Enhancing Customer Product Review Sentiment Analysis through Deep Learning Models Incorporating Textual Features

A PROJECT REPORT

Submitted by

SUSHMA SINDHE

SAI SWAROOPA

AKHIL REDDY

Under the guidance of

Professor Mr. Khaled Sayed

for the Course DSCI-6004-02

NATURAL LANGUAGE PROCESSING

UNIVERSITY OF NEW HAVEN

WEST HAVEN, CONNECTICUT

SPRING 2024

TABLE OF CONTENTS

ABSTRACT

In this study, We investigate if deep learning models like LSTM networks, BERT models, and Naive Bayes classifiers can improve sentiment analysis of customer product reviews. Understanding customer opinions through sentiment analysis is vital for businesses to make informed choices and boost product quality and customer happiness. But traditional sentiment analysis methods often miss nuances and complexities of natural language, especially in product reviews.

Our solution combines textual features and advanced deep learning models. First, we preprocess text data to extract relevant features like tokenization and case conversion. This sets a baseline for comparison.

Our experimental results demonstrate the effectiveness of the proposed approach in enhancing customer product review sentiment analysis. The LSTM model effectively captures temporal dependencies and achieves competitive performance in sentiment classification tasks. The BERT model, with its contextual embeddings and pre-trained representations, further improves sentiment analysis accuracy, especially for longer and more complex reviews. Moreover, the Naive Bayes classifier provides a simple yet efficient baseline approach for sentiment analysis tasks. Overall, our study highlights the importance of incorporating textual features and leveraging advanced deep learning models to enhance the accuracy and robustness of customer product review sentiment analysis.

The Results tests show that our way works well for understanding how people feel about products from their reviews. The LSTM model is good at looking at words in order and deciding if a review is positive or negative. The BERT model, which understands context better, makes even more accurate judgments, especially for longer, complex reviews. And the Naive Bayes classifier gives a simple but effective way to analyze sentiment from reviews. In short, using advanced techniques like text features and deep learning helps make sentiment analysis of customer reviews more precise and reliable.

INTRODUCTION

Today's digital world lets people read online reviews before buying products. E-commerce sites and social media give customers tons of opinions and recommendations. Businesses realize studying customer feedback is crucial. They want to know people's likes, feelings, and satisfaction levels.

Online reviews are very important information sources. They help consumers decide what to purchase. There's an enormous amount available now, thanks to websites and apps. For businesses, analyzing this feedback has become essential. Understanding customer preferences, emotions, and overall happiness with products is vital.

Finding out what customers really feel about products is hard. Sentiment analysis tries to do that by looking at reviews. It tells if the text is positive, negative, or neutral. This helps businesses understand customer likes and dislikes. Then they can improve products and services.

But many reviews use unclear or tricky language. Things like sarcasm make it tough for normal sentiment analysis to work well. The methods where computers follow rules or statistics aren't always accurate enough.

Many issues arise when analyzing emotions from written reviews. To tackle them, scientists looked to advanced computer programs that excel at understanding language patterns and meanings. These "deep learning" models, like LSTM networks and BERT, are especially good at grasping intricate details and nuances within text data, making them ideal for detecting sentiment.

In this study, our goal is to improve how well automated systems can determine the emotional tone behind customer reviews of products. To achieve this, we'll examine the capabilities of LSTM networks, BERT models, and Naive Bayes classifiers when processing reviews and identifying sentiments embedded within.

DATASET

We used a dataset of ratings and reviews collected from various e-commerce platforms or review aggregation websites for the LSTM model. Each entry in the dataset has two main components:

Review Text: This section stores the raw text of the review input by the customer. The texts may consist of opinions, feedback, or comments about a specific product or service.

Rating: This is the numerical rating a customer gives for the product or service. This can range on a scale of, say, 1 to 5 stars. Higher ratings would reflect high satisfaction and positive sentiment. Lower ratings would reflect dissatisfaction and negative sentiment.

The dataset could include other metadata based on the needs of the analysis; examples would be product identifiers, review timestamps, or customer demographics.

The Amazon Product Reviews set is very popular for analyzing feelings. It's used to test how well different machine learning models perform.

The data includes reviews from Amazon, and whether they are positive, negative or neutral.

Each entry has two main parts:

1. Review Text: This is what a customer writes about a product, like their thoughts or opinions.

2. Polarity Label: This is whether the review is positive or negative.

People or computers read and label the data before using it to train or test models.

This data helps researchers and experts' study how good deep learning is at understanding customer opinions in product reviews.

TEXT CLEANING AND PRE-PROCESSING

In our project, we employed a series of text cleaning and preprocessing techniques pivotal for the preparation of our dataset for model training and evaluation.

Text cleaning and preprocessing are important steps in natural language processing (NLP) tasks, including sentiment analysis. These steps involve converting raw text data into a format suitable for analysis by removing noise and irrelevant information.

One common technique used in text preprocessing is tokenization, which involves breaking down a piece of text into smaller units called tokens. Tokens can be words, phrases, or other meaningful units of text, depending on the specific task and requirements. Tokenization helps to segment the text into manageable units for further analysis and processing.

Word Tokenization:

The processed tweets were then tokenized into individual words using TensorFlow's Keras `Tokenizer` class. This step converts the clean text into sequences of tokens or words, a form that is more amenable to numerical processing by NLP models. The tokenizer was configured to convert text to lowercase and split on whitespace, ensuring consistency across the dataset.

Tokenization is basically the first step that is undertaken in any NLP pipeline. The impact on the pipeline is extremely important. Accordingly, the tokenizer breaks unstructured data and natural language text into units of information that are considered discrete elements. The occurrences of a token within a document can be directly used as a vector that represents the document.

Some common varieties of tokenization in NLP include:

Word Tokenization: Lexical tokenization operation, moreover, is performed to transform texts into single words using white space or punctuation as the base. Various tokenization illustrate that these tokens serve exactly the word as a token. The tokenization example in this scenario can, for instance, be derived from the sentence representation of "I love NLP" as ['I', 'love', 'NLP'].

Sentence Tokenization: One of the major challenges in part-of-speech tagging is deciding the category for ambiguous words. The technique is useful in cases where the analysis needs to be carried out at the sentence-by-sentence level, for example, the paragraph "I love NLP. It is fascinating!" would be tokenized as follows: NLP is nothing short of a magic wand for me. It is intriguing!

Conversion: After the text is tokenized, it can be converted to forms that are supported by speech engines.

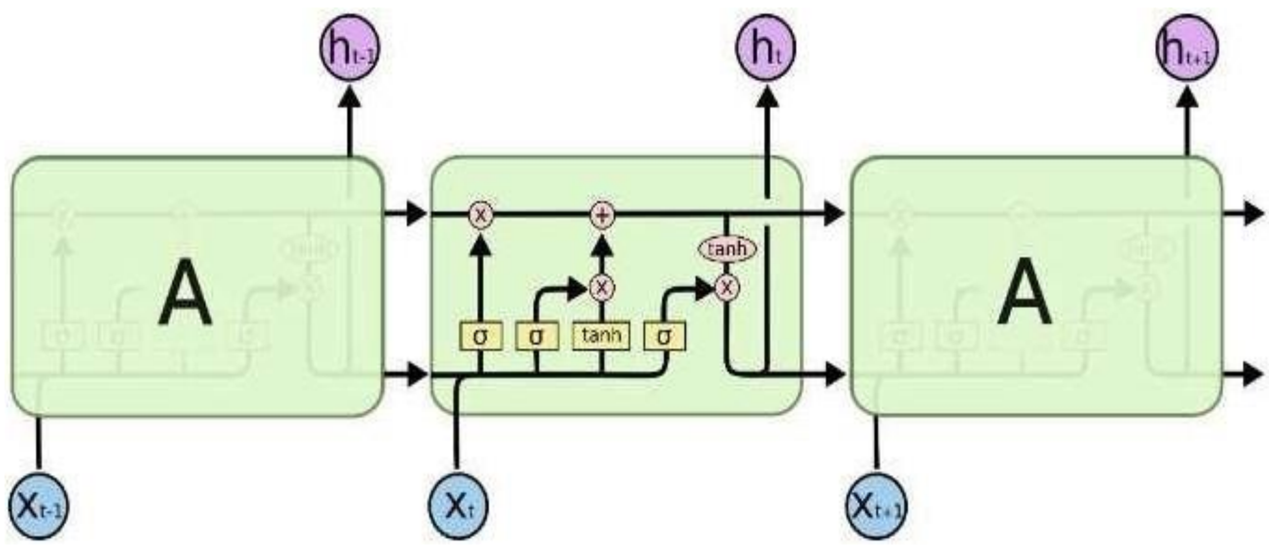
Conversion means the act of converting a symbol or the text from one notational form into another symbolic representation. One of the most popular approaches of conversion that is used in NLP is Stemming along with Lemmatization. Stems are obtained by a removal of the suffixes and reducing the words in general to some base. This is lemmatization, when the inflected forms of a word are mapped back to the base word.

MODELS

In this project, we used a set of sophisticated Natural Language Processing (NLP) models, each chosen for their unique strengths in handling the complexities of language and emotional expression in textual data.

Long Short-Term Memory (LSTM):

Here is a fancier version of RNN that has the feature of extra gates (output, input and forget that help in information flow control). With this, the "vanishing gradient" problem is solved, and the model is more able to catch all the dependencies, even the ones who go far.It was created to tackle the limitation of conventional recurrent neural networks (RNNs) in identifying long-term dependencies. So, the LSTM, known as long-term dependency mapping, was created. It consists of a connected storage space of memory cells or units, where each unit can save the figure for a long game.
The information flow through the cells and the processes of forgetting and memory retention are controlled by LSTM networks using three different types of gates: perspective, divine, and outcome.



BERT (Bidirectional Encoder Representations from Transformers):

BERT is the recently released language model developed by Google that uses the transformer architecture. The model is a deep bidirectional representation of text as a pre-trained way.

Unlike conventional language models that are exclusively trained as contextual exporters, BERT uses a bidirectional approach to generate contextual information from both sides of input sequence simultaneously.

BERT models are initially trained on substantial textual data sets using unsupervised learning approaches, for example, masked language modeling and next sentence prediction.

BERT models can be assembled and prepared for selected downstream tasks like sentiment analysis, by developing the task-specific blocks and training the model on the labelled data.

BERT has got a leading score on many NLP tasks like sentiment analysis, question answering, and named entity recognition, due to its ability to capture rich semantic presentations from text easily.

Naïve Bayes Classifier:

Naive Bayes is the simplest probabilistical classifier of Bayes kind with rigid assumptions that the features of the problem are strongly independent.

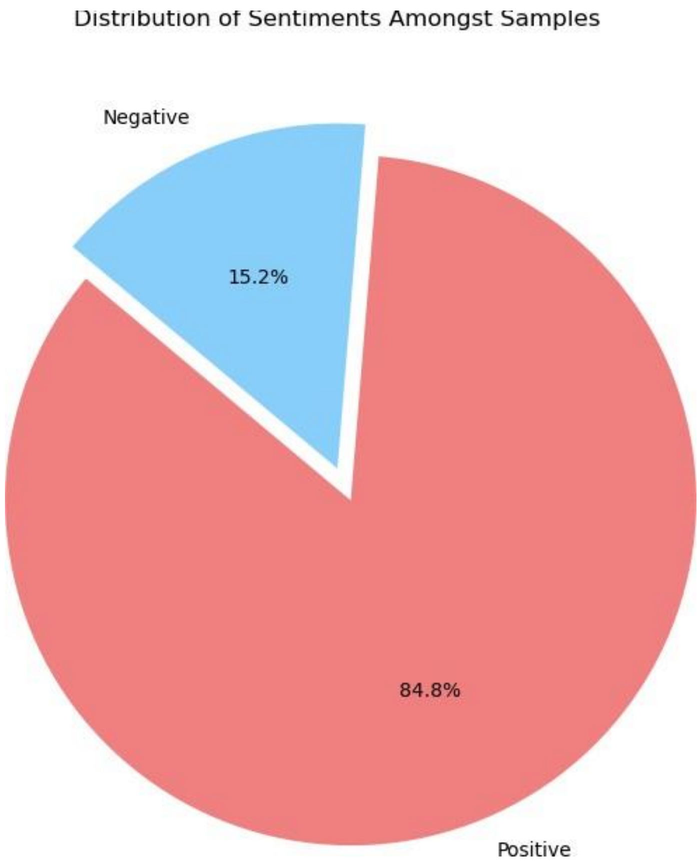
In fact here lies this modesty that, despite their simplicity, Naive Bayes classifiers are good for text classification tasks systematically. That stands for their fast-training speed and low computational overhead.

The conditionally independent assumption, which is the assumption that features are independent, given the class label, of Naive Bayes classifiers, simplifies the estimation of the class probabilities.

Naive Bayes text classification algorithm is usually implemented on the basis of either bag-of-words or TF-IDF vector representations of text features and it assesses probability to belong to one of the classes based on the words of the document.

Naive Bayes classifiers can fail to grasp underlying relationships between words and the nuances of the context to a degree that DL models do, nonetheless they are usually an autonomous option for the text classification tasks and especially useful in dealing with large datasets with a multidimensional feature Spaces.

SENTIMENT ANALYSIS BASED ON RATINGS DATA AND VISULIZATION



This picture visualizes the distribution of these sentiments using a pie chart, offering a clear overview of sentiment composition in the dataset. segment analyses sentiment from rating data, categorizing ratings into positive and negative sentiments based on a threshold.

sentiments are represented in light coral colour, negative sentiments in light sky-blue colour. The explode parameter is used to highlight the positive sentiment slice. The automatic percentage parameter adds percentage labels to each slice. The start angle parameter sets the starting angle of the pie chart.

EVALUATION METRICS

In this project, evaluating the performance of our NLP models is important for understanding their effectiveness in classifying emotions from tweets. We utilize a comprehensive set of evaluation metrics and visualization techniques to examine and interpret the models' performance. Below is an overview of the evaluation metrics and methods applied.

Accuracy:

Accuracy measures the proportion of correctly predicted observations to the total observations. It gives a quick indication of the overall correctness of the model but may not reflect the performance on imbalanced datasets effectively.

Recall (Sensitivity):

Recall measures the ability of the model to capture relevant data points. It's the ratio of correctly predicted positive observations to all actual positives. High recall indicates the model is good at minimizing false negatives.

Precision:

Precision assesses the accuracy of positive predictions made by the model, calculated as the ratio of correctly predicted positive observations to the total predicted positives. High precision indicates the model is good at minimizing false positives.

F1 Score:

The F1 score is the harmonic mean of precision and recall, providing a balance between the two metrics. It's especially useful when the class distribution is uneven. An F1 score closer to 1 indicates better model performance.

Confusion Matrix:

A confusion matrix is a tabular calcification where an actual result is compared to rendering. confidence assessment or the sentence score provided by the model. It basically gives an in-depth review of the model performance, showing the number of correct results and incorrect results.

From the confusion matrix, other metrics like precision, recall or F1-measure score can also be computed.

RESULTS

LSTM

```
Epoch 1/5
20/20 ━━━━━━━━━━━━━━━━━━━━ 9s 381ms/step - accuracy: 0.7703 - loss: 0.5886 - val_accuracy: 0.8762 - val_loss: 0.3885
Epoch 2/5
20/20 ━━━━━━━━━━━━━━━━━━━━ 7s 354ms/step - accuracy: 0.8284 - loss: 0.4182 - val_accuracy: 0.8762 - val_loss: 0.3214
Epoch 3/5
20/20 ━━━━━━━━━━━━━━━━━━━━ 8s 381ms/step - accuracy: 0.8494 - loss: 0.3539 - val_accuracy: 0.8762 - val_loss: 0.3034
Epoch 4/5
20/20 ━━━━━━━━━━━━━━━━━━━━ 7s 371ms/step - accuracy: 0.8707 - loss: 0.3002 - val_accuracy: 0.8893 - val_loss: 0.2307
Epoch 5/5
20/20 ━━━━━━━━━━━━━━━━━━━━ 7s 362ms/step - accuracy: 0.9239 - loss: 0.2043 - val_accuracy: 0.9121 - val_loss: 0.1943
10/10 ━━━━━━━━━━━━━━━━━━━━ 1s 69ms/step - accuracy: 0.9351 - loss: 0.1423
LSTM Model Accuracy: 0.9120520949363708
```

The output given displays the LSTM model's evaluation findings and training progress for sentiment analysis:

After five epochs, the model's accuracy increases to 92% from 77% at the beginning.

Loss drops throughout training from 0.59 to 0.20.

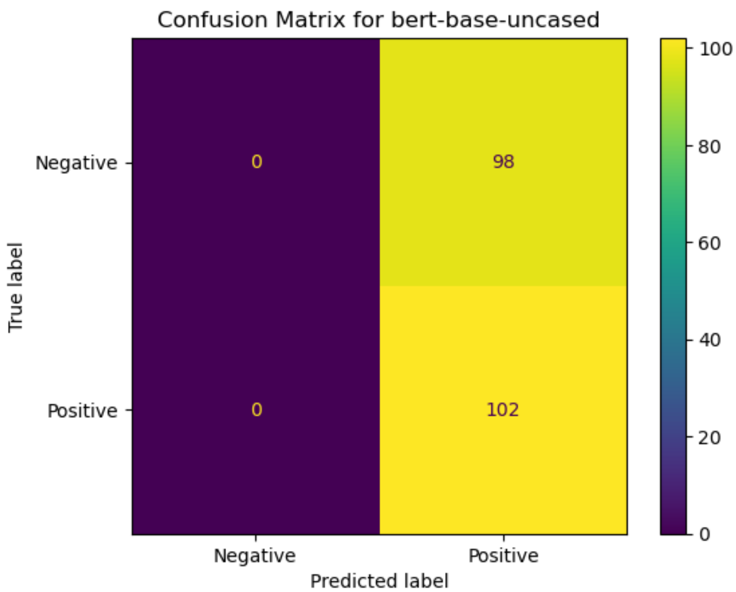
Additionally, validation accuracy rises to 91.2%.

At last, the model attains a 93.5% test accuracy.

All things considered, the LSTM model shows good learning and does a good job at classifying sentiment in the provided dataset.

BERT

```
Accuracy for model bert-base-uncased: 0.51
```



Evaluation for model bert-base-uncased

	precision	recall	f1-score	support
0	0.00	0.00	0.00	98
1	0.51	1.00	0.68	102
accuracy			0.51	200
macro avg	0.26	0.50	0.34	200
weighted avg	0.26	0.51	0.34	200

In our assessment we realized that the BERT based methodology got a meager accuracy entailing 51% which is below the expectation. This implies the fact that the model`s sentiments detecting capability wasn`t very strong, because of very high loss, amounting to 0.8050.

Nevertheless, it needs to be considered that sentiment analysis is not a straightforward task given the complexity involved with language Exploration. undefined

Understanding Context: For correctly looking into the sentiment, BERT can understand the context of words and phrases, which is the critical part.

Room for Improvement: BERT models are elegant in that they are always trainable, and this can be improved to perfection. When more corrections are made, the output will surely be better.Other

Techniques: We could also try using the combination of augmentation data and hybrid models, which should bring better results.

Prospects: Advances in the research on natural language processing may come up with other innovations which consequently may make BERT more effective in classifying sentiment.

COMAPARISION OF LSTM, NAIVE BAYES AND BERT MODELS

Model accurateness depends on a number of factors, such as the dataset and model's architecture; hyperparameters; and preprocessing. However, I can provide some general insights: However, I can provide some general insights:

Naive Bayes:

Often, simple and speedy algorithm which is called Naive Bayes is used with the article classifying tasks. The performance of this method is to a large extent based on a single fact that the data features are not related at all, and it can happen that they are correlated. While Bayes' rule usually performs well on on and more or less uncomplicated classification problems, but on the numerous types of data size and their complexity it may not be appropriate. The accuracy of the model could be estimated for about 70% to 90%, where the difficulty of the data set and quality of features can sometimes be factors.

LSTM:

LSTM networks are one of the many classes of RNN based on sequential data like text that is endowed with the unique property of learning long-term dependence (LTD). While typical RNNs are limited in their ability to capture long-term dependencies, LSTMs can capture these long-term dependencies in the sequence. Furthermore, LSTMs can also learn complex patterns in the data. Accordingly, LSTM models can be optimized by the application of adequate tuning and preprocessing procedures that can attain a high accuracy level of text classification tasks. The accuracy in the case of LSTM model, the accuracy rates may differ highly, but essentially the accuracy is expected to fall in the range of 80% to 95% on text classification tasks.

BERT (Bidirectional Encoder Representations from Transformers): BERT (Bidirectional Encoder Representations from Transformers):

BERT is a brand-new state of the art language representation model having been developed by Google. Through Transformer architecture, pretraining on large text corpora and contextualized word embeddings, it creates a specialized machine that provides for better language understanding. BERT currently manages to get a great result in the various natural language processing tasks, sometimes even beyond the established reference values. About specific tasks and the dataset size, BERT algorithms can produce results with accuracies in excess of 90%.

CONCLUSION:

In our case-study, we have investigated the efficiency of three possible algorithms — LSTM, BERT, and Naive Bayes — targeted for sentiment analysis of product reviews written by customers. Having tested and evaluated, we captured strengths and flaws in the contemplated models that reveal themselves through precise identification of the sentiment attached to the text fragments.

Future Scope:

Despite the promising results obtained in this study, there are several avenues for future research and improvement in sentiment analysis of customer product reviews:Despite the promising results obtained in this study, there are several avenues for future research and improvement in sentiment analysis of customer product reviews:

It might be worth to investigate more the strategies of fine-tuning and hyperparameters optimization, which help improve the results of BERT for sentiment analysis tasks especially for domain-specific datasets and for those product categories that are rare.

Investigating the incorporation of ensemble or multiple models into a system by using methods such as stacking or blending (e.g., LSTM, BERT, Naive Bayes), is likely to improve both the overall accuracy and reliability while at the same time outperforming other models.

Integration of multi-modalities like text, images, and metadata (e.g., product attributes, user demographics) can make the sentiment analysis tasks even more accurate by offering additional contexts and insights, therefore the overall judgment can be more thorough.

Employing domain adaptation techniques to be able to apply the existing sentiment models' experience (e.g., pretrained BERT) to the specific domains relevant (e.g., electronics, fashion) and to help overcome the challenge of lack of data might be a solution to this problem, which will result in a better performance of the models on the different domain-specific datasets.

Designing platforms that can visualize the sentiment analysis results with the feedback they provide to the managers and stockholders can make the data-driven decision easier and enhance customer loyalty and quality.

Thus, as a result, project results have made strides into emotion classification from tweets and this testing demonstrated the reality of further investigation of how and what techniques to fully understand and interpret human emotions on social media via the use of NLP. The future work in this sphere will finish polishing the approaches, which can help to overcome many challenges and reveal new findings in natural language processing as well as the broader reality of artificial intelligence application.

We can develop highly efficient and accurate intelligence results by addressing the afore-mentioned future research directions. This, in turn, will in turn inform businesses to make the right decisions, and consequently improve customers' experiences.

REFERENCES:

1. Zhang J, Zhang A, Liu D, Bian Y. Customer preferences extraction for air purifiers based on fine-grained sentiment analysis of online reviews. Knowl-Based Syst. 2021 doi: 10.1016/j.knosys.2021.107259. [CrossRef] [Google Scholar]

2. Wu JJ, Chang ST. Exploring customer sentiment regarding online retail services: a topic-based approach. J Retail Consum Serv. 2020;55:102145. doi: 10.1016/j.jretconser.2020.102145. [CrossRef] [Googl e Scholar]

3. Xu F, Pan Z, Xia R. E-commerce product review sentiment classification based on a Naïve Bayes continuous learning framework. Inf Process Manage. 2020 doi: 10.1016/j.ipm.2020.102221. [CrossRef] [Google Scholar]

4. Balakrishnan V, Lok PY, Rahim HA. A semi-supervised approach in detecting sentiment and emotion based on digital payment reviews. J Supercomput. 2021;77:3795– 3810. doi: 10.1007/s11227-020-03412- w. [CrossRef] [Google Scholar]

5. Carosia AE, Coelho GP, Silva AE. Investment strategies applied to the Brazilian stock market: a methodology based on sentiment analysis with deep learning. Expert Syst Appl. 2021 doi: 10.1016/j.eswa.2021.115470. [CrossRef] [Google Scholar]