A Data Analytics Methodology for Benchmarking of Sentiment Scoring Algorithms in the Analysis of Customer Reviews

Tesneem Abou-Kassem[1], Fatima Hamad Obaid Alazeezi[1], Gurdal Ertek[1]

United Arab Emirates University, Al Ain, UAE
201640019@uaeu.ac.ae, 201601253@uaeu.ac.ae, gurdal@uaeu.ac.ae

Abstract. Due to the digitalization, there exists an increased amount of user-generated content on the Internet, where people express their opinions on various topics. Sentiment analysis is the statistical examination of people's emotions and opinions regarding a certain subject. Substantial body of research has readily been carried out on sentiment analysis. Our study extends the literature by developing and demonstrating the applicability of a data analytics methodology for the benchmarking of sentiment scoring algorithms in the context of online customer reviews. In this paper, Amazon product reviews were used as the source data. Analyzing text-based content such as Amazon customers' reviews through text analytics and sentiment analysis can help Amazon and other online retailers discover valuable actionable insights regarding their products. The contributions of this study are two-folds; to examine the predictive power of machine learning (ML) algorithms with respect to predicting scores, and to analyze patterns in the differences between scores obtained from different sentiment scoring algorithms.

Keywords: Online Customer Reviews, Sentiment Analysis, Text Analytics, Machine Learning, Gap Analysis.

1 Introduction

Massive amounts of digital data and information are captured almost every moment, regarding almost every aspect of our lives. Human behavior is strongly influenced by sentiments/emotions and beliefs, which affect judgments and decisions. Considering how different people perceive and propogate the world and its various aspects can significantly influence our decisions [1]. In the context of e-commerce, which is the domain of interest in this paper, analyzing sentiment is very important to understand customers' needs and wants, as well as improving products or services delivered to customers. Forums, blogs, customer reviews, social networks, all coexist in the ever-growing social media world, all of which can be analyzed through techniques referred to as "Sentiment Analysis" [1]. Sentiment analysis is the process of determining whether a piece of writing is positive, negative, or neutral. It can also be used to analyze a variety of different types of data, including social media posts, reviews, articles, and more [2].

There are a few different ways to perform sentiment analysis. One common way is to use a lexicon, which is a list of words that categorize text with common characteristics. Sentiment analysis determines whether the opinion is positive or negative for a topic or entity on the Internet, for topics such as economy and finances, and entities such as movies and products. The majority of social media data is unstructured because of the variety of available formats for messages, posts, and other content, and due to the easy accessibility of the social platforms. To make a decision, users typically search for and take as reference others' reviews, opinions, and experiences which can yield valuable information for users, but can also be used to mislead them.

2 Literature Review

Our study is based on data obtained from Harvard Dataverse [3], which was originally collected by Chatterjee et al. [4]. We refer to this dataset as Dataset A. The authors carried out outlier detection and sentiment analysis using the data, as a case study of Amazon customer reviews. It shows a statistics-based outlier detection and correction method (SODCM) that finds reviews and fixes their star ratings. This makes sentiment analysis algorithms better without hurting the quality of the data. Fang and Zhan [5] discussed the process of sentiment polarity categorization using both sentence-level and review-level categorizations. Furthermore, they split their work into three phases; their main work was in phases 2 and 3, where they conducted the sentiment score. The authors then did tests to compare and evaluate the results of different algorithms for scoring sentiment. Naseem et al. [6] present a large-scale benchmark Twitter dataset for COVID-19 Sentiment Analysis. They evaluated and labeled the sentiment scores as positive, negative, and neutral using the TextBlob algorithms only. As part of their sentiment classification task, they used different machine learning methods and deep learning-based classifiers. Onan [7] presented a deep learning-based architecture for sentiment analysis using Twitter product reviews, which combined glove-weighted TF-IDF word embedding with a CNN-LSTM-based architecture. Also, the author discussed how words and sentences make sense based on how they are arranged in a dictionary. This is how the orientation of a text document is found. For machine learning-based classification models, the author used labeled datasets as training sets for supervised learners. Rezaeinia et al. [8] introduce Improved Word Vectors (IWV), as a new technique to make pre-trained word embeddings in sentiment analysis more accurate. Part-of-Speech (POS) tagging techniques, lexicon-based approaches, word position algorithms, and Word2Vec/GloVe approaches were all used in their method. Mowlaei et al. [9] state that to get a better idea of how the public feels about a campaign, it is important to look at written reviews by extending two lexicon generation methods for aspect-based issues: one that uses statistical methods and the other that uses genetic algorithms to create the sentiment analysis. Al-Shabi [10] uses lexicon-based sentiment analysis as its method. It focuses on VADER [11], SentiWordNet, SentiStrength, the Liu and Hu opinion lexicon, and AFINN-111, which are among five of the most important and

well-known sentiment analysis lexicons/algorithms for Twitter data. The author's results show how well these lexicons/algorithms perform at classifying the polarity of tweets by comparing the overall accuracy of classification with the F1 measure.

3 Methodology

The objective of the study presented in this paper is to extend the methodological and practical body of knowledge in sentiment analysis, in the context of online customer reviews. To this end, an application-oriented data analytics methodology has been developed, documented, and implemented using real-world data.

The methodology developed for and applied during the study is given in Figure 1. Firstly, the source data is processed and prepared for analysis. This preparation step also includes the engineering, computation, transformation, and generation of existing and new attributes. Secondly, standard text analytics steps are followed to analyze the text corpus (collection), which is the collection of online customer reviews at Amazon.com. Thirdly, word frequency tables are used in conjunction with sentiment scores for the purpose of predictive analytics to benchmark the various algorithms. Finally, gaps between scores generated by two sentiment scoring algorithms are analyzed.

Using Natural Language Processing (NLP) as the text analytics technique, Python programming language, and KNIME data analytics platform to explore trends and sentiment analysis in Amazon's customer reviews toward a specific product can enable various insights. The presented can help in understanding what/how customers feel about their purchases, what they like and dislike about products, which factors are highly associated with sentiments, which sentiment scoring algorithms generate scores that have the highest performance with respect to predictability, and how gaps between gaps between sentiment scores of different algorithms can be analyzed.

For the data preprocessing, we used KNIME software to preprocess and clean the textual data. For the text analysis, we examine the perfume product dataset to find the most frequent words that have occurred in the reviews using Lancaster and Porter libraries in Python as well as N-gram analysis for the positive and negative reviews. For the sentiment analysis, we apply three different sentiment scoring algorithms/implementations, namely the VADER Python library [11], TextBlob Python library [12], and KNIME Lexicon-based algorithm [13]. For the model evaluation, we use the random forest machine learning method to compare the prediction performance of the three methods. And lastly, we conduct a gap analysis between the VADER and TextBlob sentiment results and examine which factors are related to gaps in the sentiment scores of the selected two algorithms.

Two critical aspects and also steps of the methodology are (a) the prediction of sentiment scores obtained by different algorithms, and (b) analysis of gap between the sentiment scores obtained by different algorithms. To conduct (a), a new Dataset C was created, that combines Dataset B, which includes sentiment scores obtained through various algorithms, together with the data of term frequencies in each document.

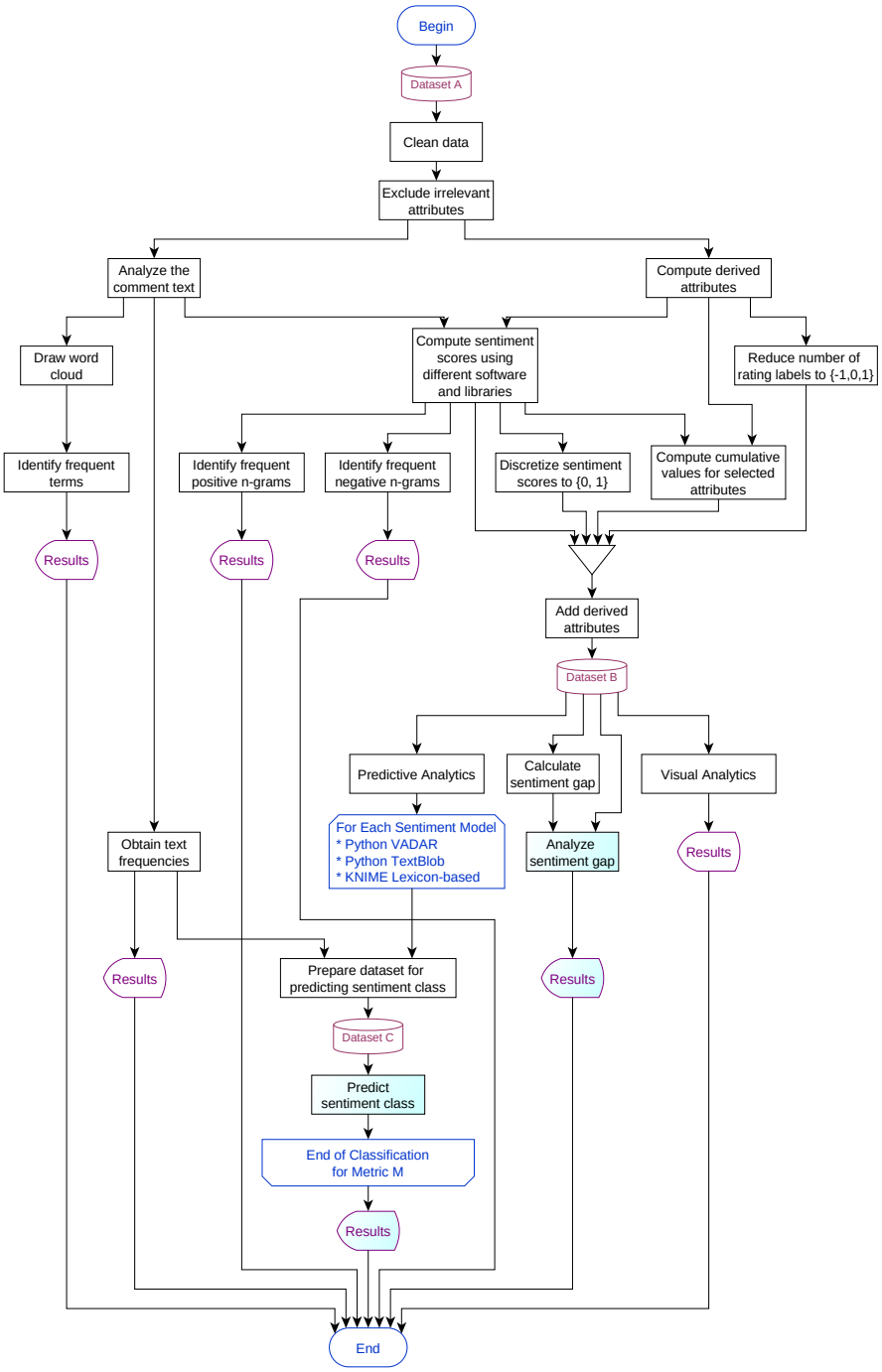

Fig. 1. The data analytics methodology applied in the presented research study.

4 Data

The original Amazon Product reviews dataset, which we refer to as Dataset A, was collected from Amazon.com by Ishani Chatterjee [4], and publicly shared online [3]. The data is separated into seven different CSV files, where each file includes data for a different product, Perfume, Book, Mask, Movie, Food, Curcumin, and Electronic. The reviews in each dataset were created between 2008 and 2020 and each of them has a collection of 5,000 reviews and 8 attributes. Each row in the dataset includes a review from an individual customer as well as additional review information such as ratings. Dataset A lists and describes the attributes included in the datasets, Product name, Ratings, Reviews, Helpful, Date, Asin, Target, and Text.

Using this source Dataset A, after excluding irrelevant attributes, completing sentiment scores, discretizing sentiment scores, and deriving new attributes, especially for cumulative values, a new Dataset B is obtained. While Dataset B has many attributes, the scope of the current study was limited to only some of these attributes, as a first step. One of the future research possibilities is to enrich and extend the current methodology to become much more comprehensive, yielding much richer insights by design. Still, the full meta-data for Dataset B is readily provided in Table 1 in this paper, to lay the founodation for future studies, as well as motivate other researchers to work with this readily enriched dataset.

Table 1. Shows the attributes in Dataset B, other types and brief descriptions.

No.	Attribute	Type	Description
1	RowID	Numerical	unique ID for each review (each row is a review)
2	ProductNumber	Numerical	number of each review for each product
3	ASIN	Numerical	Amazon Standard Identification Number
4	ProductName	Categorical	name of the product
5	Ratings	Numerical	rating of the product in that review: 1-5 (Likert scale)
6	RatingClass	Binary	rating class; Positive rating (1): 4-5; Negative rating (0): 1-3
7	Review	Text	customers review
8	WordCount	Numerical	number of words in each review
9	CharacterCount	Numerical	number of characters in each review
10	Helpful	Numerical	how helpful the review is for other customers
11	Date	Date type	date of the review
12	Year	Numerical	year of when the review was written
13	Month	Numerical	month of when the review was written
14	Day	Numerical	day of when the review was written
15	DateCode	Numerical	unique code of date of the review
16	DateGap	Numerical	number of gap days from the first review to the date of this review
17	DaysSinceLastReview	Numerical	number of days past since the last review
18	Target	Categorical	targeted reviews: Positive or Negative

19	ProductType	Categorical	product type/category (Food, Books, Masks, Perfume, Curcumin, Electronics, Movies)
20	CumulAvgRating	Numerical	average of all the ratings until this review, excluding this review
21	CumulSumHelpful	Numerical	summation of helpful values for all reviews until now
22	CumulSumWordCount	Numerical	summation of wordcounts of all reviews until now
23	CleanedReview	Text	review text after text cleaning and preprocessing
24	ScorePythonVADER	Numerical	sentiment score of the Python library VADER [-1,1]
25	SentimentPythonVADER	Numerical	sentiment label computed in VADER library of Python, for the text in Review {0, 1}
26	ScorePythonTextBlob	Numerical	sentiment score of the Python library TextBlob [-1, 1]
27	SentimentPythonTextBlob	Numerical	sentiment label computed in TextBlob library of Python, for the text in Review {0,1}
28	KNIMENegativeWords	Numerical	number of negative words from the preprocessed review
29	KNIMEPositiveWords	Numerical	number of positive words from the preprocessed review
30	WordCountCleanedReview	Numerical	Number of words in the preprocessed review
31	SentimentScoreKNIME	Numerical	sentiment score of KNIME [-1,1]
32	SentimentKNIME	Numerical	sentiment label computed in KNIME for the text in Review {0, 1}

5 Analysis and Results

5.1 Data Preprocessing

For data preprocessing, we will examine different approaches using the KNIME Platform to simplify the analysis and make the textual data ready for sentiment analysis without any noisy words or text errors. Some of the different KNIME nodes that were used in the text preprocessing are Case Convertor, Punctuation Erasure, Stope Word Filter, Dictionary Filter, N Chars Filter, and Number Filter. In addition, we have also filtered the infrequent terms that have occurred less than 10 times by using the Bag of Words (BoW) and GroupBy nodes.

Feature Selection and Engineering. Since we are only interested in the customer reviews and their associated ratings, we will drop the other attributes (Product name, ASIN, Target, and Text) from the datasets and develop new variables that could create informative insights. The sentiment is determined by the customer's rating based on a scale of 1 to 5 (5 being the most favorable). As we are using classification methods to classify customer reviews, these scores will need to be converted into two categories, namely 1 and 0. Ratings above and including 4 will be labeled as positive reviews “1”. Ratings with a score of 3 and below will be labeled as negative Reviews “0”.

Other features have been added to the dataset that could be contributed to the analysis of the customer review data, such as Cumulative Average Rating, Word Count, which is the number of words in each review, Cumulative Sum of Word Count, Character Count, Date Gap, and Day Since Last Review.

5.2 Text Mining

Most Frequent Words. Figure 2 displays the most frequent terms occurring in the customer's reviews for the perfume product. The word cloud has been generated by deleting stop words, such as "that," "the," and pronouns, as well as frequent words like "perfume," "product", and "amazon," which naturally occur in big portion of the reviews.

Fig. 2. Word cloud of the perfume product most frequent words that have occurred in the customer reviews.

It is observed immediately from Figure 2 that the term set retrieved from the reviews is mostly positive, which makes sense since the number of positive reviews is much higher than non-positive reviews. Most reviews/comments discuss the characteristics of the perfume, like the smell, scent, how long it lasts, etc. There are also terms indicating the feelings of customers, such as love and like.

5.3 Sentiment Scoring

Sentiment analysis is one of the most common core areas where NLP has been used. Businesses need to know how customers act and what they expect from the products and services they buy. Sentiment scoring is the process of assigning a score to a text document, usually between -1.0 and 1.0, where a score of 1.0 indicates a very positive sentiment and a score of -1.0 indicates a very negative sentiment. The outcome of the score is calculated based on the number of positive and negative words in the document, and the way those words are used (e.g., whether they are used in a positive or negative context). The feedback a customer gives about a product can be "positive" or "negative." Interpreting feedback from customers through ratings and reviews enables

businesses to measure customer satisfaction with their products and services. In addition to analyzing the polarity of a text, it can also identify certain sentiments and emotions, such as anger, happiness, and sadness. Even intentions, such as whether a person is interested or not, may be deduced using sentiment analysis.

For the sentiment analysis, several sentiment methods were conducted using the Python programming language and the KNIME Platform. Our goal is to compare the performance of each method and find out which one of them has the most accurate performance in classifying customer reviews.

For the Python programming language, VADER and TextBlob sentiment analysis libraries were conducted using the Natural Language Processing (NLTK) package to determine the text's mood.

VADER Sentiment Analysis. VADER (Valence Aware Dictionary and Sentiment Reasoner), which is a rule-based and lexicon-based pre-built library in NLTK, is one of the best choices for sentiment analysis in Python. This library, which was developed particularly for social media sentiment analysis [11], includes a sentiment lexicon and a collection of lexical properties that are commonly categorized according to their sentiment polarity as either positive or negative.

VADER computes text sentiment and returns the likelihood that a particular input statement is positive, negative, or neutral. The library returns a compound score, which is also known as a polarity score, which is a measure that calculates the total of all normalized lexicon ratings between -1 (extremely negative) and +1 (extremely positive). To label the sentiment scores as positive and negative, we have classified the polarity scores as positive sentiment (polarity score > 0), and negative sentiment (polarity score <= 0).

TextBlob Sentiment Analysis. TextBlob is an ideal substitute for sentiment analysis. The basic Python library provides extensive textual data analysis and processing. TextBlob defines a sentence's mood based on its sentiment polarity and the intensity of each word, which requires a predefined dictionary to distinguish negative and positive terms. The tool gives each word a separate score and calculates what the overall emotion is [12]. TextBlob returns a sentence's polarity and subjectivity, with polarity ranging from negative to positive.

KNIME Sentiment Analysis. As part of KNIME's text processing feature, textual data was read, processed, and transformed into numerical data (documents and term vectors) to be used in regular KNIME data mining nodes for classification [13]. KNIME can analyze and parse texts in different formats and store the results in a table. In this way, the document can be further semantically enhanced by recognizing and tagging different kinds of named entities, such as those with positive and negative sentiments. Documents can be filtered in many ways, such as by using stop words nodes or named entities, stemming with stemmers that work with more than one language, and preprocessing in many different ways. Furthermore, it is possible to compute the frequency of words, extract keywords, and do some visualization in KNIME. Based on the document sentiment results, you can apply regular KNIME nodes to classify documents using

numerical vectors. In this paper, we used the MPQA subjectivity lexicon [14] to identify contextual polarity depending on the Lexicon-based approach.

5.4 N-Grams

N-Grams are combinations of "n" words within a sentence that can play an important part in text categorization and language modeling. In this analysis, the “ngram” method in the NLTK library is used to discover all n-grams in the review texts.

N Grams for Positive Reviews. Using the N-Gram python method ranging from 1 to 3 grams, we have divided the positive and negative VADER sentiment score results separately to see what are the terms that are most repeated in each of the sentiment reviews. The results of the N Grams for the VADER positive sentiments for the perfume product, indicated that most of the customers feel good about purchasing the perfume, where the number of times the words great and good have occurred more than 1,000 times. Moreover, we observed that people also think about the scent as being fresh, lasting a long time, and smelling good and great as most of the written “smells great”, “love smell”, “fresh scent”, and “smells pretty good”. As well as the price of the product, where some of them have written that the price of the perfume is great such as “great price”.

N Grams for Negative Reviews. These negative reviews of the perfume product, people who wrote these negative reviews believe the product is fake or smells bad and these words have occurred more than 50 times. Others have also given negative reviews because they have received broken perfume. But compared to the number of these terms that have occurred, it’s not comparable to the number of the positive terms and how much they have occurred.

6 Sentiment Prediction

For the comparison of sentiment analysis between the three methods (VADER, TextBlob, and Lexicon-based algorithm), we labeled the sentiment scores for each product's reviews as either positive or negative. Text mining was carried out to identify the most frequent terms, which in turn were considered as predictive features/attributes (columns in a tabular dataset) whose term frequency values were used for sentiment prediction. The classification algorithm used was the random forest machine learning algorithm, which enabled the comparison of the predictability of sentiments from the three sentiment scoring lexicons/algorithms. We would like to check which machine learning algorithm model has the highest accuracy in predicting the sentiment of a customer review for each of these three algorithms.

6.1 Reviews Sentiment Prediction Accuracy Comparison

Table 2 displays the sentiment prediction accuracy results using random forest classification. By looking at the accuracy of the product review sentiment prediction, we can

figure out which algorithm is the most accurate and works best. As it is shown below, VADER algorithms got the highest accuracy for the perfumes and masks datasets. However, TextBlob algorithms performed well for the curcumin and movie datasets. While KNIME algorithms worked accurately with books, electronics and food datasets. This shows that the three methods are good predictors for sentiment analysis since classification accuracies for all three methods are close to each other. However, to obtain the highest classification accuracy for different products, all three algorithms can be considered.

Table 2. Sentiment analysis methods accuracy comparison

Products	VADER Sentiment Method	TextBlob Sentiment Method	KNIME Sentiment Method
Perfume	**0.938**	0.921	0.915
Books	0.886	0.888	**0.906**
Curcumin	0.898	**0.902**	0.865
Electronics	0.887	0.885	**0.906**
Food	0.893	0.891	**0.928**
Masks	**0.907**	0.905	0.904
Movies	0.874	**0.878**	0.865

6.2 Sentiment Score Gap Analysis

The sentiment score gap analysis was conducted to find the difference between the results of the VADER and the TextBlob sentiment scores. By calculating the gap between the sentiment score results of the two methods (VADER score minues TexBlob score), we first analyze the correlation between the sentiment gap (y axis) and the other continuous variables for all the products (x axis). The results of the correlation suggested that in most of the products, the sentiment gap has a positive correlation with the word and character count of the review. Figure 3 depicts as scatter plots, the relationship between the sentiment gap (y axis) and the word count of the review (y axis). As we look into the scatter plots for the seven selected products, we can notice a consistent patterns: As the number of words in the review increases, it's more likely to be labeled as a positive review by VADER, compared to TextBlob. Also, the number of positive reviews for all the products is much higher than the number of negative reviews. There are many other analyses that have been and can be conducted, yet the content of this paper was kept limited to the analysis of only one gap analysis relation, due to the paper's space limitations.

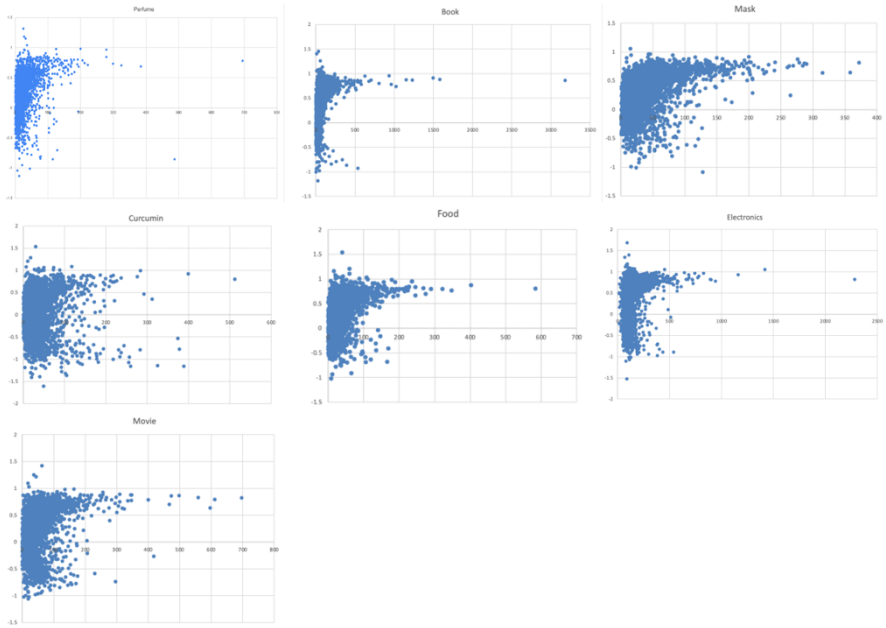

Fig. 3. Scatter plots of the relationship between the sentiment gap of VADER and TextBlob sentiment scores and the word count of the review for all the products.

7 Conclusion and Future Work

In this paper, we primarily focused on sentiment mining basics and their levels. The identification of sentiment from content can be achieved in several different ways. Sentiment analysis analyzes people's sentiments, attitudes, and emotions toward certain entities. In this paper, we addressed sentiment polarity categorization as a fundamental problem in sentiment analysis, which we focused on by categorizing customer/user opinions on select Amazon products as positive or negative. Furthermore, we studied the differences between three different sentiment algorithms (VADER, TextBlob and KNIME). Moreover, we used the correlation matrix to extract potentially useful and actionable insights, which motivated the gap analysis.

References

1. Mehta P, Pandya S. A review on sentiment analysis methodologies, practices and applications. International Journal of Scientific and Technology Research 9(2):601-9 (2020).
2. Oxford languages, https://languages.oup.com/, last accessed 2022/10/19
3. Harvard Dataverse, https://doi.org/10.7910/DVN/W96OFO, last accessed 2022/09/14.
4. Chatterjee I, Zhou M, Abusorrah A, Sedraoui K, Alabdulwahab A. Statistics-Based Outlier Detection and Correction Method for Amazon Customer Reviews. Entropy. 2021; 23(12):1645. https://doi.org/10.3390/e23121645.

5. Fang X, Zhan J. Sentiment analysis using product review data. Journal of Big Data. 2015 Dec;2(1):1-4.
6. Naseem U, Razzak I, Khushi M, Eklund PW, Kim J. COVIDSenti: A large-scale benchmark Twitter data set for COVID-19 sentiment analysis. IEEE Transactions on Computational Social Systems. 2021 Jan 29;8(4):1003-15.
7. Onan A. Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks. Concurrency and Computation: Practice and Experience. 2021 Dec 10;33(23):e5909.
8. Rezaeinia SM, Rahmani R, Ghodsi A, Veisi H. Sentiment analysis based on improved pre-trained word embeddings. Expert Systems with Applications. 2019 Mar 1;117:139-47.
9. Mowlaei ME, Abadeh MS, Keshavarz H. Aspect-based sentiment analysis using adaptive aspect-based lexicons. Expert Systems with Applications. 2020 Jun 15;148:113234.
10. Al-Shabi MA. Evaluating the performance of the most important Lexicons used to Sentiment analysis and opinions Mining. IJCSNS. 2020 Aug;20(1):1.
11. vaderSentiment, https://pypi.org/project/vaderSentiment/, last accessed 2022/10/19.
12. TextBlob, https://textblob.readthedocs.io/en/dev/, last accessed 2022/10/19.
13. Bessa, A., Lexicon-Based Sentiment Analysis: A Tutorial, 2022/03/17 https://www.knime.com/blog/lexicon-based-sentiment-analysis, last accessed 2022/09/14.
14. MPQA Opinion Corpus Release Page, https://mpqa.cs.pitt.edu/corpora/mpqa_corpus/, last accessed 2022/10/19.