

American International University-Bangladesh (AIUB)

# Sentiment Analysis on Social Media Data for Brand Monitoring

***Subtitle goes here***

Rahaman, Hasibur (19-40823-2)

Rahman, Naimur (19-40397-1)

Sithi, Sharannaya Dey (19-40811-2)

Rahman, AKM Shahriyar (19-40787-2)

*A Thesis submitted for the degree of Bachelor of Science (BSc) in Computer Science and Engineering (CSE) at*

*American International University Bangladesh in July, 2021*

Faculty of Science and Technology (FST)

## Abstract

*(All candidates must edit this page )*

The abstract should outline the main approach and findings of the thesis / software project and normally must be between **300** and **800** words.

An abstract is a summary of your entire thesis and should provide a complete overview of the thesis, including your key results and findings.

An abstract is different to your introduction, and shouldn’t be used to advertise your thesis — it should provide enough information to allow readers to understand what they’ll learn by reading the thesis.

Your abstract should answer the following questions:

1. What did you do?
2. How did you do it?
3. Why was it worth doing?
4. What were the key results?
5. What are the implications or significance of the results?

As your abstract will have a word limit, you may be unable to answer every question in detail. If you find yourself running out of words, make sure you include your key findings before other information.

## Declaration by author

*(All candidates to reproduce this section in their thesis verbatim)*

This thesis is composed of our original work, and contains no material previously published or written by another person except where due reference has been made in the text. We have clearly stated the contribution of others to our thesis as a whole, including statistical assistance, survey design, data analysis, significant technical procedures, professional editorial advice, financial support and any other original research work used or reported in our thesis. The content of our thesis is the result of work we have carried out since the commencement of Thesis / Software project.

We acknowledge that copyright of all material contained in my thesis resides with the copyright holder(s) of that material. Where appropriate we have obtained copyright permission from the copyright holder to reproduce material in this thesis and have sought permission from co-authors for any jointly authored works included in the thesis.

## Approval

*(All candidates must edit this page )*

The thesis titled **“Thesis title goes here”** has been submitted to the following respected members of the board of examiners of the department of computer science in partial fulfilment of the requirements for the degree of Master of Science in Computer Science on **(date of defence)** and has been accepted as satisfactory.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Supervisor’s Name  Rank & Position | |  | Name of the External  Rank & Position | |
|  | | | | |
|  | Name of the Head (UG/PG)  Rank & Position | | |  |
| Prof. Dr. Dip Nandi  Associate Dean  Faculty of Science and Technology  American International University-Bangladesh | |  | Mashiour Rahman  Sr. Associate Professor & Dean-in-charge  Faculty of Science and Technology  American International University-Bangladesh | |

## Publications included in this thesis

If you choose to include publications as part of your thesis use this section to detail accepted or in press publication(s) using the standard citation format for your discipline.

Papers submitted for publication and awaiting review should appear in the next section, **Submitted manuscripts included in this thesis**.

If you have not included any of your publications in the thesis then state “No publications included”.

#### Example:

1. [[Dum, 2021](#_bookmark17)] **Your Name**, Co-author 1, and Final Author, Title of your paper, *Journal*, Issue, Number, Year
2. [[Dum, 2021](#_bookmark17)] **Your Name**, Co-author 1, and Final Author, Title of your paper, *Journal*, Issue, Number, Year

## Submitted manuscripts included in this thesis

List manuscript/s submitted for publication here. As described above for **Publications included in the thesis**, on the page immediately preceding the chapter that includes the submitted manuscript, in no more than one (1) page, detail your contribution to the authorship if you are not the sole author.

If you have no submitted manuscripts from your candidature then state “No manuscripts submitted for publication”.

#### Example:

1. [[Dum, 2021](#_bookmark17)] **Your Name**, Co-author 1, and Final Author, Title of your paper, submitted to

*Journal* on 4th June 2018.

## Other publications during candidature

List other publications arising during your candidature using the standard citation format for your dis- cipline. Divide your publications into sub-sections as appropriate in your discipline *e.g.* peer-reviewed papers, book chapters, conference abstracts. Papers submitted for publication and awaiting review are not considered publications and cannot be included in this section.

If you have no publications from your candidature then state “No other publications”.

**Example:**

### Conference abstracts

1. [[Dum, 2021](#_bookmark17)] **Your Name**, Co-author 1, and Final Author, Title of your conference paper,

*Proceedings of Conference*, other details.

### Book chapters

1. [[Dum, 2021](#_bookmark17)] **Your Name**, Co-author 1, and Final Author, Title of your chapter, Book, editor,

*etc.*.

## Research involving human or animal subjects

All reserch involving human or animal subjects requires prior ethical review and approval by an independent review committee. At UQ, the relevant committee for research involving human subjects is the [Human Ethics Unit](https://research.cs.aiub.edu/human-ethics) and the relevant committee for research involving animal subjects is the relevant [Animal Ethics Committee](https://research.cs.aiub.edu/animal-welfare). Please provide details of any ethics approvals obtained including the ethics approval number and name of approving committees. A copy of the ethics approval letter must be included in the thesis appendix.

If no human or animal subjects were involved in this research please state: “No animal or human subjects were involved in this research”.

## Contributions by authors to the thesis

List the significant and substantial inputs made by different authors to this research, work and writing represented and/or reported in the thesis. These could include significant contributions to: the conception and design of the project; non-routine technical work; analysis and interpretation of research data; drafting significant parts of the work or critically revising it so as to contribute to the interpretation.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Full Name** | **Full Name** | **Full Name** | **Full Name** | **Contribution (%)** |
|  | *xx-xxxxx-x* | *xx-xxxxx-x* | *xx-xxxxx-x* | *xx-xxxxx-x* |
| Conceptualization |  |  |  |  | 100 % |
| Data curation |  |  |  |  | 100 % |
| Formal analysis |  |  |  |  | 100 % |
| Investigation |  |  |  |  | 100 % |
| Methodology |  |  |  |  | 100 % |
| Implementation |  |  |  |  | 100 % |
| Validation |  |  |  |  | 100 % |
| Theoretical derivations |  |  |  |  | 100 % |
| Preparation of figures |  |  |  |  | 100 % |
| Writing – original draft |  |  |  |  | 100 % |
| Writing – review & editing |  |  |  |  | 100 % |

If your task breakdown requires further clarification, do so here. Do not exceed a single page.

## Acknowledgments

Acknowledgements recognize those who have been instrumental in the completion of the project. Acknowledgements should include any professional editorial advice received including the name of the editor and a brief description of the service rendered.

## Keywords

Maximum 10 words; use lower case throughout, separating words/phrases with commas. For example: word, word word, word, word, word word

Table of Contents

[Title of the research i](#_Toc76225891)

[Abstract ii](#_Toc76225892)

[Declaration by author iii](#_Toc76225893)

[Approval iv](#_Toc76225894)

[Publications included in this thesis v](#_Toc76225895)

[Submitted manuscripts included in this thesis v](#_Toc76225896)

[Other publications during candidature vi](#_Toc76225897)

[Conference abstracts vi](#_Toc76225898)

[Book chapters vi](#_Toc76225899)

[Research involving human or animal subjects vi](#_Toc76225900)

[Contributions by authors to the thesis vii](#_Toc76225901)

[Acknowledgments viii](#_Toc76225902)

[Keywords ix](#_Toc76225903)

[List of Tables xii](#_Toc76225904)

[List of Abbreviations and Symbols xiv](#_Toc76225905)

[Introduction 1](#_Toc76225906)

[1.1 Your thesis topic 1](#_Toc76225907)

[Literature review 3](#_Toc76225908)

[2.1 Introduction 3](#_Toc76225909)

[Methods 6](#_Toc76225910)

[3.1 Introduction 8](#_Toc76225911)

[Results or findings 9](#_Toc76225912)

[4.1 Introduction 9](#_Toc76225913)

[Discussion 12](#_Toc76225914)

[5.1 Introduction 12](#_Toc76225915)

[Conclusion 14](#_Toc76225916)

[Bibliography 16](#_Toc76225917)

[Appendix 18](#_Toc76225918)

[A.1 Name of Appendix-1 18](#_Toc76225919)

[Example of Citations 21](#_Toc76225920)

[Example of Equations 23](#_Toc76225921)

[Example of Figures 25](#_Toc76225922)

[Example of Tables 27](#_Toc76225923)

[Example of algorithm procedure 30](#_Toc76225924)

[Example of Code 33](#_Toc76225925)

[G.1 Find the greatest number from a list of numbers in *Python* 33](#_Toc76225926)

**List of Figures**

[Figure D. 1 : American International University-Bangladesh (AIUB) 36](#_Toc76225927)

[Figure D. 2: Example of placing images side by side 38](#_Toc76225928)

# List of Tables

[Table E. 1: AIUB currently operates under four distinct Faculties x 41](#_Toc76193928)

[Table E. 2: Row span example 41](#_Toc76193929)

[Table E. 3 : Test Table 41](#_Toc76193930)

# List of Abbreviations and Symbols

Mention all the abbreviations and the different symbols that is used in this document.

Abbreviations

CS Computer Science

CSE Computer Science and Engineering

NN Neural Network

HCI Human Computer Interaction

NLP Natural Language Processing

*etc. etc.*

Symbols

*ρ*ˆ Density operator

® Convolution

*etc. etc.*

**Chapter 1**

# Introduction

It's now common and popular practice to read online customer reviews and ratings before making a purchase. Based on feedback, consumers are more likely to purchase a product. Because online marketplaces have grown in popularity over the past few decades, many online retailers and sellers ask their customers to leave reviews of the goods they have purchased. Millions of reviews are generated every day throughout the Internet regarding various goods, services, and locations. Due to this, the internet is the most reliable resource for ideas and reviews regarding a good or service. Product reviews offer valuable insights into a product's features, quality, and recommendations, helping potential buyers gain a comprehensive understanding of it. These reviews benefit not only consumers but also sellers, particularly those who manufacture their own products, by providing a better understanding of consumer preferences and needs. However, it is getting harder for a customer to decide whether or not to purchase a product as more reviews become available for it. Customers are more confused when trying to make the right decision due to conflicting opinions about the same product and unclear reviews. In this case, it appears that all e-commerce enterprises must analyze this content.

## 1.1 Motivation

Introduce your topic.

## 1.2 Objective

This project aims to perform sentiment analysis on Amazon reviews of Smartphones using two different approaches: **VADER** (valence-aware dictionary and sentiment Reasoner) and a pre-trained **RoBERTa** (Robustly optimized BERT approach) model.

## 1.3 Research Questions

Introduce your topic.

## 1.4 Research Questions

The rest of the document is as follows. Chapter 2 provides a literature analysis of the sentiment analysis as well as relevant research, journals, and articles reviewed for the idea. Chapter 3 explains justification and explanation of the methodological approach. The results from the experiments are gathered in Chapter 4 and discussed in Chapter 5. Finally, Chapter 6 concludes the study.

.

**Chapter 2**

# Literature review

The literature review should provide a more detailed analysis of research in the field, and present more specific aims or hypotheses for your research. What’s expected for a literature review varies depending on your program – a Master’s thesis requires a more extensive literature review than a BSC thesis.

Introduce the broad layout of the chapter.

## Introduction

Add your text here.

**Chapter 3**

# Methods

This chapter gives the reader an overview of the different tools and procedures used to accomplish the Sentiment Analysis on Social Media Data for Brand Monitoring project. The reader will also become familiar with each technology's basic overview and the design choices made to achieve it. In the first section, the programming environments will be discussed. The second section will cover about the data collection procedure. And then in the third section Data Cleaning Procedure will be explain. The fourth section will explain the RoBERTa and VADER models. The entire methodology used for this research will be explained in the final section.

## Programming Environment and Technologies

One of the most popular programming languages for data science and machine learning is Python. Numerous machine learning algorithms can be solved using the extensive library of Python. That’s why we have chosen python as the programming language for this research purpose. Python, libraries that we have used in this project are,

* + 1. Pandas,
    2. BeautifulSoup
    3. Matplotlib,
    4. Seaborn,
    5. NLTK (Natural Language Toolkit),
    6. Transformers,
    7. Torch,
    8. TensorFlow,
    9. Flax
    10. tdqm

## Data Collection

Data was collected from amazon website by scrapping Python code as Amazon does not have an API like Twitter to download reviews with. Later, the dataset was saved in the Comma Separated Values (CSV) format because Python can handle these kinds of files more easily. The dataset consists of Smartphone reviews. The fields listed below are included in the dataset:

* + 1. **product\_title:** Name of the product is stored here
    2. **user\_name:** Name of the reviewer is stored in this field
    3. **rating:** rating of the product
    4. **review:** text of the review

The dataset contains a total of 1,045 reviews all the data are selected for analysis in this project.

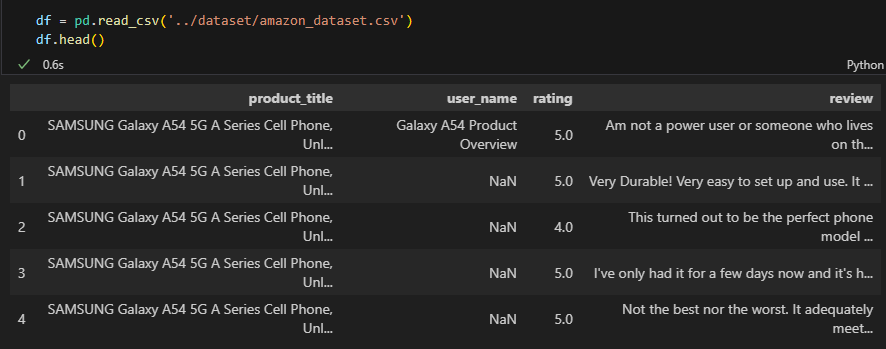


Fig-1: Sample for collected data

## Data Cleaning and Preprocessing

As we have collected the dataset by scraping website so the data is not clean. We had cleaned the data using regular expression and stop words. At first, we have cleaned product\_title and user\_name by keeping only the first 3 words and replacing the empty user\_name with ‘Amazon user’. Then convert the review text into lower case, remove numbers, punctuation mark, extra spaces and Strip leading and trailing spaces. After that we have tokenized each review, removed the stop words and joined filtered words back into a single string. At last, we have saved this data into another CSV file name amazon\_datasets.csv. After that the data will be ready for the next process of sentiment analysis.

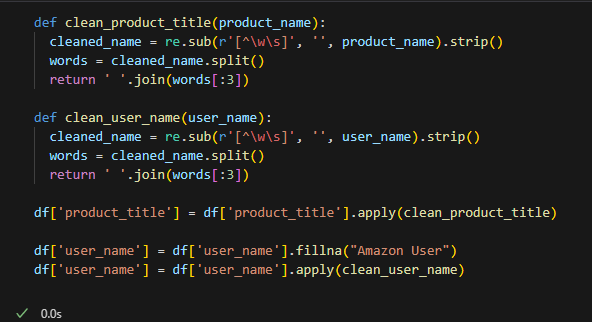


Fig-2: user\_name and product\_title data cleaning

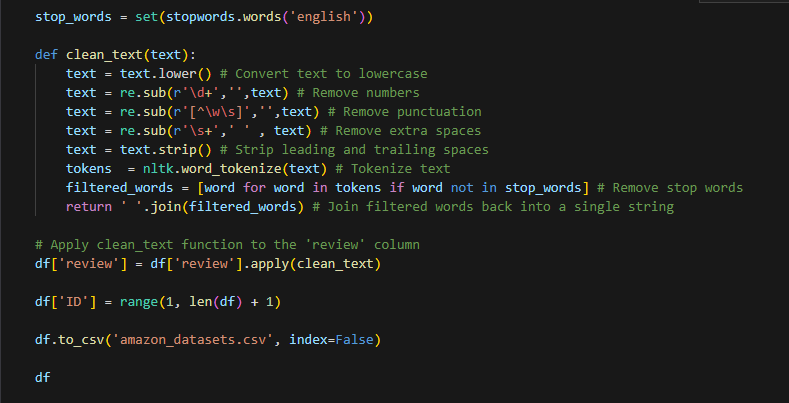


Fig-3: Cleaning reviews and insert into CSV file

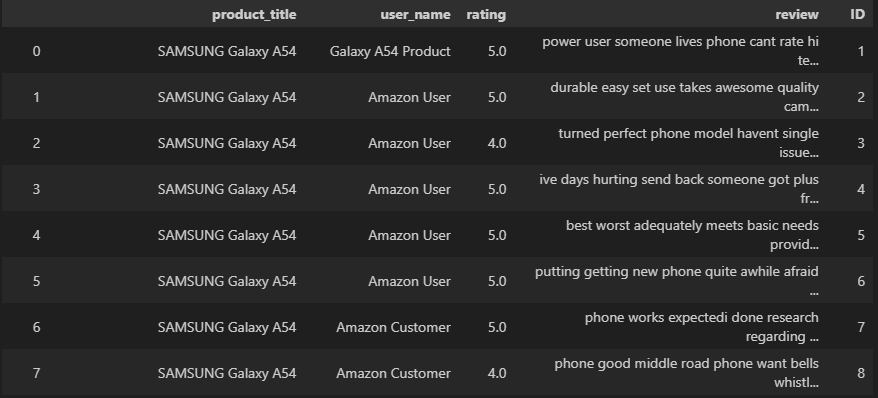


Fig-4: Sample of Cleaned data

## Sentiment Analysis Models

* + 1. **Vader Model:**

VADER (Valence Aware Dictionary and Sentiment Reasoner) is a rule-based sentiment analysis tool that is specifically attuned to the sentiments expressed in social media. Developed by C.J. Hutto and Eric Gilbert in 2014, VADER is designed to be effective for analyzing sentiment in texts that are typically informal, such as tweets, Facebook posts, online reviews, and other types of short content. Unlike many other sentiment analysis tools, VADER is capable of handling emojis, slang, acronyms, and various other elements that are common in social media text.

VADER provides four sentiment metrics for a given text:

1. **Positive (pos):** Proportion of the text that conveys a positive sentiment.
2. **Negative (neg):** Proportion of the text that conveys a negative sentiment.
3. **Neutral (neu):** Proportion of the text that is neutral in sentiment.
4. **Compound (compound):** A normalized, weighted composite score that represents the overall sentiment of the text. The compound score is computed by summing the valence scores of each word in the text, adjusted according to various rules, and then normalized to range from -1 (most extreme negative) to +1 (most extreme positive).
   * 1. **RoBERTa Model**:

Facebook AI created the cutting-edge natural language processing (NLP) model known as RoBERTa, or Robustly Optimized BERT Pretraining Approach, in 2019. It is an enhanced version of the BERT (Bidirectional Encoder Representations from Transformers) model that improves performance on various NLP tasks by incorporating major enhancements to the BERT architecture and pretraining process. By altering a number of crucial hyperparameters and training techniques, RoBERTa outperforms BERT and makes use of more data, bigger batch sizes, and more training iterations. RoBERTa is able to outperform other NLP benchmark tasks like sentiment analysis, question answering, and natural language inference thanks to these optimizations.

RoBERTa (Robustly Optimized BERT Pretraining Approach) itself is a general-purpose language representation model and does not inherently perform sentiment analysis. Instead, it is a powerful transformer-based model that can be fine-tuned on specific tasks, including sentiment analysis, by using a labeled dataset.

## Methodology

The whole methodology that is applied during this sentiment analysis research will be explained here line by line.

**Data Exploration and Processing:**

* + 1. **Import relevant Libraries:**

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

plt.style.use("ggplot")

import nltk

import re

**pandas**, **matplotlib**, and **seaborn** are commonly used libraries for data manipulation, numerical computation, and visualization.

The line **plt.style.use('ggplot')** sets the plotting style to emulate the visual aesthetic of the popular R package, ggplot2. This style choice results in plots with a distinctive appearance characterized by bold lines, a gray background, and a combination of colorful elements.

**nltk** is the Natural Language Toolkit, which provides tools for natural language processing tasks.

**Read the data:**

df = pd.read\_csv('../dataset/amazon\_dataset.csv')

df.head()

**Find the data size:**

df.shape

Here the dataset has 1045 rows and 4 columns.

**Generate a bar plot** showing the distribution of review scores, this helps to visualize the distribution of reviews based on star ratings.

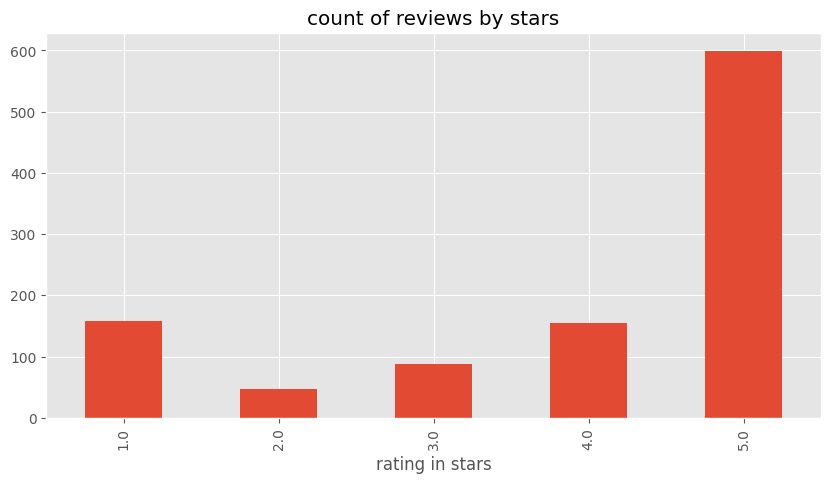
ax = df['rating'].value\_counts().sort\_index().plot(kind= 'bar' ,

                                        title= 'count of reviews by stars' ,

                                        figsize= (10,5))

ax.set\_xlabel('rating in stars')

plt.show()



This shows that most of the reviews are positive, and the negative reviews are very few.

* + 1. **Data cleaning and preprocessing:**

**Import and download necessary nltk resources:**

from nltk.corpus import stopwords

# nltk.data.path.append('c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('vader\_lexicon', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('words', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('maxent\_ne\_chunker', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('punkt', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('averaged\_perceptron\_tagger', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('stopwords', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

nltk.download('punkt\_tab', download\_dir='c:/Users/Hasib282/anaconda/nltk\_data')

**Clean the product title and user\_name field of the datasets.**

def clean\_product\_title(product\_title):

  cleaned\_name = re.sub(r'[^\w\s]', '', product\_title).strip()

  words = cleaned\_name.split()

  return ' '.join(words[:3])

def clean\_user\_name(user\_name):

  cleaned\_name = re.sub(r'[^\w\s]', '', user\_name).strip()

  words = cleaned\_name.split()

  return ' '.join(words[:3])

df['product\_title'] = df['product\_title'].apply(clean\_product\_title)

df['user\_name'] = df['user\_name'].fillna("Amazon User")

df['user\_name'] = df['user\_name'].apply(clean\_user\_name)

**Clean the reviews of datasets and upload the clean data into a new CSV file.**

stop\_words = set(stopwords.words('english'))

def clean\_text(text):

    text = text.lower() # Convert text to lowercase

    text = re.sub(r'\d+','',text) # Remove numbers

    text = re.sub(r'[^\w\s]','',text) # Remove punctuation

    text = re.sub(r'\s+',' ' , text) # Remove extra spaces

    text = text.strip() # Strip leading and trailing spaces

    tokens  = nltk.word\_tokenize(text) # Tokenize text

    filtered\_words = [word for word in tokens if word not in stop\_words] # Remove stop words

    return ' '.join(filtered\_words) # Join filtered words back into a single string

# Apply clean\_text function to the 'review' column

df['review'] = df['review'].apply(clean\_text)

df['ID'] = range(1, len(df) + 1)

df.to\_csv('amazon\_datasets.csv', index=False)

The whole preprocessing method is described in the section no 3.3. Read it there.

* + 1. **Vader Model implementation:**

**Initialize Vader Sentiment Analyzer:**

from nltk.sentiment import SentimentIntensityAnalyzer

from tqdm import tqdm

sia = SentimentIntensityAnalyzer()

**SentimentIntensityAnalyzer** is a pre-trained model included in nltk for performing sentiment analysis on text data.

**tqdm** library provides a fast, extensible progress bar for Python and wraps around the iterable object to provide a progress indicator during iterations.

**sia = SentimentIntensityAnalyzer()** creates an instance of the **SentimentIntensityAnalyzer** class and assign it to the variable sia. This analyzer is capable of analyzing the sentiment of text data by assigning polarity scores, such as positive, negative, neutral, and compound scores, to each piece of text.

**Apply Vader:**

#Run the polarity score on the whole dataset

res = {}

for i, row in tqdm(df.iterrows() , total= len(df)):

    review = row['review']

    myid   = row['ID']

    res[myid] = sia.polarity\_scores(review)

This code snippet iterates over each row in the DataFrame df, retrieves the text content of each review, and performs sentiment analysis using the SentimentIntensityAnalyzer (sia). It then stores the sentiment scores for each review in a dictionary called res, with the review ID ('Id' column) as the key.

**Plot Vader:** Plot a data frame for the Vader results and merge it with the original data frame.

vaders = pd.DataFrame(res).T

vaders= vaders.reset\_index().rename(columns={'index' : 'ID'})

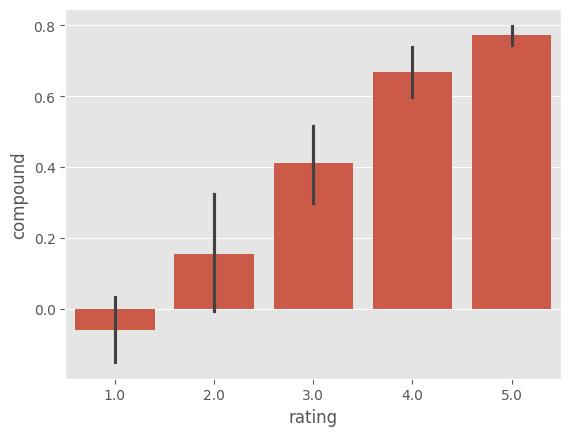
vaders = vaders.merge(df,how='left')

**Visualize Vader:** Plot a bar graph to visualize the results.

sns.barplot(data=vaders , x='rating' , y = 'compound')

ax.set\_title('Compound score by Amazon star review')

plt.show()



**Sub-plot for each category**: Create a subplot with three bar plots showing the distribution of positive, neutral, and negative sentiment scores across different review scores.

fig, axs=plt.subplots(1, 3 , figsize=(15 , 3))

sns.barplot(data=vaders , x='rating' , y= 'pos' , ax=axs[0])

sns.barplot(data=vaders , x='rating' , y= 'neu' , ax=axs[1])

sns.barplot(data=vaders , x='rating' , y= 'neg' , ax=axs[2])

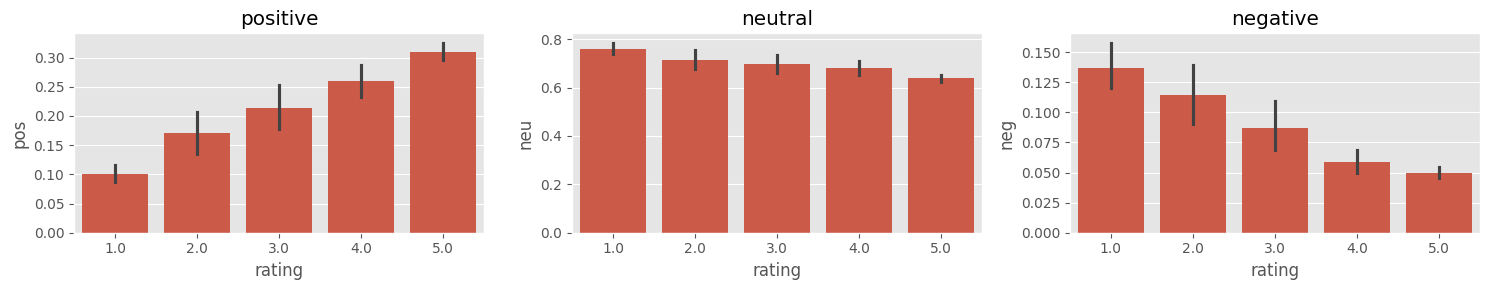
axs[0].set\_title('positive'),

axs[1].set\_title('neutral'),

axs[2].set\_title('negative'),

plt.tight\_layout()

plt.show()



* + 1. **Roberta Model Implementation:** Install necessary packages and dependencies required for using the Roberta pre-trained model.

# pip install transformers

# pip install torch

# pip install tensorflow

# pip install flax

# pip install --upgrade tensorflow-intel

# pip install ml-dtypes==0.2.0

import torch

import sys

from transformers import AutoTokenizer

from transformers import AutoModelForSequenceClassification

from scipy.special import softmax

The code begins by using the pip install command to install several Python packages. These packages include:

* torch: PyTorch, a popular open-source machine learning library.
* tensorflow: TensorFlow, another widely-used machine learning library developed by Google.
* flax: Flax, a neural network library that is tightly integrated with JAX, a high-performance numerical computing library.
* tensorflow-intel: An optimized version of TensorFlow for Intel architectures.
* ml-dtypes==0.2.0: A specific version of the ml-types library.

**Initialize RoBERTa Model:**

MODEL = "cardiffnlp/twitter-roberta-base-sentiment"

tokenizer = AutoTokenizer.from\_pretrained(MODEL)

model = AutoModelForSequenceClassification.from\_pretrained(MODEL)

* **Define the Model:** The variable MODEL specifies the name of the pre-trained model to be loaded. In this case, it's "cardiffnlp/twitter-roberta-base-sentiment", which refers to a RoBERTa model fine-tuned on Twitter data for sentiment analysis.
* **Load the Tokenizer**: The AutoTokenizer.from\_pretrained() function is used to load the tokenizer associated with the specified pre-trained model.
* **Load the Model**: Similarly, the AutoModelForSequenceClassification.from\_pretrained() function loads the pre-trained model for sequence classification. This model has been fine-tuned on sentiment analysis tasks and is capable of classifying the sentiment of a given text sequence into categories like positive, negative, or neutral.

**Create a function for finding polarity of Roberta Model:**

# Apply to the whole Datasets

def polarity\_scores\_roberta(example):

    encoded\_text = tokenizer(example, return\_tensors="pt" ,padding="max\_length", max\_length=512, truncation=True)

    output = model(\*\*encoded\_text)

    rating = output[0][0].detach().numpy()

    rating = softmax(rating)

    rating\_dict = {

        'roberta\_neg' : rating[0],

        'roberta\_neu' : rating[1],

        'roberta\_pos' : rating[2]

    }

    return rating\_dict

**Run Roberta Model on Data:**

for i, row in tqdm(df.iterrows(), total=len(df)):

  try:

    review = row['review']

    myid   = row['ID']

    vader\_result = sia.polarity\_scores(review)

    vader\_result\_rename = {}

    for key, value in vader\_result.items():

        vader\_result\_rename[f"vader\_{key}"] = value

    roberta\_result = polarity\_scores\_roberta(review)

    both = {\*\*vader\_result\_rename , \*\*roberta\_result}

    res[myid] = both

  except RuntimeError:

    print(f'Broke for Id {myid}')

**Create Dataframe to Plot the result:**

result\_df = pd.DataFrame(res).T

result\_df= result\_df.reset\_index().rename(columns={'index' : 'ID'})

result\_df = result\_df.merge(df,how='left')

pd.set\_option('display.max\_rows', None)

pd.set\_option('display.max\_columns', None)

result\_df

* + 1. Plot a Pair Plot to compare the result:

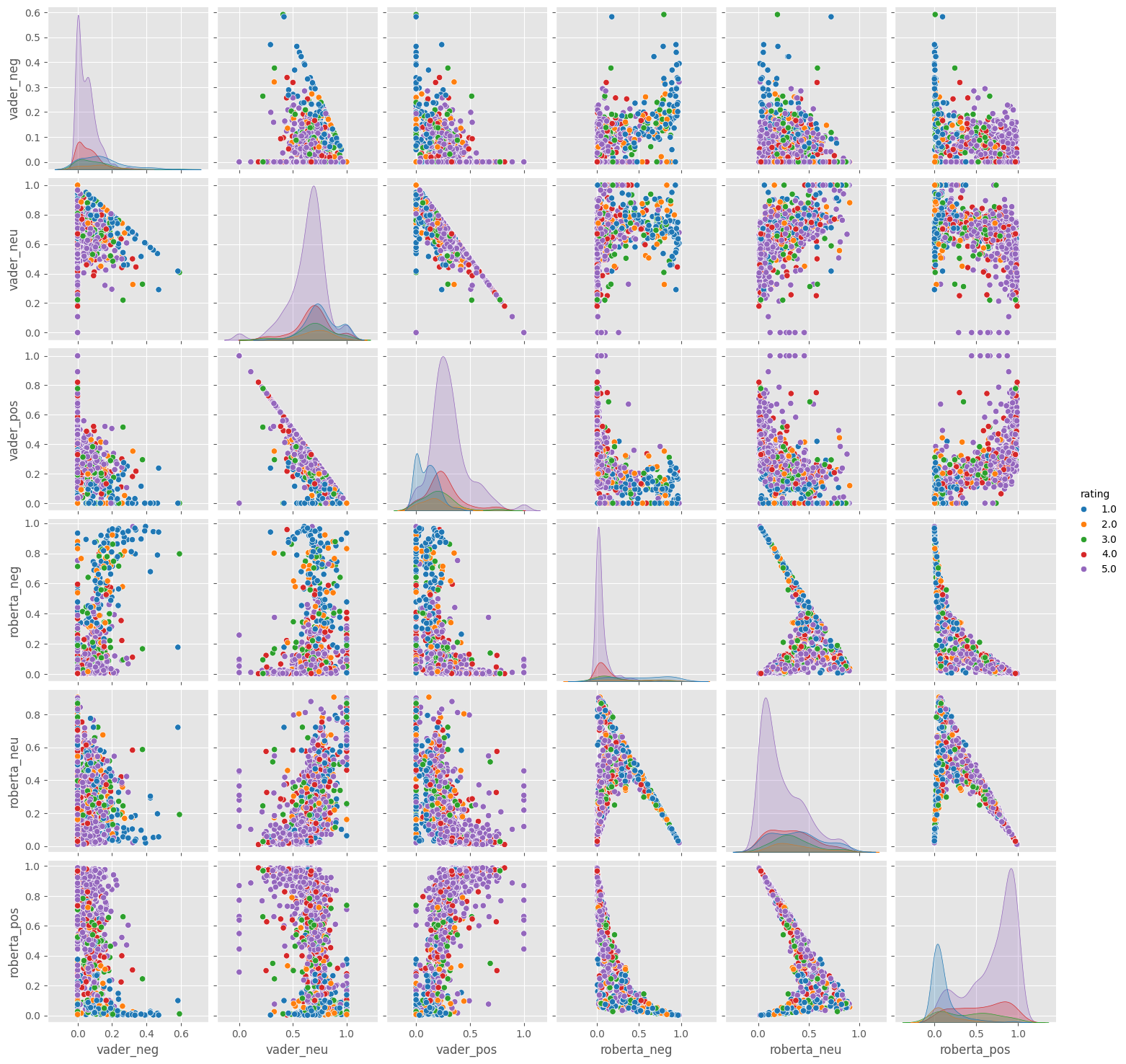
sns.pairplot(data=result\_df,

             vars=['vader\_neg', 'vader\_neu', 'vader\_pos', 'roberta\_neg', 'roberta\_neu', 'roberta\_pos'],

             hue='rating',

             palette="tab10")

plt.show()



**Chapter 4**

# Results or findings

Use the results section to:

* + 1. specify the data you collected and how it was were prepared for analysis
    2. describe the data analysis (e.g. define the type of statistical test that was applied to the data)
    3. describe the outcome of the analysis
    4. present a summary and descriptive statistics in a table or graph.

#### Use tables and figures effectively

Reports usually include tables, graphs and other graphics to present data and supplement the text. To learn how to design and use these elements effectively, see examples provided in Appendix B ([D](#_bookmark27), [E](#_bookmark30), [C](#_bookmark26), [F](#_bookmark34)).

Introduce the broad layout of the chapter.

## Introduction

Add your text here.

**Chapter 5**

# Discussion

Use the discussion section to:

* + - * 1. comment on your results and explain what they mean
        2. compare, contrast and relate your results back to theory or the findings of other studies
        3. identify and explain any unexpected results
        4. identify any limitations to your research and any questions that your research was unable to answer
        5. discuss the significance or implications of your results.
        6. If you find that your research ends up in a different direction to what you intended, it can help to explicitly acknowledge this and explain why in this section.

Introduce the broad layout of the chapter.

## Introduction

Add your text here.

**Chapter 6**

# Conclusion

Use the conclusion section to:

* + 1. summarise the main findings of your research
    2. emphasise that you’ve met your research aims. A good strategy is to repeat your research questions and demonstrate how your findings answer them.
    3. restate the limitations of your research and make suggestions for further research.

In some cases, the discussion and conclusion sections can be combined. Check with your supervisor if you want to combine these sections. your conclusion chapter should not exceed two pages.

Conclude your thesis.

# Bibliography

[Dum, 2021] (2021). Demo title.

[Hasan et al., 2013] Hasan, K. T., Abdullah, S. S., Ahmed, R., and Giunchiglia, F. (2013). The history of temporal data visualization and a proposed event centric timeline visualization model. *International Journal of Computer Applications*, 70(27).

[Hasan et al., 2021] Hasan, K. T., Rahman, M. M., Ahmmed, M. M., Chowdhury, A. A., and Islam,

M. K. (2021). 4p model for dynamic prediction of covid-19: a statistical and machine learning approach. *Cognitive Computation*, pages 1–14.

[Nandi et al., 2012] Nandi, D., Hamilton, M., and Harland, J. (2012). Evaluating the quality of interaction in asynchronous discussion forums in fully online courses. *Distance education*, 33(1):5– 30.

[Nandi et al., 2011] Nandi, D., Hamilton, M., Harland, J., and Warburton, G. (2011). How active are students in online discussion forums? In *Proceedings of the Thirteenth Australasian Computing Education Conference-Volume 114*, pages 125–134.

**Appendix A**

# Appendix

Write your appendix here. Following two are examples.

## Name of Appendix-1

* 1. **Name of Appendix-2**

**Appendix B**

# Example of Citations

This text is only for Bibliography testing purposes.

Dr. Dip Nandi currently works as an Associate Professor and the Director of Faculty of Science and Technology in American International University-Bangladesh (AIUB). His research area includes: Software Engineering, E-Learning Technologies, Data Mining and Information systems and has produced several publications in these domains [[Nandi et al., 2012](#_bookmark20), [Nandi et al., 2011](#_bookmark21)].

Dr. Tabin Hasan primarily focuses in the research Domain of Human Computer Interaction. He is been a active researcher for more than a decade and produced many high quality journals [[Hasan et al.,](#_bookmark18) [2013](#_bookmark18)], conferences [[Hasan et al., 2021](#_bookmark19)] and book chapters.

**Appendix C**

# Example of Equations

The well known Pythagorean theorem *x*2 + *y*2 = *z*2 was proved to be invalid for other exponents. Meaning the next equation has no integer solutions:

*xn* + *yn* = *zn*

The ampersand character & determines where the equations align. Let’s check a more complex example:

*x* = *y w* = *z a* = *b* + *c*

2*x* = *y* 3*w* = 1 *z a* = *b*

*−*

2

*−*4 + 5*x* = 2 + *y w* + 2 = *−*1 + *w ab* = *cb*

The mass-energy equivalence is described by the famous equation

*E* = *mc*2

discovered in 1905 by Albert Einstein. In natural units (*c* = 1), the formula expresses the identity

Some random examples ...

*E* = *m* (C.1)

∞ 1 1

∑ *ns* = ∏ 1 *− p−s* (C.2)

*i*=1 *p*

∞

∑ 2*−n* = 1 (C.3)

*n*=1

*V µ*(*t, u, v, w*) *dt dudvdw* (C.4)

**Appendix D**

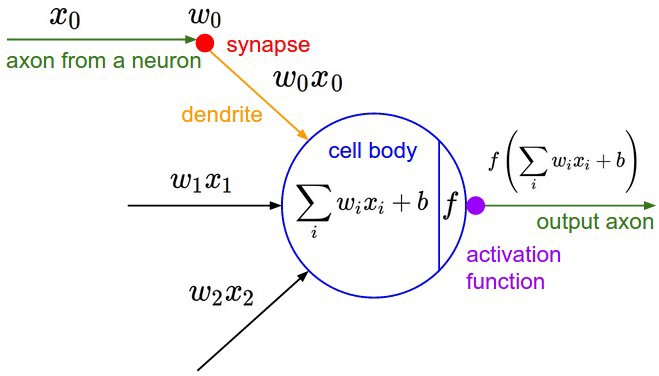
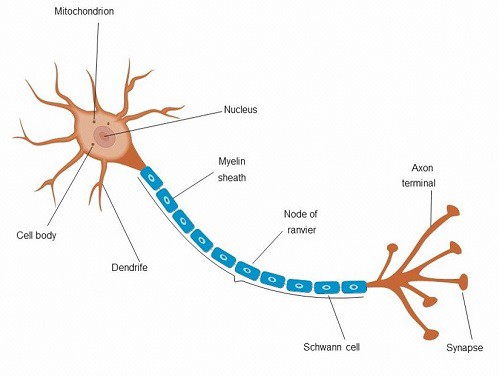
# Example of Figures



Figure D. 1 : American International University-Bangladesh (AIUB)

The Figure [D.1](#_bookmark28) represents beauty of the AIUB campus.

*APPENDIX D. EXAMPLE OF FIGURES*



1. Anatomy of a multipolar neuron (b) Architecture of a artificial neuron

Figure D. 2: Example of placing images side by side

**Appendix E**

# Example of Tables

Here is a really simple table [E.1](#_bookmark31).

Table E. 1: AIUB currently operates under four distinct Faculties x

|  |  |
| --- | --- |
| **Number** | **Name** |
| 1 | Faculty of Science and Technology (FST) |
| 2 | Faculty of Engineering (FE) |
| 3 | Faculty of Business Administration (FBA) |
| 4 | Faculty of Arts and Social Sciences (FASS) |

Here is another example of table row merged [E.2](#_bookmark32).

Table E. 2: Row span example

|  |  |  |
| --- | --- | --- |
| col1 | col2 | col3 |
| Multiple row | cell2  cell5 cell8 | cell3  cell6 cell9 |

Here is another example of controlling table width [E.3](#_bookmark33).

Table E. 3 : Test Table

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| One | Two | Three | Four | Five | Six | Seven | Eight | Nine | Ten | Eleven | Twelve | Thirteen | Fourteen |
| 1*.*111 | 2*.*222 | 3*.*333 | 4*.*444 | 5*.*555 | 6*.*666 | 7*.*777 | 8*.*888 | 9*.*999 | 0*.*000 | 1*.*111 | 2*.*222 | 3*.*333 | 4*.*444 |

**Appendix F**

# Example of algorithm procedure

**Algorithm 1:** Example code

**Input:** A graph *G* **Output:** A vertex of *G* **Data:** Testing set *x*

∞

**1** ∑

*i*=1

:= 0 // this is a comment

/\* Now this is an if...else conditional loop \*/

**2 if** *Condition 1* **then**

**3** Do something // this is another comment

**4 if** *sub-Condition* **then**

**5** Do a lot

**6 else if** *Condition 2* **then**

**7** Do Otherwise

/\* Now this is a for loop \*/

**8 for** *sequence* **do**

**9** loop instructions

**10 else**

**11** Do the rest

/\* Now this is a While loop \*/

**12 while** *Condition* **do**

**13** Do something

Example of writing algorithms is shown here [1](#_bookmark35).

**Appendix G**

# Example of Code

## G.1 Find the greatest number from a list of numbers in *Python*

a=[1,2,3,4,6,7,99,88,999]

max= 0

for i in a:

if i > max:

max=i print(max)

*End quote goes here.*