|  |  |
| --- | --- |
| Internship Project Title | RIO-125: Automate Sentiment Analysis of Textual Comments and Feedbacks |
| Name of the Company | TCS iON |
| Name of the Industry Mentor | Mr. Debashis Roy |
| Name of the Institute | Vishwakarma University, Pune |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Start Date | End Date | Total Effort (hrs.) | Project Environment | Tools used |
| 19/06/2024 | 18/07/2024 | 128 Hrs. | Python | Jupyter Notebook -Pandas, NLTK (Natural Language Toolkit), TextBlob, Scikit-learn, Matplotlb, Seaborn, GridSearchCV, Piprline, CountVectorizer, LSTM, Dense, StandardScaler, Tokenizer, Sequential Embedding, etc. |

# TABLE OF CONTENT

* Acknowledgements
* Objective
* Introduction / Description of Internship
* Internship Activities
* Approach / Methodology
* Assumptions
* Exceptions / Exclusions
* Charts, Table, Diagrams
* Algorithms
* Challenges & Opportunities
* Risk Vs Reward
* Reflections on the Internship
* Recommendations
* Outcome / Conclusion
* Enhancement Scope
* Link to code and executable file
* Research questions and responses

## ACKNOWLEDGEMENT

I am sincerely appreciative of the unwavering support and guidance provided to me throughout my project, RIO-125: Automate Sentiment Analysis of Textual Comments and Feedback. I wish to convey my heartfelt gratitude to my industry mentor, Mr. Debashis Roy from TCS-iON, and my academic mentor, Prof. Shriprada Chaturbhuj from Vishwakarma University. Their continual encouragement was crucial in my journey.

Furthermore, I extend my genuine thanks to TCS-iON and Vishwakarma University for offering me this invaluable opportunity, which has deepened my understanding of the industry landscape. I would like to emphasize that I completed the project independently, without any external help.

## OBJECTIVE

To create sophisticated deep learning models aimed at effectively identifying different types of sentiments expressed within English sentences or extensive paragraphs, with the ultimate objective of accurately determining the overall sentiment communicated by the entire text.

## INTRODUCTION/DESCRIPTION OF THE INTERNSHIP

This internship offers an exciting exploration into teaching computers to understand emotions expressed in written text. We will investigate deep learning techniques, enabling computers to determine whether a piece of writing conveys happiness, sadness, or any emotion in between. Our goal is to equip our computer counterparts with the ability to accurately interpret the overall emotional tone of any text, be it a brief message or an extensive essay. It's like giving our computer friends the skill to read between the lines and grasp the nuances of human emotions expressed through words. Through this project, we are delving into the intriguing world of emotional understanding in written language, aiming to enhance our computer companions' ability to recognize the wide range of sentiments people convey through their writing.

## 

## INTERNSHIP ACTIVITIES

The internship activities include the following:

1. Research and study sentiment analysis and deep learning algorithms.
2. Collect diverse textual data, preprocess it, and prepare it for training.
3. Experiment with various deep learning architectures for sentiment analysis.
4. Train models, evaluate their performance, and iterate for improvement.
5. Explore hyperparameter tuning techniques for model optimization.
6. Validate models generalization ability with unseen data.
7. Document the process and prepare reports summarizing findings.
8. Engage in discussions, workshops, and team meetings for collaborative learning.

## 

## 

## 

## 

## 

## 

## 

## APPROACH/METHODOLOGY

1. **Text Preprocessing:**
   1. **Tokenization:** Splitting the text into individual words or tokens.
   2. **Part-of-Speech (POS) Tagging:** Assigning grammatical roles (such as noun, verb, or adjective) to each token.
   3. **Lemmatization:** Converting words to their root or base form.
   4. **Removing Punctuation and Stopwords:** Getting rid of unnecessary words and symbols.
2. **Sentiment Analysis:**
   1. Using the TextBlob library to evaluate sentiment polarity, determining if the sentiment expressed in the text is positive, neutral, or negative.
   2. Based on polarity score, sentiment labels are assign to each review.
3. **Regression Analysis:**
   1. Applying Linear Regression to examine the relationship between sentiment polarity and product ratings.
   2. Linear Regression Model is trained to predict product ratings based on sentiment polarity score.
4. **Hyperparameter Tuning:**
   1. Hyperparameters controls the learning process of a machine learning algorithm.
   2. Using GridSearchCV to find the best hyperparameters for Linear Regression, enhancing the model's performance.
5. **N-gram Modeling:**
   1. Incorporating both unigrams and bigrams in the text vectorization process to capture more contextual details.
   2. N-grams are contiguous sequences of n items from a given sample of text.
   3. CountVectorizer is used to convert text data into numerical vector.
6. **Deep Learning:**
   1. Deep learning models, particulary recurrent neural networks.
   2. Building a Long Short-Term Memory (LSTM) model for sentiment classification.
   3. Improving the model by integrating LSTM with a Convolutional Neural Network (CNN) in an ensemble approach to boost performance.
7. **Visualization:**
   1. Using Matplotlib and Seaborn to visualize the regression analysis results, such as the connection between sentiment polarity and ratings.
   2. Plotting the training and validation loss curves during the training of deep learning models to monitor performance and avoid overfitting.

## ASSUMPTIONS

1. **Homogeneity of Sentiment Analysis**:
   * The assumption that sentiment analysis can be accurately performed based solely on the polarity score generated by TextBlob. This assumption implies that the sentiment expressed in a review is sufficiently captured by a single numerical value, disregarding potential nuances and complexities in language use and context.
2. **Linearity in Regression Analysis**:
   * The assumption of linearity between sentiment polarity and product ratings in the regression analysis. This assumption suggests that the effect of sentiment on ratings follows a linear relationship, implying that a unit change in sentiment polarity leads to a consistent change in the rating score. Nonlinear relationships between sentiment and ratings may not be adequately captured by linear regression.
3. **Effectiveness of Hyperparameter Tuning**:
   * The assumption that hyperparameter tuning, particularly for the fit\_intercept parameter in linear regression, significantly impacts model performance. This assumption presupposes that optimizing hyperparameters can lead to substantial improvements in the model's ability to predict ratings based on sentiment polarity, potentially overlooking other factors contributing to model performance.
4. **Generalization of Deep Learning Models**:
   * The assumption that the ensemble of LSTM and CNN models will generalize well to unseen data and outperform individual models. This assumption relies on the premise that combining different neural network architectures can effectively capture diverse features and patterns in the text data, leading to improved sentiment classification performance across various review texts.
5. **Effectiveness of Text Preprocessing**:
   * The assumption that the text preprocessing steps, including tokenization, POS tagging, lemmatization, and stopwords removal, adequately capture the relevant linguistic information for sentiment analysis. This assumption implies that the preprocessing techniques used effectively transform raw text data into a format suitable for subsequent analysis, potentially overlooking the impact of preprocessing choices on model performance and interpretability.

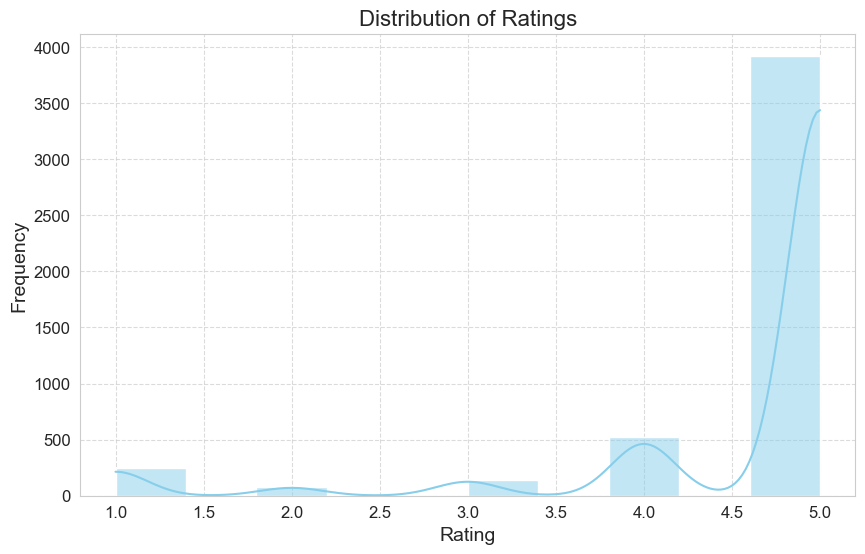
## 

## EXCEPTIONS/EXCLUSIONS

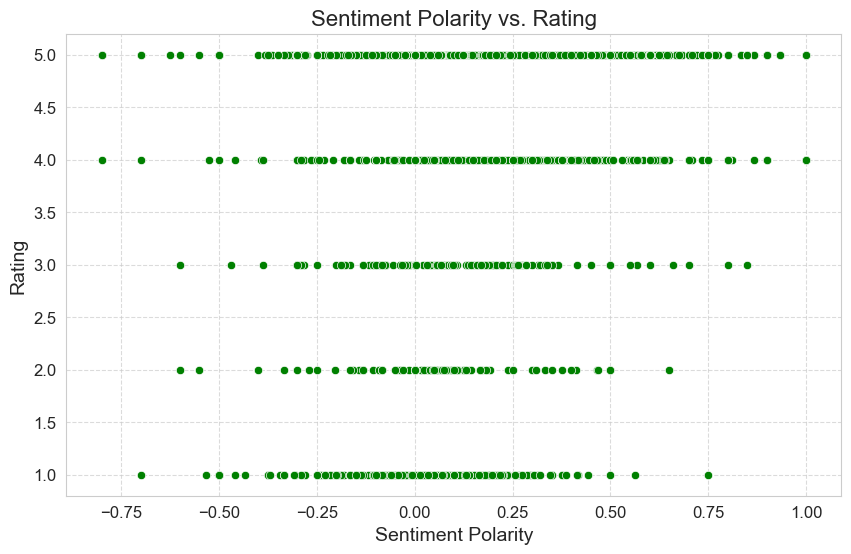
Here are the exceptions and exclusions to consider in the provided code:

1. **Exceptions**:
   * **Assumption of Linearity**: Linear regression assumes a linear relationship between sentiment polarity and product ratings. However, real-world relationships may be nonlinear, leading to potential inaccuracies in predictions.
   * **Assumption of Homogeneous Sentiment Analysis**: Sentiment analysis based solely on polarity scores may overlook nuances in language and context, leading to misinterpretations of sentiment in certain cases.
   * **Language Variations and Slangs**: Sentiment analysis model is trained on standard English may struggle to interpret reviews written in a informal language, dialects and slangs.
2. **Exclusions**:
   * **Domain-Specific Factors**: The code doesn't consider domain-specific factors that may influence sentiment and ratings, such as product type, brand reputation, or user demographics. Ignoring these factors could limit the model's predictive accuracy and generalizability.
   * **Cultural and Contextual Variations**: The code doesn't account for cultural or contextual variations in language use and sentiment expression, which could affect the effectiveness of sentiment analysis across different regions or communities. Incorporating cultural and contextual insights could enhance the model's robustness and applicability in diverse contexts.

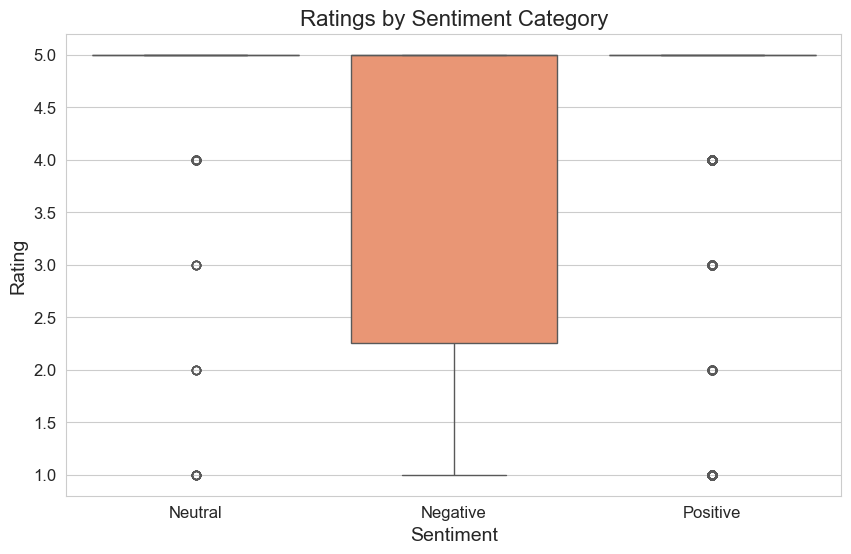
## CHARTS, TABLE AND DIAGRAMS



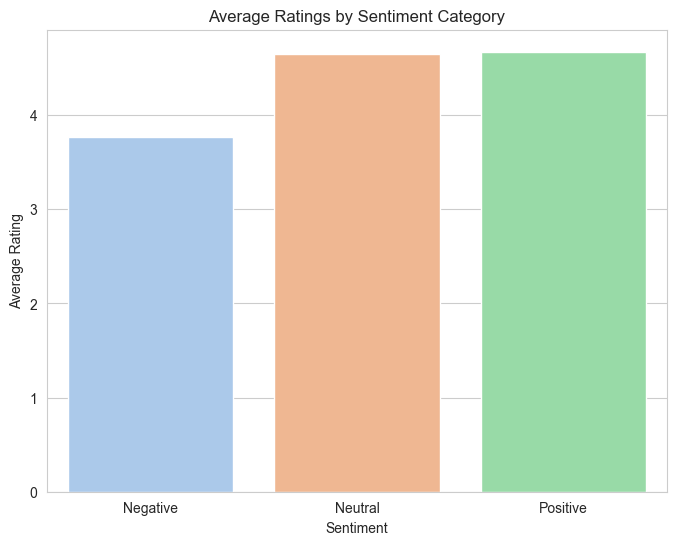
Histogram of customer ratings for various products on the online shopping platform “Amazon”. The graph display ratings ranging from 1.0 to 5.0 on the x-axis and their corresponding frequencies on the y-axis. Notably, there is a scarcity of ratings between 1.0 and 4.0, rating of 4.5-5.0 is most(sharp peak).



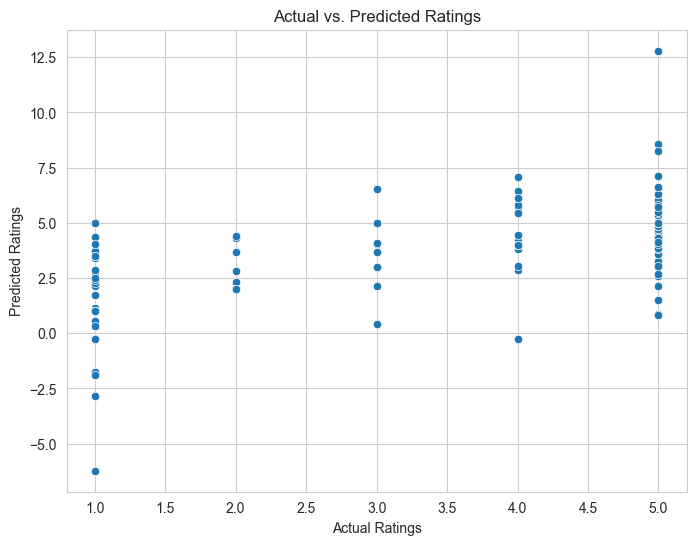
Scatter Plot titled “Sentiment Polarity vs Rating” which illustrates the relationship between the sentiment polarity (ranging from -1.00 to 1.00) and the rating (ranging from 1.0 to 5.0). Green dots represent data points, indicating varying sentiment polarities and ratings. Data points cluster around integer rating levels, with denser concentrations near zero sentiment polarity. The highest concentration of points aligns with the top rating (5.0 , 4.0 and 1.0), predominantly displaying positive sentiment polarities. Lower ratings (2.0 to 3.0) exhibit fewer data points dispersed across positive sentiment polarities, suggesting a broader range of opinions when ratings are lower.



Boxplot titled "Ratings by Sentiment Category" which illustrates the average ratings assigned to Positive, Negative, and Neutral sentiment categories. Positive sentiment and Neutral registers the highest average rating, followed by Negative sentiments. Notably, outliers are visible within the Positive and Neutral categories, signifying substantial deviations from the average ratings within these groups.



Vertical Bar Chart titled "Average Ratings by Sentiment Category," depicting the average customer sentiments categorized as Positive, Negative, or Neutral. The chart highlights that Positive sentiment garners the highest average rating, suggesting that customers are notably satisfied with the products or services. Following Positive sentiment, Neutral sentiments range between 4 and 5, indicating moderate satisfaction. Negative sentiments, ranging between 3 and 4, represent the lowest average ratings, suggesting dissatisfaction among customers.



Scatterplot for "Actual Ratings" against "Predicted Ratings" using blue dots to denote individual data points. Actual ratings ranging from 1.0 to 5.0 are plotted on the x-axis, while predicted ratings are represented on the y-axis within the -5.0 to 5.0 range. Notably, for lower actual ratings, such as 1.0, the predicted ratings exhibit wide dispersion, ranging from -5.0 to over 5.0, indicating varying levels of accuracy in the prediction model. Conversely, as actual ratings increase, particularly for ratings of 1.0 and 5.0, predicted ratings cluster closer to the actual values with less variation. Overall, while the model demonstrates greater accuracy for higher actual ratings, it lacks consistency across all rating levels, suggesting room for improvement.

## ALGORITHMS

The following algorithms have been used in the project:

1. **Tokenization**:
   * Algorithm: Word Tokenization using libraries like NLTK or spaCy.
   * Method: Splitting the text into individual words or tokens, preserving the semantic meaning of each word. For subword tokenization, algorithms like Byte Pair Encoding (BPE) or WordPiece can be used to handle rare or out-of-vocabulary words effectively.
2. **Lowercasing and Punctuation Removal**:
   * Algorithm: String Manipulation or Regular Expressions.
   * Method: Converting all text to lowercase ensures uniformity and reduces the vocabulary size. Removing punctuation marks eliminates non-alphanumeric characters that do not contribute to sentiment analysis.
3. **Stopwords Removal**:
   * Algorithm: Stopword Removal using predefined lists.
   * Method: Removing common stopwords helps in reducing noise and improving the quality of features used for sentiment analysis. Libraries like NLTK or spaCy provide built-in lists of stopwords for different languages.
4. **Stemming or Lemmatization**:
   * Algorithm: Porter Stemmer for stemming, WordNet Lemmatizer for lemmatization.
   * Method: Stemming reduces words to their base or root form by removing suffixes, while lemmatization reduces words to their canonical form based on a dictionary of word forms. Lemmatization is preferred for sentiment analysis tasks as it retains the semantic meaning of words.
5. **Model Architecture**:
   * Algorithm: Logistic Regression, Multinomial Naive Bayes, or Bernoulli Naive Bayes.
   * Method: Logistic Regression is a linear classification model that can handle binary or multiclass sentiment classification tasks effectively. Naive Bayes classifiers, such as Multinomial Naive Bayes and Bernoulli Naive Bayes, are probabilistic models based on Bayes' theorem and are well-suited for text classification tasks.
6. **Implementation using scikit-learn**:
   * Algorithm: Utilizing scikit-learn's **CountVectorizer** for feature extraction and **LogisticRegression**, **MultinomialNB**, or **BernoulliNB** for classification.
   * Method: Transforming preprocessed text data into numerical features using **CountVectorizer**, which converts text into a matrix of token counts. Then, training a classification model using **LogisticRegression** for binary classification or **MultinomialNB** / **BernoulliNB** for multinomial or binary Naive Bayes classification.

## 

## 

## 

## 

## CHALLENGES & OPPORTUNITY

1. **Sentiment Analysis Basics:**

**Challenges:** Deciphering emotions from text can be difficult because language is intricate and can have different meanings depending on the context.

**Opportunities:** Gaining insights into how people feel about products, services, or issues can help companies make more informed decisions and enhance customer satisfaction.

1. **Different Sentiment Analysis Approaches:**

**Challenges:** Different techniques for sentiment detection each have their own issues. Some may overlook key details, while others require significant amounts of data and time to be effective.

**Opportunities:** Combining multiple methods can yield a more accurate understanding of sentiment. Ongoing advancements in technology are continually improving the effectiveness of sentiment analysis.

1. **Rule-Based Methods:**

**Challenges:** Rules designed for sentiment analysis may not cover all possible scenarios and can be challenging to create and update.

**Opportunities:** These methods are simple and straightforward, and they can be particularly effective when the rules are well-defined.

1. **Lexicon-Based Approaches:**

**Challenges:** Predefined lists of words and their sentiment scores may not capture every way people express feelings and can miss the context of some words.

**Opportunities:** These methods are quick to implement and can provide a basic sense of sentiment, especially useful for analyzing large amounts of text.

1. **Machine Learning Methods:**

**Challenges:** Training algorithms to understand sentiment requires a lot of examples, and the models may sometimes misinterpret or get confused by the data.

**Opportunities:** With enough data and sophisticated algorithms, machines can effectively learn to detect sentiment and handle a wide range of scenarios and languages.

1. **Dataset Selection and Preprocessing:**

**Challenges:** Sourcing high-quality text samples with various emotions can be difficult, and preparing the text for analysis can be labor-intensive.

**Opportunities:** A well-chosen and diverse dataset enhances the learning process for models, leading to more accurate sentiment analysis results.

## 

## 

## 

## RISKS vs REWARDS

1. **Sentiment Analysis Basics**:
   * **Risk:** The risk lies in misinterpreting sentiment due to language complexities and subjective interpretations, leading to inaccurate insights and decisions.
   * **Reward:** The reward is gaining valuable insights into customer opinions, market trends, and brand perception, enabling informed decision-making and improved customer satisfaction.
2. **Different Sentiment Analysis Approaches**:
   * **Risk:** Using a single approach may lead to incomplete or biased results, while combining multiple approaches may introduce complexity and resource-intensive processes.
   * **Reward:** Leveraging various approaches allows for a more comprehensive understanding of sentiment, enhancing the accuracy and reliability of sentiment analysis results.
3. **Rule-Based Methods**:
   * **Risk:** Overreliance on predefined rules may result in oversimplified or inaccurate sentiment analysis, especially in cases with nuanced or context-dependent sentiment expressions.
   * **Reward:** Rule-based methods offer transparency and interpretability, providing clear guidelines for sentiment analysis and enabling straightforward implementation in specific domains or applications.
4. **Lexicon-Based Approaches**:
   * **Risk:** Dependency on sentiment lexicons may lead to limited coverage of sentiment expressions, potentially missing out on nuanced or domain-specific sentiment nuances.
   * **Reward:** Lexicon-based approaches are computationally efficient and can provide quick insights into sentiment, serving as a useful starting point for sentiment analysis tasks with limited resources.
5. **Machine Learning Methods**:
   * **Risk:** Machine learning models require large amounts of labeled data for training, and inaccuracies or biases in the training data can lead to poor performance and unreliable predictions.
   * **Reward:** With proper training and validation, machine learning models can achieve high accuracy in sentiment analysis, providing scalable and adaptable solutions for a wide range of applications.
6. **Dataset Selection and Preprocessing**:
   * **Risk:** Selecting biased or inadequate datasets may lead to models that generalize poorly or produce biased results, while improper preprocessing can introduce noise and distort sentiment analysis outcomes.
   * **Reward:** Well-curated datasets and effective preprocessing techniques improve the quality and reliability of sentiment analysis models, leading to more accurate insights and informed decision-making.

## 

## 

## 

## 

## 

## 

## 

## 

## 

## REFLECTION ON THE INTERNSHIP

1. **Learning Opportunities**:
   * The internship provided an excellent opportunity to learn about various aspects of sentiment analysis, including different approaches, methodologies, and algorithms.
   * Tasks such as researching sentiment analysis basics and understanding different approaches helped in gaining a solid foundation in natural language processing concepts.
2. **Hands-on Experience**:
   * Engaging in practical tasks such as dataset selection, preprocessing, and model implementation using machine learning libraries like scikit-learn provided valuable hands-on experience.
   * Preprocessing text data and training sentiment analysis models helped in understanding the complexities and challenges involved in real-world sentiment analysis projects.
3. **Problem-Solving Skills**:
   * Tackling challenges such as dataset selection, preprocessing, and model selection required critical thinking and problem-solving skills.
   * Addressing issues related to dataset biases, model performance, and feature engineering enhanced problem-solving abilities and analytical thinking.
4. **Collaboration and Communication**:
   * Collaborating with mentors and colleagues to discuss ideas, seek feedback, and troubleshoot issues fostered teamwork and communication skills.
   * Presenting findings and insights from sentiment analysis tasks to stakeholders helped in honing presentation and communication skills.
5. **Reflection on Challenges**:
   * Overcoming challenges related to dataset selection, preprocessing, and model implementation highlighted the importance of thorough planning, attention to detail, and adaptability.
   * Dealing with uncertainties and limitations in sentiment analysis methods underscored the need for continuous learning and exploration of innovative techniques.
6. **Future Growth Opportunities**:
   * The internship experience laid a strong foundation for further exploration and specialization in sentiment analysis and natural language processing.
   * Building upon the knowledge and skills gained during the internship, there are opportunities for future growth in research, industry projects, and advanced studies in related fields.

## 

## RECOMMENDATIONS

1. **Keep Learning:** Stay abreast of the latest developments and methods in sentiment analysis by exploring research papers and enrolling in online courses.
2. **Practice Hands-on:** Take on personal projects and join competitions to put your knowledge into practice.
3. **Network:** Build connections with professionals and peers in the field to exchange ideas and find mentorship opportunities.
4. **Build a Portfolio:** Create a well-organized portfolio to highlight your projects and skills in sentiment analysis.
5. **Specialize:** Consider focusing on niche areas within sentiment analysis to deepen your expertise.
6. **Invest in Professional Development:** Attend workshops and seek out opportunities for both personal and career growth.

## OUTCOME/CONCLUSIONS

1. **Understanding Sentiment Analysis Basics**:
   * Gain foundational knowledge about sentiment analysis, its importance, and applications in natural language processing.
2. **Exploring Different Sentiment Analysis Approaches**:
   * Learned about rule-based, lexicon-based, machine learning, and deep learning methods for sentiment analysis, understanding their strengths and weaknesses.
3. **Diving Deeper into Rule-Based and Lexicon-Based Methods**:
   * Explored how rule-based methods rely on predefined rules and patterns, and how lexicon-based approaches use sentiment lexicons or dictionaries to associate words with sentiment scores.
4. **Studying Machine Learning Methods for Sentiment Analysis**:
   * Investigated machine learning techniques such as logistic regression, Naive Bayes, and Support Vector Machines for sentiment analysis, focusing on their implementation and performance.
5. **Identifying Suitable Datasets and Preprocessing Data**:
   * Explored the challenges and opportunities in dataset selection and preprocessing for sentiment analysis, including tokenization, stop word removal, and stemming/lemmatization.
6. **Choosing and Implementing Model Architecture**:
   * Selected basic model architectures like logistic regression or Naive Bayes for sentiment classification and implemented them using scikit-learn in Python.

## ENHANCEMENT SCOPE

Some potential enhancement scopes for further improving the sentiment analysis are:

1. **Advanced Models**: Explore RNNs, CNNs, or transformer-based models for better performance.
2. **Domain-Specific Analysis**: Specialize in analyzing sentiment in specific domains like healthcare or finance.
3. **Fine-Tuning Pretrained Models**: Adapt pretrained models to specific tasks using transfer learning.
4. **Aspect-Based Analysis**: Analyze sentiment at a more granular level, focusing on specific aspects or entities.
5. **Multimodal Analysis**: Combine text with other modalities like images or audio for richer sentiment analysis.
6. **Real-Time Analysis**: Develop systems for analyzing sentiment in streaming data in real-time.
7. **Ethical Considerations**: Address biases and ethical concerns in sentiment analysis algorithms to ensure fairness and privacy.

**LINK TO THE EXECUTABLE FILE**

**Repository Link:** [**https://github.com/YashSathe1/RIO-125**](https://github.com/YashSathe1/RIO-125)

**RESEARCH QUESTIONS AND RESPONSES**

**Question:** How does sentiment analysis contribute to understanding customer feedback in the e-commerce industry?

**Responses:**

1. Sentiment analysis helps e-commerce businesses gain insights into customer opinions and emotions expressed in product reviews, allowing them to identify trends, strengths, and areas for improvement.
2. By analyzing sentiment, e-commerce companies can gauge customer satisfaction levels, identify common pain points, and tailor their products and services to meet customer expectations more effectively.