WSCAR: Web Scraping and Contextual analysis of Reviews

Sri Sai Charan Veliseti, Pranav Limbekar, Vaishanth Ramaraj
University of Maryland
7999 Regents Dr, College Park, MD 20740
svellise@umd.edu,pranav05@umd.edu, vrmrj@umd.edu

Abstract

The majority of people use online products, so we looked at how people felt about them based on reviews of those products in the current technological era. They give their feedback and based on that and other factors, products are then recommended for purchase and sale. Users are able to rate and review products on major e-commerce sites like Flipkart, Myntra, Amazon, and many others. Before purchasing a product, the consumer will conduct research to gain a better understanding of how the product functions. The interpretation will consist of a very straightforward product that has positive, neutral, and negative Product checks. We might conduct this experiment using machine learning techniques. Consumers who are aware of a product reaction participate in research known as sentiment analysis. The data is collected using a Kaggle of Amazon product reviews. To classify feedback and achieve the highest level of precision, we employ various Logistic Regression, Naive Bayes, and Random Forest methodologies. The Random Forest machine learning algorithm, out of all the ones used, is the most accurate, according to our research.

1. Introduction

Everything is moving online in the current technological and digital age. People prefer shopping online for items like food, clothing, and electronics to going outside. E-commerce platforms have thus increased significantly. On these platforms, various products from various brands are offered. Choosing a trustworthy and useful product will therefore be very challenging. An individual reads product reviews in order to find a useful product, comprehend it, and make a purchasing decision. A user will always check reviews of a product before making a purchase when shopping online. The opinions and experiences of others are more trusted by the user. Most of the time, people only base their decisions to purchase or return a product on reviews. To read through thousands of reviews each time a person considers purchasing a product, however, will be quite challenging. It will be advantageous to glean some relevant information from these reviews. Machine learning is useful in this situation. One can learn about a person's sentiment or opinions about a product using the computational technique known as sentiment analysis. It is, in essence, a classification process that highlights the fact that a particular review can express a neutral, adverse, or favorable feeling. It is, in essence, a classification process that emphasizes how a particular review can convey a neutral, adverse, or positive feeling. This field of study is more well-liked now than it ever was in the earlier days of the internet. However, there are some drawbacks to this volume of feedback. The original product is not guaranteed or has a guarantee of it, despite the fact that all of these online reviews claim to. This is due to the possibility of fake users posting comments and opinions. The second drawback is the lack of availability of the majority of online reviews and reviewers. This can occasionally lead to the closure of these shopping platforms. Sentiment analysis can be used to review structural products, determine what customers like and dislike, compare opinions of our products to those of our competitors' products, understand information about current products, and save several hours of physical conversion of time. The primary goal is to categorize user sentiments in the reviews into positive and negative ones so that the user can decide quickly whether to buy a product or not. To categorize the reviews, however, a number of methods have been used. Several of the strategies are covered in this paper.

2. Problem statement

The goal of this project is to develop an application that can analyze reviews posted by numerous customers on Amazon for a product and make recommendations based on those reviews. When a customer searches for a product on Amazon, they take a lot of time reading the reviews and determining whether the item is as good as claimed. In order to help the customer make an informed choice, our application would reduce that time by automatically recommending the product to them. Natural Language Processing

(NLP)[4] algorithms and web scraping will both be used by our application to analyze the reviews. The project is intriguing because our application offers a real-world use case and because our application is accessible by millions of customers who use Amazon to shop for their products. Our project, where we plan to use cutting-edge ML algorithms to process reviews and offer suggestions, faces the challenge of judging customer reviews.

3. Related Works

Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data. The goal is a computer capable of "understanding" the contents of documents, including the contextual nuances of the language within them. The technology can then accurately extract information and insights contained in the documents as well as categorize and organize the documents themselves. In particular, we are applying this technology to product reviews by searching for two primary events. The first is the occurrence of the product's attribute that we are interested in (eg: an electronic device's battery life/durability), and the second is the contextual clues present in the user reviews. Whether these are positive or negative, if positive then by what margin, and if negative then by what margin? Collecting such contextual information from a vast number of reviews gives us the ability to generate a prediction.

Various algorithms have been developed to perform Natural Language Processing(NLP)[4] efficiently and reliably. After analyzing various papers on NLP algorithms, we have decided to implement BERT [6] transformer algorithm. BERT[6], which stands for Bidirectional Encoder Representations from Transformers. Unlike recent language representation models, BERT[6] is designed to pretrain deep bidirectional representations from the unlabeled text by jointly conditioning on both the left and right context in all layers. As a result, the pre-trained BERT[6] model can be fine-tuned with just one additional output layer to create state-of-the-art models for a wide range of tasks, such as question answering and language inference, without substantial task-specific architecture modifications.

BERT[6] is conceptually simple and empirically powerful. It obtains new state-of-the-art results on eleven natural language processing tasks, including pushing the GLUE[8] score to 80.5% (7.7% point absolute improvement), MultiNLI accuracy to 86.7% (4.6% absolute improvement), SQuAD v1.1 question answering Test F1 to 93.2 (1.5 points absolute improvement) and SQuAD v2.0 Test F1 to 83.1 (5.1 points absolute improvement).

3.1. Naive Bayes

Naive Bayes [3] is one of the most common generative learning algorithms for classification problems. This algorithm assumes that x is conditionally independent given y, which is called the Naive Bayes[3] assumption.

3.2. LSTM

Deep learning and artificial intelligence both use long short-term memory (LSTM [5]), a type of artificial neural network. The LSTM[5] has feedback connections, in contrast to conventional feedforward neural networks. Such a recurrent neural network (RNN) can analyze whole data sequences as well as individual data points, such as photographs (such as speech or video). For instance, LSTM[5] can be used for applications like linked, unsegmented handwriting identification, speech recognition, machine translation, robot control, video games, and medical imaging. The LSTM[5] neural network has amassed the most citations from the 20th century. Because a normal RNN has both "long-term memory" and "short-term memory," the moniker "LSTM" alludes to this.

A cell, an input gate, an output gate, and a forget gate make up a typical LSTM unit. The three gates control the flow of information into and out of the cell, and the cell remembers values across arbitrary time intervals. Since there may be lags of uncertain length between significant occurrences in a time series, LSTM networks are well-suited to categorizing, processing, and making predictions based on time series data. To solve the vanishing gradient issue that can arise when training conventional RNNs, LSTMs were created.

3.3. SVM

The supervised(feed-me) machine learning algorithm known as SVM [1] can be applied to classification or regression problems. Regression predicts a continuous value, whereas classification predicts a label or group. By locating the hyper-plane that distinguishes the classes we plotted in n-dimensional space, SVM accomplishes classification. By converting our data with the aid of mathematical operations referred to as "Kernels," SVM creates that hyperplane. Linear, sigmoid, RBF, non-linear, polynomial, etc. are some types of kernels. When there is no prior knowledge of the data, the general-purpose kernel Kernel — "RBF" is employed as a tuning parameter for non-linear situations. Kernel — " linear" is for issues with linear separability. We will use "linear SVM" since our problem is linear (consisting only of positive and negative values).

3.4. BERT

Bidirectional Encoder Representations from Transformers, or BERT[6], is a new paper from Google AI

Language researchers. By offering cutting-edge findings in a wide range of NLP tasks, such as Question Answering (SQuAD v1.1), Natural Language Inference (MNLI), and others, it has stirred up controversy in the machine learning community.

The primary technological advancement of BERT[6] is the application of Transformer's bidirectional training, a well-liked attention model, to language modeling. In contrast, earlier research looked at text sequences from either a left-to-right or a combined left-to-right and right-to-left training perspective.

Models	Training Acc.	Val Acc.
BERT	98.80%	89.32%
LSTM	90.98%	85.34%
Multinomial NB	79.42%	70.30%
SVM	82.20%	67.50%

Table 1. Algorithm Comparison for sentiment analysis

4. Data

We have extracted data from URLs that the user inputs. This is done through web-scrapping and is explained in detail in Section 4.1. We briefly cover how data is pre-processed and what benchmarks we have used to test and improve our model.

4.1. Web Scraping

Several python modules are available to perform web scraping. The most widely used is a package called "BeautifulSoup". This package provides all the necessary tools to handle HTML content processing to extract text from specific HTML tags by describing the attributes. We have built a custom package to scrape review data for an Amazon Product by searching through all the review pages. Our code fetches features like a product review, username, review date, review rating, etc... We then implement NLP algorithms on the review text and used review ratings as the labels.

Web scraping requires two parts, namely the crawler and the scraper. The crawler is an artificial intelligence algorithm that browses the web to search for the particular data required by following the links across the internet. The scraper, on the other hand, is a specific tool created to extract data from the website. The design of the scraper can vary greatly according to the complexity and scope of the project so that it can quickly and accurately extract the data.

4.2. Pre-Procesing

We have the web-scrapped data for the product that is searched. We first clean the data by removing all special characters and emojis. We then Map the total review rating of each review from -1 to +1. Where -1 is labeled as a negative sentiment and +1 as a positive sentiment. This is how we create labels for our data as we are creating a unique data set each time a product is searched.

4.3. GLUE: Benchmark

A group of tools are included in the General Language Understanding Evaluation (GLUE) standard for natural language understanding systems that may be used to train, test, and analyze these systems. GLUE includes: Using known existing datasets as a foundation, nine benchmark tasks for interpreting sentences or phrase pairs were chosen to represent a variety of dataset sizes, text genres, and levels of difficulty. An evaluation and analysis of model performance with regard to a variety of linguistic phenomena seen in natural language, and a dashboard for visualizing model performance on the diagnostic set and a public scoreboard for monitoring performance on the benchmark.

Any system capable of processing sentences and phrase pairs and delivering appropriate predictions is allowed to compete since the GLUE benchmark's structure is model-neutral. The benchmark problems are chosen to encourage models that use parameter sharing or other transfer learning approaches to communicate knowledge across tasks. The main objective of GLUE is to stimulate research into the creation of comprehensive and reliable natural language understanding systems.

4.4. SQuAD v1.1

The Stanford Question Answering Data set (SQuAD[7]) comprises crowd-sourced questions on a collection of Wikipedia articles, with each question's response being a span of text taken from the associated reading passage.

5. Methods

We have two primary aspects to cover, the generation of the best prediction model Fig: 1, the integration of text summarize, and a GUI to complete our implementation Fig: 2. In our prediction pipeline, we first pre-process the amazon reviews that have been extracted, then clean and map them as mentioned in section 4.2. We then split the dataset into train validation and test data. The training and validation data is passed through a Pre-trained Bert tokenizer and Then passed through our implementation of the BERT model, where the loss is calculated and the weights are updated. Once our model is done training we use the test data to evaluate our accuracy. Over multiple epochs, the model which achieves the best accuracy is then stored.

The best classifier model is used to generate a prediction on users' sentiments towards a particular product, in Fig: 2 we can see how the best classification model from

our prediction pipeline is used to generate a Product recommendation score which is based on the average predicted sentiment of all the reviews of the searched product.

Furthermore, the positive classified reviews are then passed through a Pegasus Tokenizer [9] which generates a summary of all the classified reviews.

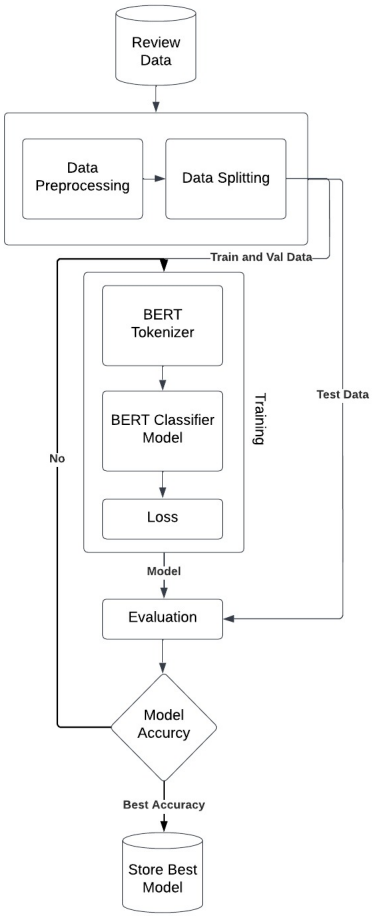

Figure 1. Prediction Pipeline

6. Experiments

6.1. Visualization

6.1.1 BertViz

The BertViz tool visualizes attention as lines connecting the position being updated (left) with the position being attended to (right). Colors identify the corresponding attention head(s), while line thickness reflects the attention score. At the top of the tool, the user can select the model layer, as well as one or more attention heads (by clicking on the color patches at the top, representing the 12 heads).

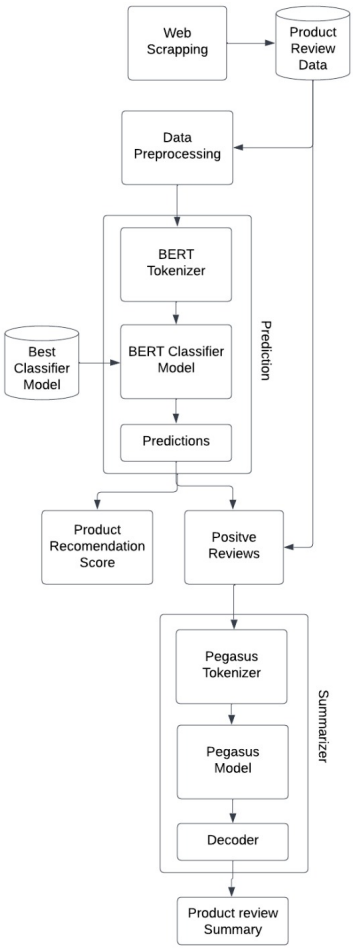

Figure 2. Algorithm Overview

6.1.2 GUI

We have developed a GUI that allows users to search for a particular product on amazon. Then it displays the positive sentiment score which indicates how positively received is the amazon listing. The GUI also provides a summary of all the extracted reviews from the amazon URL. As shown in Fig: 3.

6.2. Evaluation Metrics

6.2.1 F1 - Score

Whenever the accuracy metric is used, we aim to learn the closeness of a measured value to a known value. It's therefore typically used in instances where the output variable is categorical or discrete — Namely a classification task. In instances where we are concerned with how exact the model's predictions are we would use Precision. The pre-

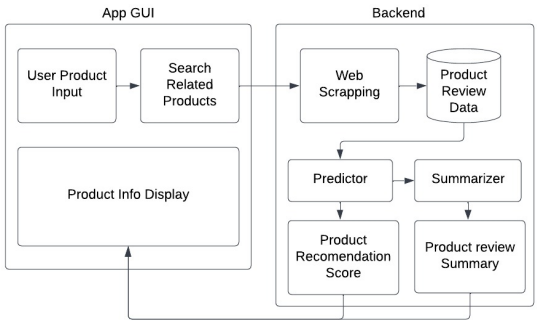

Figure 3. GUI Overview

cision metric would inform us of the number of labels that are actually labeled as positive in correspondence to the instances that the classifier labeled as positive. Recall measures how well the model can recall the positive class (i.e. the number of positive labels that the model identified as positive). Precision and Recall are complementary metrics that have an inverse relationship. If both are of interest to us then we'd use the F1 score to combine precision and recall into a single metric. Compared to the raw accuracy metric, the F1 score provides more robust evaluations of imbalanced datasets in

Given the True Positives (TP), False Positives (FP), True Negatives (TN), False Negatives (FN) between ground truth and predictions, Following is the mathematical formulation of F1 Score:

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

	Precision	Recall	F1- Score
Negative	95.00%	92.00%	93.00%
Positive	92.00%	95.34%	94.00%
Accuracy	-	-	93.00%
Macro Avg	93.00%	93.00%	93.00%
Weighted Avg	93.00%	93.00%	93.00%

Table 2. Evaluation Metric

6.2.2 Confusion Matrix

An algorithm's performance may be seen using a particular table structure called a confusion matrix, also known as an error matrix. This type of method is commonly one for supervised learning (in unsupervised learning it is usually called a matching matrix). Both variations of the matrix, where each row represents examples in an actual class and each column represents instances in a predicted class, are documented in the literature. The name was chosen since it is simple to determine whether the system is conflating two classes (i.e. commonly mislabeling one as another).

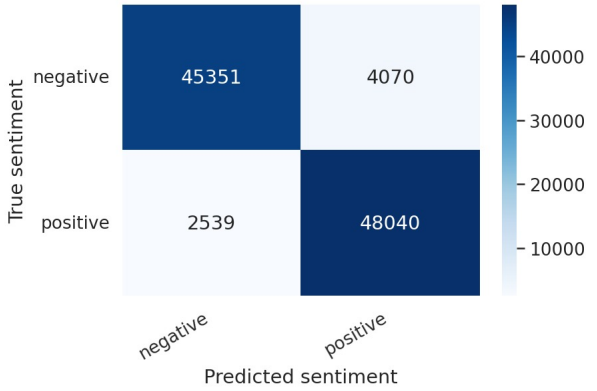

Figure 4. Confusion Matrix

7. Limitations

7.1. Sarcasm

People use positive phrases to convey their negative feelings in sarcastic writing. Because of this, sentiment analysis models can be readily tricked by sarcasm unless they are specially created to take this possibility into consideration. There are different types of sarcasm such as Propositional: Sarcasm appears to be a non-sentiment proposition but has an implicit sentiment involved. Embedded: Sarcasm has an embedded sentiment incongruity in the form of words and phrases themselves. Like-prefixed: A like-phrase provides an implied denial of the argument being made. Illocutionary: Non-speech acts (body language, gestures) contributing to the sarcasm [2].

7.2. Tone

It may be challenging to determine tone when speaking, and it can be even more challenging when writing. When attempting to examine a sizable amount of data that may include both subjective and objective replies, things get much more challenging. Finding subjective thoughts and correctly assessing them for their intended tone may be challenging for brands.

7.3. Polarity

Words with strong positive (+1) and negative (-1) polarity ratings include "love" and "hate." These are simple to comprehend. However, there are word conjugations that fall in the middle of the polarity spectrum, such as "not so awful," which may also indicate "average" (-75). These kinds of sentences are occasionally omitted, which lowers the sentiment score.

7.4. Emojis

Emojis are overused in text-based social media posts, such as those on Twitter, which is a concern. Language-specific tasks are drilled into NLP tasks. Emojis are a language in and of themselves, even though they can extract words from simple photographs. Emojis are often treated as special characters that are deleted from the data during sentiment mining in most emotion analysis software. However, doing so will prevent businesses from gaining comprehensive insights from the data.

7.5. Idioms

A figure of speech isn't necessarily understood by machine learning algorithms. The algorithm, which only understands things in their literal sense, will be confused by expressions like "not my cup of tea," for instance. Therefore, when an idiom is used in a remark or a review, the algorithm may misinterpret the language or even disregard it. A sentiment analysis platform must be educated to comprehend idioms in order to solve this issue. This issue multiplies when there are numerous languages involved.

7.6. Negations

Negative statements made by words like "not," "never," "cannot," "were not," etc. can throw the ML model off. For instance, a machine algorithm must comprehend that the statement "I can't not go to my class reunion" denotes a person's intention to attend the reunion. To solve this it takes training for a sentiment analysis tool to recognize that double negatives outweigh one another and make a text positive. This can only be accomplished when the algorithm has had enough training data and has access to as many negation words as practical, allowing for the greatest amount of possibilities and permutations.

7.7. Audio-visual data

Contrary to written data, videos are not the same. In addition to needing to be captioned, videos may also contain company logos that need to be looked for in order to accurately transcribe them. In addition to the video information, social media videos also include comments. If a sentiment analyzer can effectively extract information from the video as well as text, it will be able to provide reliable insights from your data. To do this, it requires a model for breaking down video content into entities that can be extracted as well as information about brand logos, consumer opinions, and product insights.

8. Conclusion

We have created a novel GUI application that leverages the prediction powers of pre-trained BERT applied to our processed dataset which results in a 93% accuracy for the case of amazon reviews, which has achieved the highest accuracy in comparison to all the other aforementioned algorithms such as LSTM, SVM, and the Multimodal NB. This in union with the Pegasus Tokenizer allows us to create a summary of all the reviews of the searched product.

References

[1] M. Ahmad, S. Aftab, M. S. Bashir, N. Hameed, I. Ali, and Z. Nawaz. Svm optimization for sentiment analysis. *International Journal of Advanced Computer Science and Applications*, 9(4), 2018. 2

[2] E. Camp. Sarcasm, pretense, and the semantics/pragmatics distinction. *Noûs*, 46(4):587–634, 2012. 5

[3] S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar, and M. Dey. A comparative study of support vector machine and naive bayes classifier for sentiment analysis on amazon product reviews. In *2020 International Conference on Contemporary Computing and Applications (IC3A)*, pages 217–220. IEEE, 2020. 2

[4] A. Hassan and A. Mahmood. Deep learning approach for sentiment analysis of short texts. pages 705–710, 2017. 2

[5] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao. Attention-emotion-enhanced convolutional lstm for sentiment analysis. *IEEE transactions on neural networks and learning systems*, 2021. 2

[6] M. Koroteev. Bert: A review of applications in natural language processing and understanding. 03 2021. 2, 3

[7] P. Rajpurkar, J. Zhang, K. Lopyrev, and P. Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016. 3

[8] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, and S. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, Nov. 2018. Association for Computational Linguistics. 2

[9] J. Zhang, Y. Zhao, M. Saleh, and P. Liu. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR, 2020. 4