.com/topic-modeling-and-latent-dirichlet-allocation-in-python-9bf156893c24