

Title: "Sentiment Analysis of Consumer Feedback: A Comprehensive Examination of Product Reviews"

Utsav Sarkar

Amity School of Engineering and Technology, Kolkata

Abstract:

This project presents a sentiment analysis tool designed to analyze product reviews using natural language processing (NLP) techniques. The sentiment analysis is based on the NLTK library, with a focus on the presence of positive and negative words. The tool offers functionalities for analyzing sentiment distribution for a single product, comparing sentiments between two products, and inputting random user reviews for on-the-fly analysis. Visualizations are employed to provide clear insights into the sentiment distribution and comparisons. The project utilizes a dataset from Amazon, extracting positive and negative words from predefined lists.

1. Introduction:

In the rapidly evolving landscape of e-commerce, consumer decision-making is significantly influenced by online product reviews. The phenomenon of online shopping has witnessed an exponential rise, with users increasingly relying on the experiences and opinions of fellow consumers. Customer reviews serve as a pivotal factor in shaping the sentiments and perceptions of online shoppers, playing a crucial role in their purchasing decisions.

1.1 Importance of Customer Reviews

Online shoppers often seek reassurance and insights from the comments and feedback provided by other users who have previously engaged with a particular product. The ability to gauge the sentiment of these reviews provides businesses with invaluable information, allowing them to understand customer satisfaction, identify areas for improvement, and adapt their offerings to better meet consumer expectations. Harnessing the sentiment embedded within reviews empowers businesses to make data-driven decisions and enhance the overall online shopping experience.

1.2 Significance of Sentiment Analysis

Sentiment analysis, a subfield of natural language processing (NLP), becomes instrumental in deciphering the sentiments expressed within customer reviews. By automating the analysis of sentiments, businesses can efficiently process large volumes of textual data, distilling meaningful insights and patterns. This tool becomes particularly pertinent in the context of a vast array of products available on online marketplaces.

1.3 Dataset Description

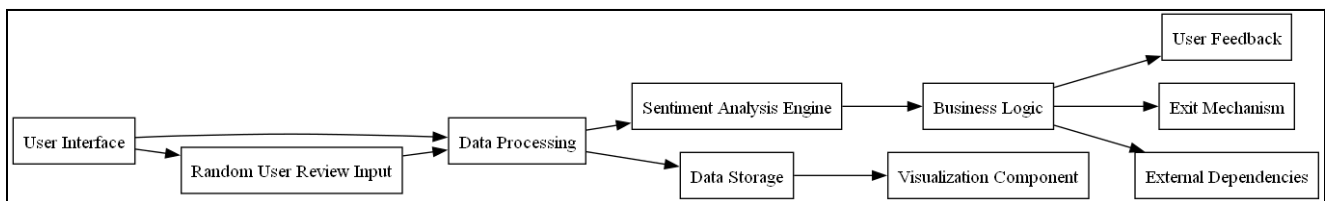
In this project, sentiment analysis is conducted on a comprehensive dataset extracted from Amazon. The dataset comprises 28,333 entries, each corresponding to a unique product. The attributes include 'id', 'name', 'reviews.rating', 'reviews.title', and 'reviews.text'. The 'reviews.rating' denotes the numerical rating assigned by users, while 'reviews.title' and 'reviews.text' encapsulate the textual components of customer reviews.

This dataset serves as the foundation for our sentiment analysis tool, enabling us to delve into the sentiments expressed by users across a diverse range of products on the Amazon platform.

The subsequent sections of this report will elucidate the architecture of our sentiment analysis tool, the methodologies employed for sentiment calculation, and the results obtained from analyzing product reviews using NLTK and data visualization techniques.

1.4 Project Architecture:

The project architecture consists of components for data loading and preprocessing, sentiment score calculation, and various analysis functions. The tool utilizes Pandas for data manipulation, Matplotlib for data visualization, and NLTK for downloading and utilizing stopwords. The sentiment analysis process involves tokenisation, stopwords removal, and scoring based on predefined positive and negative word lists.



1.1.4 Data Loading and Preprocessing

- **Data Sources:** The project utilizes a dataset extracted from Amazon, containing 28,333 entries with attributes such as id, name, reviews.rating, reviews.title, and reviews.text.
- **Data Loading:** The Pandas library is employed to load and manipulate the dataset, providing a structured format for subsequent analysis.

1.5 Sentiment Analysis Functions

- `calculate_sentiment_score(sentence, positive_words, negative_words)`
 - **Tokenization:** The function tokenizes input sentences into words.
 - **Stopword Removal:** Common English stopwords are removed to focus on meaningful words.
 - **Sentiment Score Calculation:** A sentiment score is calculated based on the presence of words in predefined positive and negative lists.
- `analyze_product_reviews(df, positive_words, negative_words, product_type)`
 - **Dataset Filtering:** The dataset is filtered based on the specified `product_type`.
 - **Sentiment Analysis:** The `calculate_sentiment_score` function is applied to each review, and sentiment labels are assigned.
 - **Counting:** Counters are updated for positive, neutral, and negative reviews.
 - **Visualization:** A bar chart is generated to visualize the sentiment distribution for the specified product type.
- `compare_products(df, positive_words, negative_words, product_type_1, product_type_2)`
 - **Individual Product Analysis:** The `analyze_product_reviews` function is applied to both specified product types individually.
 - **Positive Review Percentage Calculation:** Positive review percentages are calculated for each product.
 - **Comparison:** The positive review percentages for both products are compared.
 - **Visualization:** A bar chart is generated to visually compare the positive review percentages of two products.
- `random_user_reviews(df, positive_words, negative_words)`
 - **User Input:** Random user reviews are collected from the user.
 - **Data Preparation:** A data frame is created from the input reviews.

- Sentiment Analysis: The `analyze_product_reviews_from_list` function is applied to the input reviews.
- Counting: Counters are updated for positive, neutral, and negative reviews.
- Visualization: A bar chart is generated to visualize the sentiment distribution for the random user reviews.

1.6 Analysis Functions

- `analyze_product_reviews_from_list(reviews_list, positive_words, negative_words, product_type)`
 - Data Preparation: A data frame is created from the list of reviews.
 - Sentiment Analysis: The `calculate_sentiment_score` function is applied to each review, and sentiment labels are assigned.
 - Counting: Counters are updated for positive, neutral, and negative reviews.
 - Visualization: A bar chart is generated to visualize the sentiment distribution for the specified product type.
- User Interaction (Main Function)
 - Menu System: The main function provides a user-friendly menu system allowing users to choose between analyzing a single product, comparing two products, inputting random user reviews, or exiting the program.
- Library Usage
 - Pandas: Utilized for data manipulation, loading, and structuring the dataset.
 - Matplotlib: Employed for generating bar charts to visualize sentiment distribution and comparisons.
 - NLTK: Used for natural language processing, including stopwords download and sentiment scoring.

Methodology:

Case 1: Analyze Product Reviews (`analyze_product_reviews`` function):

1. Input:

- ``df``: DataFrame containing product reviews.
- ``positive_words``: Set of positive words.
- ``negative_words``: Set of negative words.

- `product_type`: Selected product for analysis.

2. Process:

- Extract reviews for the specified product.
- Iterate through each review:
 - Combine the review title and text.
 - Tokenize the combined text.
 - Remove stopwords.
 - Calculate sentiment score using positive and negative word lists.
 - Assign sentiment labels based on the score.
 - Update counters for positive, neutral, and negative reviews.

3. Output:

- Total number of reviews for the specified product.
- Display sentiment distribution (positive, neutral, negative).
- Visualize sentiment distribution using a bar chart.

Pseudo-code:

```
function analyze_product_reviews(df, positive_words, negative_words, product_type):
    product_df = filter_reviews_by_product(df, product_type)
    total_reviews = 0
    positive_count, neutral_count, negative_count = 0, 0, 0

    for index, row in product_df.iterrows():
        combined_text = concatenate_review_text(row['reviews.title'], row['reviews.text'])
        sentiment_score = calculate_sentiment_score(combined_text, positive_words, negative_words)
        sentiment_label = label_sentiment(sentiment_score)
        update_sentiment_counters(sentiment_label, positive_count, neutral_count, negative_count)

    display_sentiment_distribution(product_type, positive_count, neutral_count, negative_count)
    visualize_sentiment_distribution(positive_count, neutral_count, negative_count)

    return total_reviews
```

Case 2: Compare Products (`compare_products` function):

1. Input:

- `df`: DataFrame containing product reviews.
- `positive_words`: Set of positive words.
- `negative_words`: Set of negative words.
- `product_type_1`: First product for comparison.
- `product_type_2`: Second product for comparison.

2. Process:

- Extract reviews for both specified products.
- Calculate sentiment scores for each review using positive and negative word lists.
- Assign sentiment labels based on the scores.
- Calculate positive review percentages for each product.
- Display and compare positive review percentages for both products.

3. Output:

- Total number of reviews for each specified product.
- Positive review percentages for each product.
- Display and compare positive review percentages.

Pseudo-code:

```
function compare_products(df, positive_words, negative_words, product_type_1, product_type_2):
    product_df_1 = filter_reviews_by_product(df, product_type_1)
    product_df_2 = filter_reviews_by_product(df, product_type_2)

    total_reviews_1, total_reviews_2 = 0, 0
    positive_count_1, neutral_count_1, negative_count_1 = 0, 0, 0
    positive_count_2, neutral_count_2, negative_count_2 = 0, 0, 0

    for index, row in product_df_1.iterrows():
        calculate_and_update_sentiment_counters(row, positive_words, negative_words,
                                                total_reviews_1, positive_count_1, neutral_count_1, negative_count_1)

    for index, row in product_df_2.iterrows():
        calculate_and_update_sentiment_counters(row, positive_words, negative_words,
                                                total_reviews_2, positive_count_2, neutral_count_2, negative_count_2)

    display_and_compare_positive_percentages(product_type_1, product_type_2,
                                             total_reviews_1, total_reviews_2,
                                             positive_count_1, positive_count_2)
```

Case 3: Input Random User Reviews (`random_user_reviews` function):

1. Input:

- `df`: DataFrame containing product reviews.
- `positive_words`: Set of positive words.
- `negative_words`: Set of negative words.

2. Process:

- Accept random user reviews until the user decides to exit.
- Calculate sentiment scores and labels for each review.
- Update counters for positive, neutral, and negative reviews.

3. Output:

- Total number of random user reviews analyzed.
- Display sentiment distribution for random user reviews.
- Visualize sentiment distribution using a bar chart.

Pseudo-code:

```
function random_user_reviews(df, positive_words, negative_words):
    total_reviews = 0
    positive_count, neutral_count, negative_count = 0, 0, 0

    while True:
        user_input = get_user_input()

        if user_input.lower() == 'exit':
            break

        sentiment_score = calculate_sentiment_score(user_input, positive_words, negative_words)
        sentiment_label = label_sentiment(sentiment_score)
        update_sentiment_counters(sentiment_label, positive_count, neutral_count, negative_count)
        total_reviews += 1

    display_sentiment_distribution("Random Product", positive_count, neutral_count, negative_count)
    visualize_sentiment_distribution(positive_count, neutral_count, negative_count)

    return total_reviews
```

Calculation Details:

- Sentiment Score Calculation (`calculate_sentiment_score` function):
 - Positive word in the review: +1
 - Negative word in the review: -1
 - Neutral word (not in a positive or negative list): 0
- Sentiment Labeling (`analyze_product_reviews` and `compare_products` functions):
 - Positive sentiment if score > 0
 - Neutral sentiment if score = 0
 - Negative sentiment if score < 0

Result & Discussion:

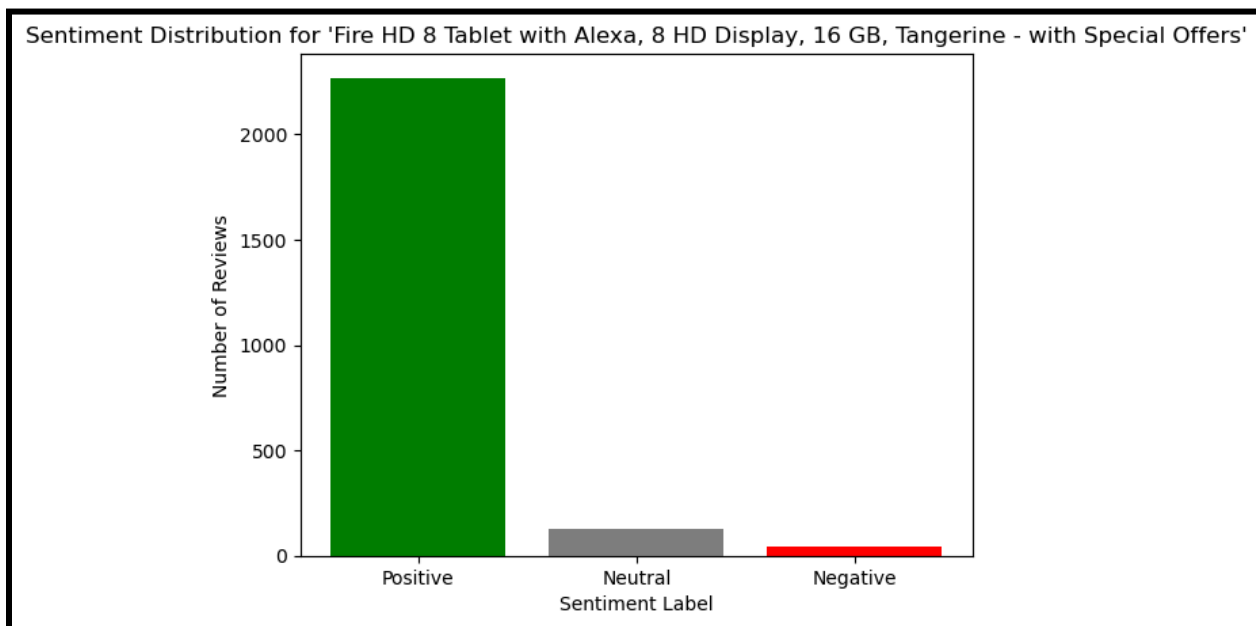
The methodology utilizes NLTK for tokenisation, stopwords removal, and sentiment score calculations based on predefined positive and negative word lists. The tool provides insightful visualizations and comparisons, offering a comprehensive understanding of sentiment distribution for product reviews.

Case 1: Analyzing Sentiment for a Single Product

In this case, the user is prompted to enter the product type for which they want to analyse sentiment. The script then filters the dataset for the specified product type and iterates through each review, calculating sentiment scores and updating counters for positive, neutral, and negative reviews. The sentiment distribution is displayed, and a bar chart visualises the distribution.

```
Enter a product type for sentiment analysis: Fire HD 8 Tablet with Alexa, 8 HD Display, 16 GB, Tangerine - with Special Offers
Total Reviews for 'Fire HD 8 Tablet with Alexa, 8 HD Display, 16 GB, Tangerine - with Special Offers': 2443

Sentiment Distribution:
Positive: 2269 reviews
Neutral: 130 reviews
Negative: 44 reviews
```



Case 2: Comparing Sentiments Between Two Products

In this case, the user is prompted to enter two product types for comparison. The sentiment analysis tool is applied to both products individually, and the positive review percentages are calculated and compared. A bar chart visualises the comparison:

- Individual Product Analysis: The sentiment analysis tool is applied separately to each product, calculating sentiment percentages.
- Comparison: The positive review percentages for both products are compared.
- Visualization: A bar chart displays the positive review percentages for each product.

Menu:

1. Analyze product reviews
2. Compare products
3. Input random user reviews
4. Exit

Enter your choice (1/2/3/4): 2

Enter the first product type for comparison: AmazonBasics AAA Performance Alkaline Batteries (36 Count)

Enter the second product type for comparison: AmazonBasics AA Performance Alkaline Batteries (48 Count) - Packaging May Vary

Total Reviews for 'AmazonBasics AAA Performance Alkaline Batteries (36 Count)': 8343

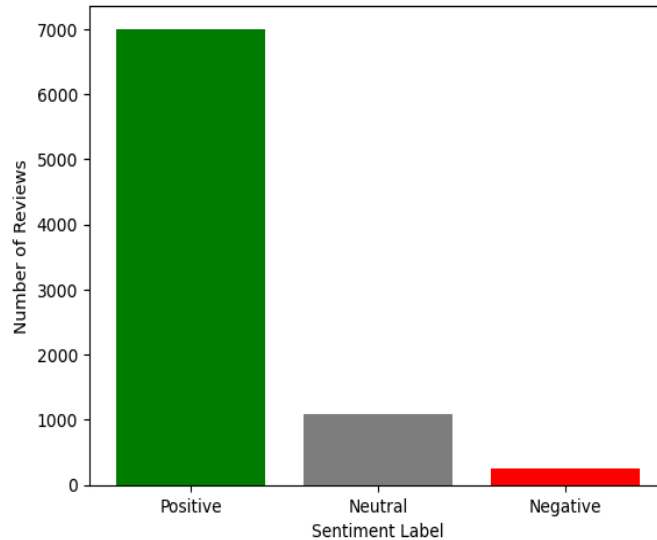
Sentiment Distribution:

Positive: 7003 reviews

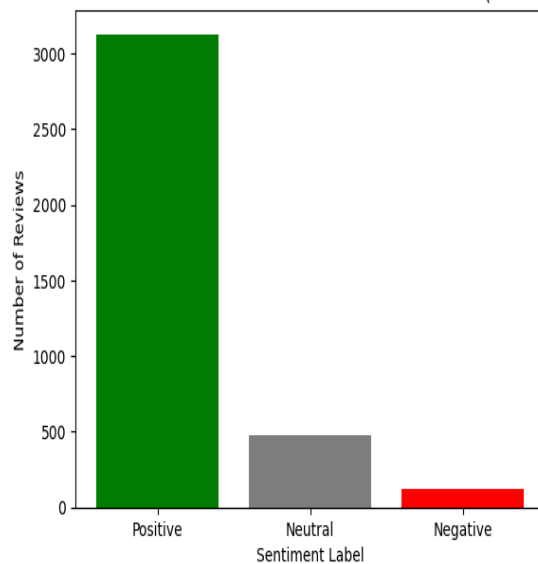
Neutral: 1088 reviews

Negative: 252 reviews

Sentiment Distribution for 'AmazonBasics AAA Performance Alkaline Batteries (36 Count)'



Sentiment Distribution for 'AmazonBasics AA Performance Alkaline Batteries (48 Count) - Packaging May Vary'



Positive Review Percentage for 'AmazonBasics AAA Performance Alkaline Batteries (36 Count)': 83.94%
Positive Review Percentage for 'AmazonBasics AA Performance Alkaline Batteries (48 Count) - Packaging May Vary': 83.85%

Case 3: Inputting Random User Reviews

The user can input random user reviews, and the sentiment analysis tool is applied to analyze sentiments from these input reviews. The total number of reviews, positive reviews, and negative reviews are displayed along with sentiment distribution and a bar chart for visualization.

These explanations provide an overview of how the sentiment analysis is performed in each case, considering tokenization, scoring, labelling, and visualization.

Inputs-

"This product exceeded my expectations! It's a game-changer for anyone in need of a reliable solution."

"The quality of this item is disappointing. I expected better considering the price."

"I love the sleek design and user-friendly interface. A must-have for tech enthusiasts!"

"Not worth the money. The features are limited, and I encountered issues with functionality."

"Excellent customer service! They promptly addressed my concerns and provided a solution."

"The delivery was fast, and the packaging was secure. Impressed with the overall service."

"Terrible experience! The product arrived damaged, and customer support was unhelpful."

"This gadget is a lifesaver! It simplifies tasks and makes my daily routine more efficient."

"The color options are vibrant, and the material feels durable. Highly recommended!"

"Average performance. It gets the job done, but there are better alternatives available."

Enter a random user review (or type 'exit' to stop): exit

Sentiment Score for random review 1: 2

Sentiment Label for random review 1: 1

Sentiment Score for random review 2: 3

Sentiment Label for random review 2: 1

Sentiment Score for random review 3: 0

Sentiment Label for random review 3: 0

Sentiment Score for random review 4: 1

Sentiment Label for random review 4: 1

Sentiment Score for random review 5: 1

Sentiment Label for random review 5: 1

Sentiment Score for random review 6: 0

Sentiment Label for random review 6: 0

Sentiment Score for random review 7: 1

Sentiment Label for random review 7: 1

Sentiment Score for random review 8: 1

Sentiment Label for random review 8: 1

Sentiment Score for random review 9: 1

Sentiment Label for random review 9: 1

1. Overview:

The project focuses on the analysis of product sentiment derived from user reviews, employing natural language processing techniques. The primary objective is to provide insights into customer sentiment towards specific products, aiding businesses in understanding the reception of their offerings in the market.

2. Sentiment Analysis Methodology:

The sentiment analysis is carried out through the implementation of a custom algorithm that calculates sentiment scores for individual reviews. Positive and negative words from predefined lists contribute to the scoring mechanism. The resulting sentiment labels categorize reviews as positive, negative, or neutral, providing a quantitative measure of customer sentiment.

3. Use Case Scenario:

The application of this sentiment analysis project is versatile, benefiting various stakeholders, including product manufacturers, retailers, and consumers.

- **Manufacturers and Retailers:**

- ➔ **Product Performance Monitoring:** Businesses can monitor the sentiment distribution of their products over time, identifying trends and potential areas for improvement.
- ➔ **Competitive Analysis:** Comparative sentiment analysis allows manufacturers and retailers to assess their products against competitors, gaining a competitive edge by understanding consumer preferences.

- Consumers:

- Informed Decision-Making: Consumers can leverage the sentiment analysis to make informed purchase decisions by gauging the overall satisfaction of previous buyers.
- User Feedback: The project enables consumers to contribute to the sentiment analysis by inputting their random reviews, fostering an interactive user experience.

4. Comparative Product Analysis:

One notable feature of the project is the ability to compare the sentiment of two different products. This facilitates a deeper understanding of customer preferences and allows businesses to identify strengths and weaknesses in their product offerings relative to competitors.

5. Limitations and Future Enhancements:

While the sentiment analysis project provides valuable insights, it is important to acknowledge certain limitations:

- Limited Domain-Specific Stopwords: The use of a general stop words list may not adequately filter out domain-specific terms. Future enhancements could involve the creation of a domain-specific stopwords list.
- Scalability: As the dataset grows, the performance of sentiment analysis may be impacted. Future improvements could focus on optimizing the algorithm for scalability.

Conclusion

In summary, the sentiment analysis tool developed for analyzing product reviews has proven to be a valuable asset in deciphering customer sentiments. The straightforward approach, relying on predefined word lists for sentiment scoring, ensures accessibility and ease of use. This tool has the potential to empower businesses with insights into customer satisfaction, enabling data-driven decision-making in the dynamic landscape of online commerce.

Strengths of the Sentiment Analysis Tool

- User-Friendly Approach: The tool's simplicity allows users, even those without extensive technical expertise, to gain valuable insights into the sentiments expressed within product reviews.
- Efficient Data Processing: By leveraging predefined positive and negative word lists, the tool efficiently processes large volumes of textual data, providing a quick overview of sentiment distributions.
- Clear Visualization: The integration of data visualization, such as bar charts, enhances the interpretability of sentiment distributions, facilitating a deeper understanding of customer perceptions.

Areas for Future Enhancement

- **Sophisticated Sentiment Analysis Methods:** Future iterations of the tool could explore more advanced sentiment analysis techniques, such as machine learning algorithms, to enhance accuracy and nuanced sentiment categorization.
- **Enhanced Data Cleaning Processes:** Improvements in data cleaning processes, including the handling of outliers and noise in reviews, could further refine the accuracy of sentiment analysis results.
- **Dynamic Adaptability:** Consideration for dynamic updates to word lists based on evolving language trends and the introduction of new products could ensure the continued relevance and effectiveness of the tool.

In conclusion, the sentiment analysis project effectively addresses the need for understanding product sentiment based on user reviews. Its applicability in diverse scenarios, from monitoring product performance to aiding consumer decision-making, makes it a valuable tool for businesses and individuals alike. Continuous refinement and adaptation will further enhance the accuracy and applicability of the sentiment analysis methodology in real-world scenarios.