

BA723 Business Analytics Capstone

**Understanding Public Perception: Sentiment Analysis of ChatGPT-Related
Tweets**

Submitted by:

Justine Tinio (301241837)

Centennial College

Submitted to:

David Parent

Savita Seharawat

Table of Contents

Executive Summary	5
0.1 Executive Introduction.....	5
0.2. Executive Objective.....	5
0.3. Executive Model Description	5
0.4. Executive Recommendations	5
Introduction	6
1.0. Background	6
2.0. Problem Statement.....	7
3.0. Objectives & Measurement.....	7
4.0. Assumptions and Limitations	8
Data Sources.....	10
5.0. Dataset Introduction.....	10
6.0. Exclusions	11
6.1. Initial Data Cleansing or Preparation	12
7.0. Data Dictionary	15
Data Exploration	16
8.0. Data Exploration Techniques	16
9.0. Data Cleansing	16
10.0. Summary.....	24
Data Preparation and Feature Engineering.....	25
11.0. Data Preparation Needs	25
11.1. SMOTE	25
12.0. Feature Engineering.....	26
Model Exploration	26
13.0. Modeling Approach/Introduction.....	26
14.0. Model Technique #1 - Linear Support Vector Classifier (SVC)	27
15.0. Model Technique #2 – K-Nearest Neighbors (KNN) classifier	27
16.0. Model Technique #3 – Random Forest	28
17.0. Model Technique #4 – Multinomial Naïve Bayes	28
18.0. Model Technique #5 – Logistic Regression	28
19.0. Model Technique #6 – Logistic Regression	29
20.0. Model Comparison	29

<i>Model Recommendation</i>	32
21.0 Model Selection	32
22.0 Model Theory.....	32
23.1 Model Assumptions and Limitations	33
24.0 Model Sensitivity to Key Drivers.....	33
<i>Conclusion and Recommendations</i>	33
25.0. Impacts on Business Problem (Scope of the recommended model).....	33
26.0. Recommended Next Steps	34
<i>References</i>	35
27.0 References.....	35

Figure 1 Updated Twitter API pricing.....	9
Figure 2 Dataset information.....	10
Figure 3 Project Methodology	11
Figure 4 Language used for the tweets	12
Figure 5 Number of ChatGPT-related tweets per hour	13
Figure 6 Count of tweets by Verified and Unverified accounts.....	14
Figure 7 Average word count of tweets per hour	15
Figure 8 Removing user mention in tweets.....	16
Figure 9 Removing hashtags in the tweets.....	17
Figure 10 Removing URLs from the tweets	17
Figure 11 Removing emojis from the tweets.....	17
Figure 12 Removing special characters from the tweets	17
Figure 13 Removing punctuations from the tweets.....	18
Figure 14 Removing stopwords from the tweets	18
Figure 15 Removing numbers from the tweets.....	18
Figure 16 Removing stopwords and converting tweets to lowercase.	19
Figure 17 Tweets generating positive comp_score	20
Figure 18 Tweet generating neutral comp_score.....	21
Figure 19 Most common neutral words/tokens.....	21
Figure 20 Tweet generating negative comp_score	21
Figure 21 Most common negative words/tokens	22
Figure 22 Positive Word Cloud	23
Figure 23 Neutral Word Cloud	23
Figure 24 Negative Word Cloud.....	24
Figure 25 Percentage of Sentiment Levels in Tweets.....	24
Figure 26 Distribution of the target variable before SMOTE.....	25
Figure 27 Distribution of the target variable after SMOTE.....	26
Figure 28 Accuracy Scores	29
Figure 29 AUROC curve for the models	30
Figure 30 Confusion Matrix for Random Forest	31
Figure 31 Classification Summary of Random Forest.....	32

Executive Summary

0.1 Executive Introduction

In November 2022, OpenAI unveiled ChatGPT, a groundbreaking chatbot from the Generative Pre-Trained Transformers lineage, aiming to reshape human-machine interactions. ChatGPT's versatility is evident in its applications, from content creation to entertainment, and it has become a hot topic of discussion across various platforms. The platform X, known as Twitter, serves as a rich sentiment analysis resource, reflecting the evolution of sentiment research methodologies over the years. As businesses integrate AI linguistic models like ChatGPT, they grapple with challenges, including ethical considerations. Amidst a competitive AI landscape, OpenAI stands out, but rivals like Google and Microsoft offer formidable alternatives, highlighting the omnipresence of AI in today's digital age.

0.2. Executive Objective

Sentiment analysis of ChatGPT-related tweets provides businesses with insights that can elevate profit margins by understanding user emotions and preferences. By automating this analysis, companies not only enhance efficiency but also ensure timely adaptations to user needs, reinforcing their commitment to user satisfaction. However, neglecting this crucial process may jeopardize brand reputation and financial stability. This project aspires to refine ChatGPT's interactions by leveraging sentiment analysis tools, with the ultimate goal of optimizing user engagement. Key deliverables include a robust feedback system, user-specific sentiment profiles, and a monitoring mechanism to continually adapt to evolving user sentiments.

0.3. Executive Model Description

The Random Forest model, characterized by ensemble learning and multiple decision trees, boasts an accuracy of 81% in sentiment analysis, especially excelling in recognizing positive sentiments with an F1 score of 0.83. However, its recall rate for negative sentiments suggests room for improvement. This model, while adept at handling both categorical and numerical data, occasionally struggles with nuanced negative sentiments, possibly due to linguistic intricacies or dataset imbalances. Its ensemble nature could lead to increased computational demands and extended training durations. For holistic sentiment analysis, refining the model's sensitivity to negative nuances and ensuring balanced datasets is paramount.

0.4. Executive Recommendations

The proposed model allows businesses to deeply understand customer feelings, enhancing product and message alignment. By utilizing its accuracy, companies can make informed decisions and promptly address negative feedback. This study advises ChatGPT to foster direct user engagement through a feedback system and offer comprehensive tutorials. The positive feedback trend suggests users value ChatGPT, and community-centric events can further strengthen this bond. The model's focus on adverse sentiments indicates areas for future improvement and refinement.

Introduction

1.0. Background

ChatGPT's Emergence: Redefining Human-Machine Interactions

On November 30, 2022, a transformative moment in artificial intelligence was observed when OpenAI introduced ChatGPT to the world. Operating as a part of the Generative Pre-Trained Transformers family, this AI-powered chatbot aimed to redefine the realm of human-machine dialogues. From its launch, ChatGPT found its footing in diverse arenas, from being a virtual companion to a gaming ally. Its capabilities to mimic human conversation made waves in fields like content generation, linguistic education, investigative studies, and entertainment, to name a few. Beyond its professional applications, it became a focal point of vibrant debates and curiosities across social platforms, highlighting its universal allure.

The emergence of ChatGPT has opened avenues for sentiment exploration. By probing into user emotions, companies, especially those akin to OpenAI, can access a wealth of consumer insights. Such knowledge allows them to refine user interactions, ensure consistent quality, evaluate model efficacy, and track shifting user emotions. When harnessed effectively, these revelations can elevate the user engagement experience, facilitating personalized exchanges, shaping the content direction, and driving informed decision-making. The potential scope of this analytical pursuit spans diverse industries, including but not limited to e-commerce, gaming, academia, and entertainment.

X: The Digital Heartbeat

X, previously identified as Twitter, due to its extensive user engagement, has become an essential medium for individuals to voice their viewpoints on myriad subjects. Its stature as a sentiment goldmine has been recognized, leading to numerous sentiment-focused research undertakings (Li et al., 2013). Traditional sentiment probes have matured into more intricate sentiment and subjective scrutiny (Pang & Lee, 2004). Classic methods, whether algorithmic or lexicon-driven, have metamorphosed into combined approaches, yielding a deeper understanding (Serrano-Guerrero et al., 2015). This progression is further highlighted by the shift from basic text analysis to sophisticated symbol and attribute detection (Liu, 2012).

The business ecosystem has rapidly adopted the capabilities of AI linguistic models like ChatGPT. Advances in AI, especially in natural language interpretation, have empowered these models to both understand and replicate human-like text. However, their journey isn't without hurdles, especially the ethical dilemmas tied to their expansion. Discussions surrounding bias mitigation, ethical AI deployment, and the responsible utilization of these models are paramount. Adapting these models to cater to specific tasks or fields is crucial for businesses to ensure alignment with their distinct needs. The smooth amalgamation of AI linguistic models into tools ranging from interactive chatbots to content generators has enriched user interfaces and optimized operational processes.

By the close of 2021, OpenAI was acclaimed as a trailblazer in AI exploration and creation. As the mastermind behind ChatGPT, OpenAI is lauded for its dedication to developing AI tools that prioritize safety and utility. While their strides with innovations like

GPT-3 are commendable, the AI arena is bustling with competitors. Tech giants, including Google, Microsoft, IBM, and Amazon, have presented their solutions, specifically Dialogflow, Azure Cognitive Tools, Watson Companion, and Amazon Lex, in that order (Pang & Lee, 2008; Poria et al., 2014; Taboada et al., 2011; Liu, 2012). These platforms offer developers the toolkit to design conversational platforms, underscoring AI's omnipotent presence in the contemporary digital era.

2.0. Problem Statement

The primary objective of this project is to answer the question, "How can sentiment analysis algorithms effectively process and interpret the nuanced emotions expressed in short-form tweets to improve ChatGPT's response accuracy and enhance the user experience?"

3.0. Objectives & Measurement

Analyzing sentiments from ChatGPT-related tweets presents valuable opportunities for enterprises. By exploring users' feelings, inclinations, and contentment, firms can access essential data that can boost their profit margins. Identifying positive feedback helps companies grasp and maximize the attributes most appreciated by their clientele. This insight also highlights areas ripe for improvement, guiding developers in rectifying user issues and enhancing the AI model's performance. Such consistent optimization promotes superior user interactions, heightens customer contentment, and encourages word-of-mouth recommendations, all contributing to increased profits.

Moreover, sentiment analysis introduces cost and time savings. Automating the assessment of tweets about ChatGPT eliminates manual oversight, helping companies save essential resources and promptly react to user insights. Leveraging AI for sentiment evaluation, firms can rapidly spot tendencies, recurring patterns, and user obstacles, setting the stage for nimble decision processes and prompt adaptations to consumer needs. Acting on sentiment information is crucial for aligning with company objectives and essential performance metrics. By actively responding to feedback, companies emphasize their dedication to user experience enhancement. This forward-thinking strategy not only strengthens the AI tool's output but also nurtures user trust and contentment. Clients appreciate the responsiveness and effective problem resolution, resulting in an AI tool better aligned with their requirements and desires.

Overlooking sentiment analysis could result in user discontent, revenue losses, and a compromised brand reputation. Prioritizing it is in sync with customer-centric objectives, fortifying brand perception and encouraging sustained expansion. Nonetheless, the outcomes of sentiment evaluation, particularly regarding ChatGPT tweets, may vary based on the sector and scenario. Comprehensive sentiment assessment employs a blend of methods, from vocabulary-driven to computational learning, and can deliver instantaneous feedback on public mood. Employing visual tools and consolidated data can further sharpen these findings, aiding in informed decision-making.

Analytical Intent

This project's central objective is to understand how sentiment evaluation tools can proficiently interpret and convey the subtle feelings present in brief tweets, all aiming to enhance ChatGPT's interaction precision and deepen user engagement.

The anticipated outputs of our endeavor include not just the analytical findings but also a suggested scheme or outline for a preliminary program focused on refining ChatGPT's interaction prowess and user engagement quality. This holistic strategy incorporates multiple recommendations. Key among them is the introduction of a cyclical feedback system to continuously educate and fine-tune ChatGPT, empowering it to proficiently understand and convey nuanced tweet emotions. Crafting sentiment profiles centered on users, accounting for their unique emotional tendencies, can further tailor and elevate ChatGPT's engagements. To address specific user feedback or criticisms discovered through sentiment analysis, a mechanized response system can be introduced, consistently bolstering ChatGPT's performance. Additionally, a regular sentiment tracking system can be implemented, providing a lens into shifting user emotions and allowing early detection of potential hurdles or upcoming trends.

Purpose of Analytics

The core aim of this project revolves around deciphering how sentiment analysis algorithms can adeptly process and decode the intricate emotions encapsulated in concise tweets, with the end goal of augmenting ChatGPT's response accuracy and enriching user interactions.

The project's deliverables encompass not only the analytical outcomes but also a proposed framework or blueprint for a pilot initiative aimed at honing ChatGPT's response acumen and user experience. This comprehensive approach entails several recommendations. One of these is the adoption of an iterative feedback mechanism to perpetually train and refine ChatGPT, enabling it to adeptly discern and decode intricate tweet-based emotions. Creating user-centric sentiment profiles that factor in distinct sentiment predilections can further personalize and enhance ChatGPT's interactions. To address user-specific grievances or critiques unearthed via sentiment analysis, an automated mechanism can be instituted, fostering consistent enhancements in ChatGPT's efficacy. Furthermore, a routine sentiment analysis monitoring system can be established, offering insights into evolving user sentiments, and facilitating the early identification of potential challenges or emerging trends.

4.0. Assumptions and Limitations

With ChatGPT's rising stature, there's been a notable uptick in its mentions and discussions on various digital platforms. The increasing volume of online dialogues, where users recount their experiences and voice their opinions about the model, highlights the growing footprint of ChatGPT in public discourse.

Assumptions:

Representative Sample: The analyzed tweets are thought to offer a well-rounded snapshot of the overarching sentiment towards ChatGPT. It's posited that active X

platform contributors and their expressed feelings reflect the sentiments of the wider audience.

Authenticity: The assumption is that the tweets are genuine expressions, free from the influence of bots, marketing drives, or any manipulative interferences.

Clear Intent: Primarily, tweets concerning ChatGPT address the model's features and performance, rather than unrelated external factors.

Sentiment analysis is deemed a reliable tool to gauge the public's perception of ChatGPT through tweets, a common avenue for sharing opinions. The underlying confidence in this approach is rooted in the notion that tweets genuinely mirror user interactions, experiences, and viewpoints regarding ChatGPT. Advanced sentiment analysis techniques, powered by cutting-edge natural language processing tools, are trusted to decode the nuanced emotions within the concise boundaries of tweets. Such belief is bolstered by the vast training datasets these tools have been exposed to, enabling them to recognize varied emotional undertones.

Limitations:

Character Limit: The inherent brevity of X's platform might result in sentiments being condensed, possibly missing deeper nuances.

Limited Context: The constraints in tweet extraction mean some context could be lost, potentially leading to misinterpretation of sentiments.

Find the right access for you			
Free	Basic	Pro	Enterprise
For write-only use cases and testing the Twitter API	For hobbyists or prototypes	For startups scaling their business	For businesses and scaled commercial projects
<ul style="list-style-type: none">• Rate limited access to v2 tweet posting and media upload endpoints• 1,500 Tweets per month - posting limit at the app level• 1 app ID• Login with Twitter• Free	<ul style="list-style-type: none">• Rate limited access to suite of v2 endpoints• 3,000 Tweets per month - posting limit at the user level• 50,000 Tweets per month - posting limit at the app level• 10,000 Tweets per month - read-limit rate cap• 2 app IDs• Login with Twitter• \$100 per month	<ul style="list-style-type: none">• Rate-limited access to suite of v2 endpoints, including search and filtered stream• 1,000,000 Tweets per month - GET at the app level• 300,000 Tweets per month - posting limit at the app level• 3 app IDs• Login with Twitter• \$5,000 per month	<ul style="list-style-type: none">• Commercial-level access that meets your and your customer's specific needs• Managed services by a dedicated account team• Complete streams: replay, engagement metrics, backfill, and more features• Monthly subscription tiers
Get started	Subscribe now	Subscribe now	Apply now

Figure 1 Updated Twitter API pricing

The recent pricing adjustments to the Twitter API, as depicted in Figure 1, further complicate the scenario. Due to shifts spearheaded by Elon Musk, users now face a cap of 1,500 tweet extractions monthly. Musk's active role on Twitter culminated in altered access conditions, transitioning from a free model to a \$100 monthly charge under the

hobby plan. This shift, a consequence of Musk's increased visibility and subsequent platform scrutiny, limits real-time data extraction and historical tweet access. Such constraints hinder in-depth sentiment analysis, especially on trending topics like Musk. In his statement, Musk emphasized that these measures were introduced to curb excessive data mining and potential platform misuse.

Data Sources

5.0. Dataset Introduction

The "ChatGPT 1000 Daily Tweets" dataset, sourced from Kaggle, encapsulates a series of tweets generated over a specific period by ChatGPT, OpenAI's expansive language model. It offers a detailed compilation of tweets concerning ChatGPT, encompassing 20 distinctive columns, each shedding light on the tweet and its author. Boasting a total of 41,003 entries, the tweets cover dates from April 3, 2023, through May 10, 2023. This broad time span provides an in-depth view of public sentiments and interactions about ChatGPT on Twitter. Each dataset record corresponds to a distinct tweet, detailing crucial aspects such as the tweet's body, timestamp, user information, and engagement metrics. Vital columns feature the tweet's unique ID, timestamp, textual content, the language used, specific user details (like handle, geographic information, and bio), and engagement figures (counts of retweets, likes, etc.). This dataset stands as a crucial tool for scrutinizing public feelings and interaction trends associated with ChatGPT on X/Twitter.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 41003 entries, 0 to 41002
Data columns (total 20 columns):
#   Column                                Non-Null Count  Dtype
---  -
0   tweet_id                             41003 non-null  object
1   tweet_created                         41003 non-null  object
2   tweet_extracted                       41003 non-null  object
3   text                                 41003 non-null  object
4   lang                                 41003 non-null  object
5   user_id                              41003 non-null  object
6   user_name                            40998 non-null  object
7   user_username                        41003 non-null  object
8   user_location                        24345 non-null  object
9   user_description                     34723 non-null  object
10  user_created                         40998 non-null  object
11  user_followers_count                 40995 non-null  float64
12  user_following_count                 40995 non-null  float64
13  user_tweet_count                     40995 non-null  float64
14  user_verified                        40995 non-null  object
15  source                               0 non-null      float64
16  retweet_count                        40995 non-null  float64
17  like_count                           40995 non-null  float64
18  reply_count                          40995 non-null  float64
19  impression_count                     40995 non-null  float64
dtypes: float64(8), object(12)
memory usage: 6.3+ MB
```

Figure 2 Dataset information

In this research, we will adopt a six-step project methodology encompassing data collection, ChatGPT tweet extraction, text preprocessing, determining polarity, data model development, and culminating in model assessment and final insights.

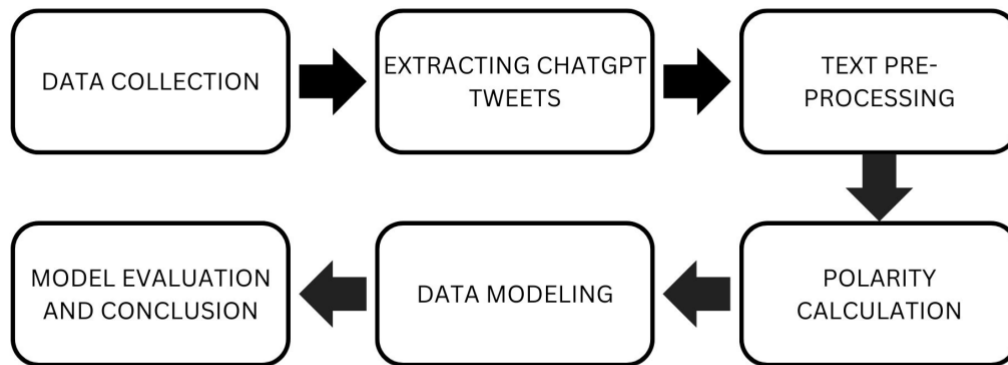


Figure 3 Project Methodology

6.0. Exclusions

The dataset encompasses tweets in various languages, as indicated by the 'lang' attribute. We have chosen to concentrate our analysis predominantly on one language, specifically English (denoted as 'en'). By excluding non-English tweets, we aim to maintain a consistent linguistic framework for our analysis. This decision was made to prevent language biases and ensure a more precise sentiment evaluation without the complexities of cross-linguistic nuances. Subsequently, relevant columns intended for deeper analysis were isolated into a new data frame, focusing on just English tweets, narrowing the dataset to 19,711 rows from 41,003.

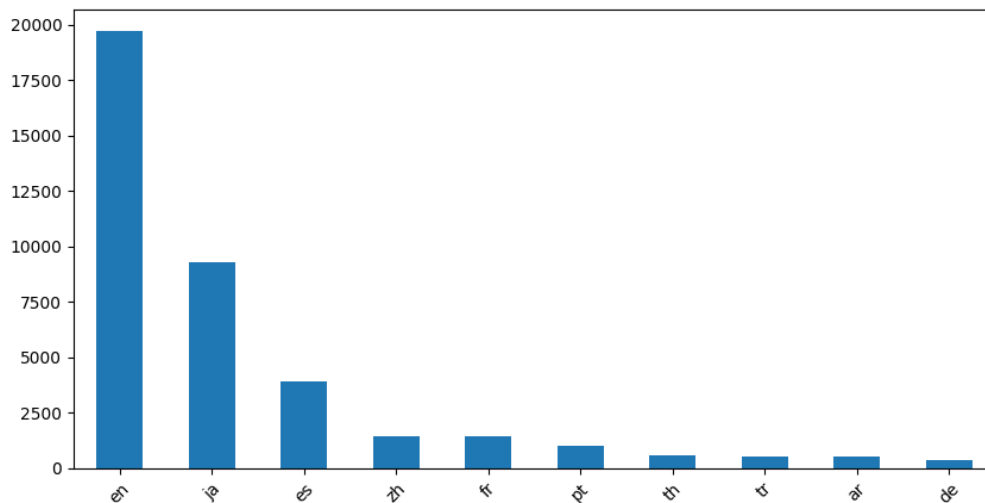


Figure 4 Language used for the tweets

In the 'lang' (Language) column, the codes represent the following languages: "en" stands for English, "ja" corresponds to Japanese, "es" denotes Espanol or Spanish, "zh" signifies Chinese, "fr" represents French, "pt" indicates Portuguese, "th" is used for Thai, "tr" is for Turkish, "ar" is the abbreviation for Arabic, and "de" designates Deutsch.

Moreover, the "tweet_id" serves as a distinct marker for each tweet. However, since it doesn't play a direct role in sentiment analysis, it's considered expendable. Similarly, identifiers related to the user, such as "user_id", "user_name", and "user_username", don't hold immediate relevance for sentiment analysis and were omitted. The attribute "user_description" provides a brief overview of the user, but it doesn't influence the sentiment of the tweet, making it suitable for exclusion. Given the dataset's lack of diverse platform or device data, the "source" attribute isn't apt for sentiment evaluation. Additionally, attributes like "user_created" and "tweet_extracted", despite being present, don't have a direct bearing on sentiment analysis and can be sidelined. Conversely, the "user_location" could be retained, especially if there's an inclination to explore sentiment analysis from a geographical or regional perspective but is not part of this project's scope, hence was dropped.

6.1. Initial Data Cleansing or Preparation

For effective sentiment analysis in Python, particularly when addressing text data from tweets, reviews, or comments, consider the following streamlined data preparation and cleansing steps:

Managing Duplicates:

One of the foremost steps is the elimination of redundant tweets. This is crucial as repeated sentiments can distort the analytical outcomes. By ensuring the uniqueness of each tweet, we guarantee that every sentiment is accurately represented without repetitive influence.

Addressing Absent or Incomplete Data:

Data integrity often faces challenges from missing or incomplete entries. Such gaps, especially when they appear in the 'text' column of a dataset, can undermine the effectiveness of sentiment analysis. Therefore, it's essential to identify and exclude rows that lack substantial information. Their presence would not only fail to add value but might also detract from the overall quality of the insights derived.

Handling Categorical Data:

Sentiment labels often come in categorical forms, such as 'positive', 'negative', or 'neutral'. For many analytical procedures, a numerical representation of these categories can be more conducive. Hence, it's recommended to transform these categorical labels into a numeric format, making the data more amenable to various computational techniques.

By carefully making these changes and updates to the data, we make it cleaner and more focused. This helps the sentiment analysis to better understand real and relevant opinions from tweets. This improved method makes our results more trustworthy and useful, helping us make better decisions later on.

An initial examination of the data reveals a notable trend: a significant majority of tweets related to ChatGPT tend to be posted around 5 pm in the afternoon. This surge in activity during the late afternoon might be attributed to several factors. It's possible that users are taking breaks from work or school and using that time to engage with ChatGPT. Alternatively, this could be the time when discussions and online interactions generally peak, leading to increased mentions of ChatGPT. It's also worth considering time zones and global user distribution, as peak times in one region might coincide with 5 pm in another. Understanding the reasons behind this trend could provide valuable insights into user behavior and the optimal times for engagement or updates.

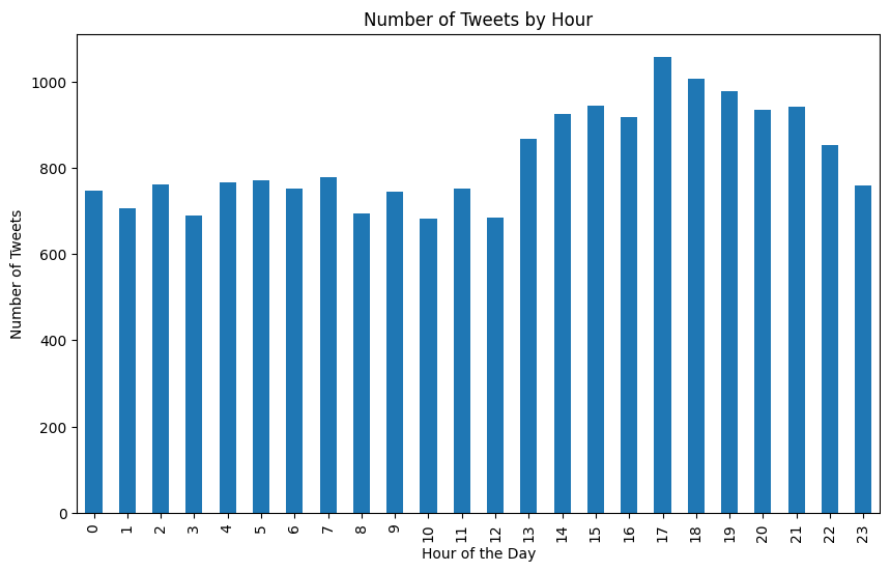


Figure 5 Number of ChatGPT-related tweets per hour

When examining the distinction between verified and unverified users in our dataset, a significant disparity emerges. Out of the 41,003 tweets related to ChatGPT, a substantial 38,398 tweets originate from unverified accounts, leaving only 2,497 tweets from verified users. This predominance of unverified user activity could be attributed to several factors. First, most social media users, particularly on platforms like X/Twitter, do not possess verified statuses, thus naturally leading to higher tweet volumes from them. Moreover, verified users often represent celebrities, influencers, or industry professionals, who might not engage as frequently with topics like ChatGPT as the general public. Additionally, the nature of verified accounts often implies a level of public scrutiny, leading these users to be more selective in their tweeting patterns. Understanding this imbalance can offer insights into the broader audience engaging with ChatGPT and help tailor strategies to both verified and unverified user demographics.

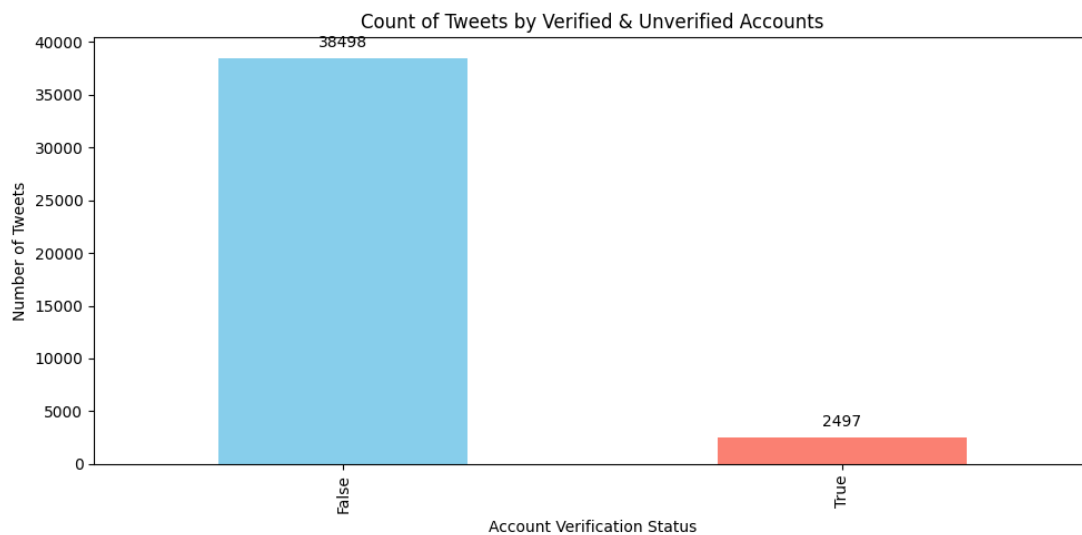


Figure 6 Count of tweets by Verified and Unverified accounts

Upon delving deeper into our dataset, it becomes evident that the average word count peaks at 22.88 during the 11th hour of the day. This trend may be attributed to several factors. For instance, it could be the result of users being more active and engaged during this period, possibly due to breaks or free time in their daily routines. Another possibility is that certain events or discussions related to ChatGPT might be scheduled around this time, leading to an influx of tweets. Understanding the reason behind this surge can provide valuable insights into user behavior and preferences, allowing for more targeted and timely interactions in the future.

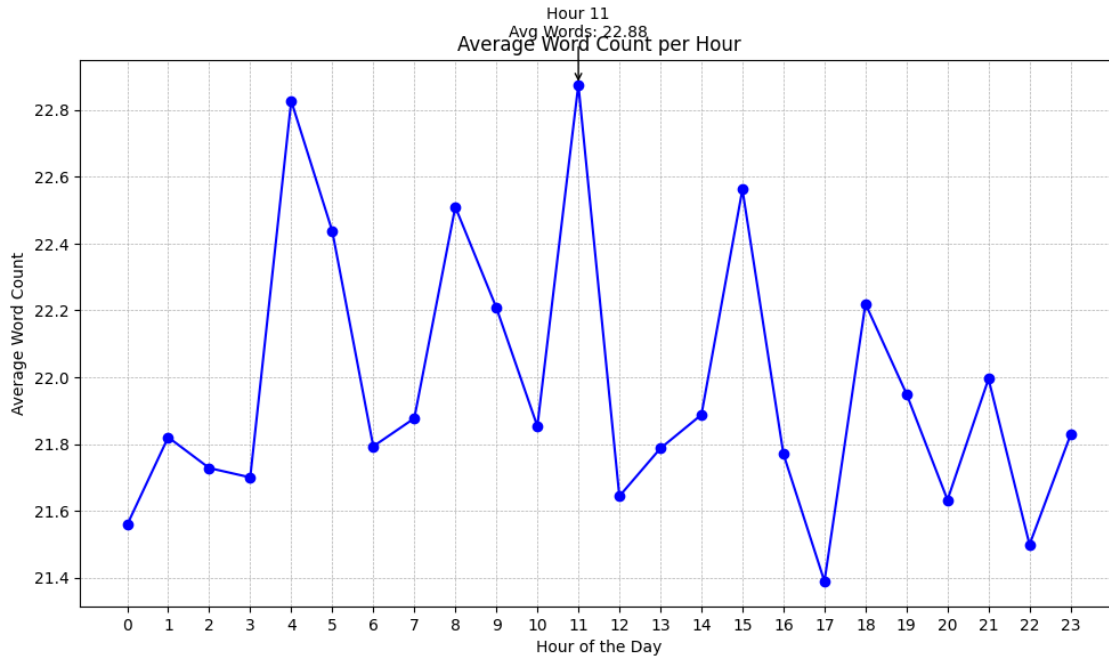


Figure 7 Average word count of tweets per hour

7.0. Data Dictionary

Attribute	Datatype	Description
tweet_id	object	Unique identifier for a tweet.
tweet_created	datetime	Date and time when the tweet was created.
tweet_extracted	datetime	Date and time when the tweet was extracted or retrieved.
text	object	The actual text content of the tweet.
lang	object	Language in which the tweet is written (ex. 'en' as English tweets).
user_id	object	Unique identifier for the user who posted the tweet.
user_name	object	Name of the user who posted the tweet.
user_username	object	Username or handle of the user who posted the tweet.
user_location	object	Location mentioned in the user's profile.
user_description	object	Description or bio provided by the user in their profile.
user_created	object	Date and time when the user's account was created.
user_followers_count	int	Number of followers the user has.
user_following_count	int	Number of accounts the user is following.
user_tweet_count	int	Total number of tweets posted by the user.
user_verified	object	Indicates whether the user's account is verified (True/False).

source	null	The source or platform from which the tweet was posted.
retweet_count	int	Number of times the tweet has been retweeted.
like_count	int	Number of times the tweet has been liked.
reply_count	int	Number of replies received by the tweet.
impression_count	int	Number of times the tweet has been seen or displayed.

Data Exploration

8.0. Data Exploration Techniques

Exploratory data analysis (EDA) serves as an initial step in the data science process, aiding in understanding underlying data structures, spotting outliers, testing hypotheses, and visualizing data through graphics. Once we have refined our data through the initial cleansing phase, we'll employ the Neattext Library, a specialized tool designed to purify our dataset further. Neattext is a robust Python library designed for text data pre-processing and cleaning. Tailored to handle challenges in text analysis, it can efficiently process multi-lingual data, removing noise and unwanted elements. Its user-friendly design, complemented by comprehensive documentation, makes it a preferred choice for both novices and experts in natural language processing.

9.0. Data Cleansing

Text Pre-processing using the Neattext Library

By distilling the dataset, we eliminate unnecessary digital noise or clutter. This step ensures that our analysis is based on pure, relevant data without any distracting elements, which makes our findings more reliable and accurate. By leveraging the capabilities of Neattext, we can efficiently undertake the following steps to refine our data:

Remove User Mentions: By eradicating mentions or user tags from our dataset, we ensure that our analysis remains focused on the main content of the text. This helps us understand the general sentiment without being influenced by specific user interactions or biases.

	text	clean_tweet
2	RT @DarrellLerner ChatGPT Plugins are the fas...	RT ChatGPT Plugins are the fastest way to ge...
40998	RT @YGPT_official 🚀 YGPT LAUNCH 🚀\n\nTime to ...	RT 🚀 YGPT LAUNCH 🚀\n\nTime to show you what ...
41000	RT @chatgpt_issac Lets see which community is...	RT Lets see which community is bigger and I'...

Figure 8 Removing user mention in tweets

Strip Away Hashtags: Removing hashtags is crucial to ensure that our analysis stays rooted in the primary context of the message. Hashtags often represent trending topics or groupings, and while they can provide context, they might also divert our analysis from the core sentiment of the text.

	text	clean_tweet
4	🔥 Hey Guys, #ZenithSwap has launched at just \$...	🔥 Hey Guys, has launched at just \$ 55,000 USD...
5	RT @sinsonetwork: Now! Join #SINSO DataLand^Ch...	RT @sinsonetwork: Now! Join DataLand^ChatGPT...

Figure 9 Removing hashtags in the tweets

Eliminate Web Links: By weeding out embedded URLs, we preserve the genuine textual essence of our dataset. Links can often lead to external content, which may or may not align with the sentiment of the original text. By excluding them, we maintain the integrity of our data.

index	text	clean_tweet
3	Get an intelligent chatbot for your website in minutes with Chatbase AI. Train ChatGPT on your data and let it answer any question your users have. Simply upload a document or link and add the chat widget - it's that easy! Make Money using AI: https://t.co/yLHEqn4w9T https://t.co/ba54JvoRsM	Get an intelligent chatbot for your website in minutes with Chatbase AI. Train ChatGPT on your data and let it answer any question your users have. Simply upload a document or link and add the chat widget - it's that easy! Make Money using AI:

Figure 10 Removing URLs from the tweets

Focus on Text, Not Emojis: Emojis, while expressive, can be subjective. By sifting them out, we pave the way for an analysis that's based purely on textual content, ensuring consistency and clarity in interpretation.

index	text	clean_tweet
4	🔥 Hey Guys, #ZenithSwap has launched at just \$ 55,000 USD Marketcap. The ChatGPT of DEX - Reimagining DeFi with AI-Powered Yield Farming. Now at 4X. Lot of up potential at such low marketcap. 🌟 👉 \$ARB \$ZSP #Arbitrum https://t.co/9VWYtYzAJD	Hey Guys has launched at just \$ 55000 USD Marketcap The ChatGPT of DEX Reimagining DeFi with AIPowered Yield Farming Now at 4X Lot of up potential at such low marketcap \$ARB \$ZSP
5	RT @sinsonetwork: Now! Join #SINSO DataLand^ChatGPT #Airdrop! 📅 23-4.6 📌 Tasks ①Log in to<https://t.co/Hlwqa7HG40> ②Try SINSO #ChatGPT& twe...	RT Now Join DataLand^ChatGPT 32346 Tasks Log in to! Try SINSO twe

Figure 11 Removing emojis from the tweets

Cleanse Special Characters and Punctuations: Removing special symbols, characters, and punctuations ensures that our analysis is concentrated on meaningful content. This step helps to minimize distractions and potential misinterpretations arising from symbols.



index	text	clean_tweet
31216	RT @crispinhunt: Bravo @eu_comission : "Companies deploying generative AI tools, such as ChatGPT, will have to disclose any  material us...	RT Bravo Companies deploying generative AI tools such as ChatGPT will have to disclose any material us
15095	RT @BrianRoemmele: This is another in a series of SuperPrompts  for ChatGPT (and other AI) from the https://t.co/MmjROXv0yF archive. In t...	RT This is another in a series of SuperPrompts for ChatGPT and other AI from the archive In t

Figure 12 Removing special characters from the tweets

index	text	clean_tweet
1848	RT @McaleerStephen: Forget plugins. ChatGPT can solve general computer tasks using a keyboard and mouse!! The trick? Recursively criticizi...	RT Forget plugins ChatGPT can solve general computer tasks using a keyboard and mouse The trick Recursively criticizi...
2236	@margal @RomanRussy I guess now we could just ask chatgpt to write a letter and then no one's the "orchestrator"!!	I guess now we could just ask chatgpt to write a letter and then no one's the "orchestrator"

Figure 13 Removing punctuations from the tweets

Standardize Language Use: By homogenizing contractions, we ensure a consistent language throughout our dataset. For example, turning contractions like "isn't" into "is not" ensures uniformity and makes the analysis process smoother.

index	text	clean_tweet
350	Most people in the world will look at generative AI and dismiss it, And i couldn't be happier that means I have a small headstart to learn more about it get into better roles and build better business. #AI #chatGPT	Most people in the world will look at generative AI and dismiss it And i could not be happier that means I have a small headstart to learn more about it get into better roles and build better business
220	@Daniel_Rubino Very weird. I wouldn't be surprised if Apple "invents" AI after the issues are worked out with Bing Chat, ChatGPT, and Bard though.	Very weird I would not be surprised if Apple invents AI after the issues are worked out with Bing Chat ChatGPT and Bard though

Figure 14 Removing stopwords from the tweets

Remove Numbers: Numbers, unless contextually relevant, don't contribute much to sentiment. By omitting them, we ensure that our sentiment analysis remains focused on the emotions and opinions conveyed through words.

47	RT @rowancheung: AI prompting is the best skill to learn right now. Companies are now paying up to \$335,000 year for Prompt Engineers. H...	RT AI prompting is the best skill to learn right now Companies are now paying up to year for Prompt Engineers H
95	🔥 Hey Guys, #ZenithSwap has launched at just \$55,000 USD Marketcap. The ChatGPT of DEX - Reimagining DeFi with AI-Powered Yield Farming. Now at 4X. Lot of up potential at such low marketcap. 🙌👉 \$ARB \$ZSP #Arbitrum https://t.co/eWLsdVy3Sm	Hey Guys has launched at just USD Marketcap The ChatGPT of DEX Reimagining DeFi with AIPowered Yield Farming Now at X Lot of up potential at such low marketcap ARB ZSP

Figure 15 Removing numbers from the tweets

Refine Textual Data: By purging common yet non-informative words (known as 'stopwords') and converting all text to lowercase, we ensure a consistent and streamlined dataset. This step helps us zero in on the most meaningful and impactful words, providing a clearer picture of the prevailing sentiment.

	text	clean_tweet
2	RT @DarrellLerner: ChatGPT Plugins are the fas...	rt chatgpt plugins are the fastest way to get ...
3	Get an intelligent chatbot for your website in...	get an intelligent chatbot for your website in...
4	🔥 Hey Guys, #ZenithSwap has launched at just \$...	hey guys has launched at just usd marketcap ...
5	RT @sinsonetwork: Now! Join #SINSO DataLand^Ch...	rt now join datalandchatgpt taskslog in toltr...
9	The plagiarism detector will introduce its #AI...	the plagiarism detector will introduce its det...
...
40995	RT @solanaturbo: Hey everyone, have you heard ...	rt hey everyone have you heard about the new m...
40996	Can ChatGPT write my SoHo House application? W...	can chatgpt write my soho house application wr...
40998	RT @YGPT_official: 🚀 YGPT LAUNCH 🚀\n\nTime to ...	rt ygpt launch time to show you what weve be...
41000	RT @chatgpt_issac: Lets see which community is...	rt lets see which community is bigger and im g...
41001	@simamaung ChatGPT	chatgpt

Figure 16 Removing stopwords and converting tweets to lowercase.

Polarity Score using VADER Lexicon

VADER (Valence Aware Dictionary and sEntiment Reasoner) is a pre-trained sentiment analysis tool tailored for social media content but is versatile for various texts. It excels in grasping the context, including intensifiers, negations, and emojis. The tool assigns a polarity score to a text, ranging from -1 (highly negative) to +1 (highly positive), offering insights into the text's emotional tone and aiding in gauging user sentiments effectively.

Tokenization:

Before diving into sentiment analysis, the text is broken down or "tokenized" into its individual components, typically words or phrases. This process ensures that each word can be analyzed in isolation, allowing for a more granular sentiment evaluation.

Lexicon Lookup:

At the heart of VADER is its lexicon—a predefined repository of words, each paired with a sentiment score. Once the text is tokenized, each token or word is matched against this lexicon. Each word's associated sentiment score is then fetched, providing an initial measure of the text's sentiment.

Compound Score Calculation:

After gathering individual sentiment scores for each word, VADER calculates an aggregate score. However, simply adding up these scores might lead to values that are too large or too small. To circumvent this, VADER uses a mathematical squashing function to normalize this total score, ensuring it remains within the -1 to +1 range. This normalized score is referred to as the 'compound' score, encapsulating the overall sentiment of the text.

Adjustments and Contextual Considerations:

What sets VADER apart is its ability to consider the broader context of a sentence or text. It doesn't just rely on individual word scores. VADER understands that the placement, combination, or juxtaposition of words can alter sentiment. For instance, intensifiers can amplify sentiment ("very good" is more positive than just "good"), while negations can flip sentiment ("not good" is negative). By accounting for these linguistic nuances, VADER ensures a more accurate and contextually relevant sentiment analysis.

In the process of sentiment analysis, the value of c plays a pivotal role in determining the emotional tone of the text:

When the compound score has a value greater than 0, the system categorizes it as 'pos'. This label signifies that the underlying sentiment of the text is positive, reflecting favorable or optimistic feelings or opinions.

	clean_tweet	sentiment	compound	comp_score
2	rt chatgpt plugins are the fastest way to get rich in ive created a stepbystep guide showing you how to earn	{'neg': 0.0, 'neu': 0.763, 'pos': 0.237, 'compound': 0.6808}	0.6808	pos
3	get an intelligent chatbot for your website in minutes with chatbase ai train chatgpt on your data and let it answer any question your users have simply upload a document or link and add the chat widget its that easymake money using ai	{'neg': 0.0, 'neu': 0.932, 'pos': 0.068, 'compound': 0.4588}	0.4588	pos

Figure 17 Tweets generating positive comp_score

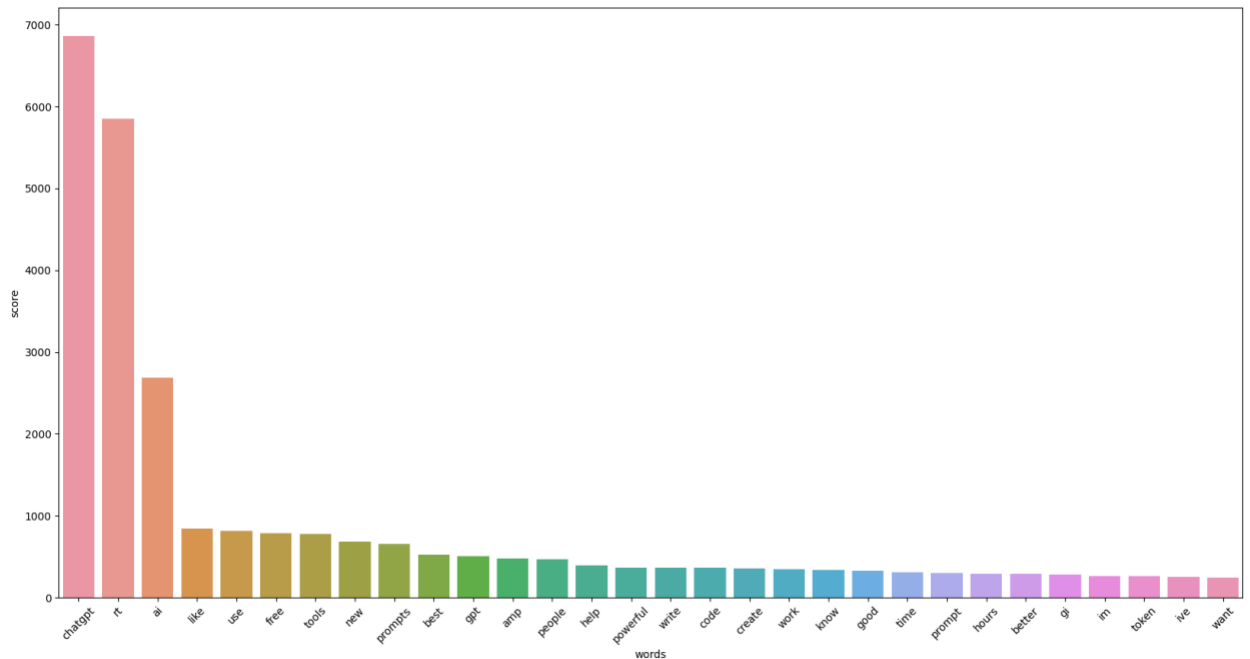


Figure 18 Most common positive words/tokens

If the compound score is exactly at 0, it is labeled as 'neu'. This means the text is neither tilted towards positivity nor negativity but strikes a balanced, neutral stance, often reflecting an unbiased or indifferent viewpoint.

index	clean_tweet	sentiment	compound	comp_score
64	rt the rise of the developer	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}	0.0	neu

Figure 18 Tweet generating neutral comp_score

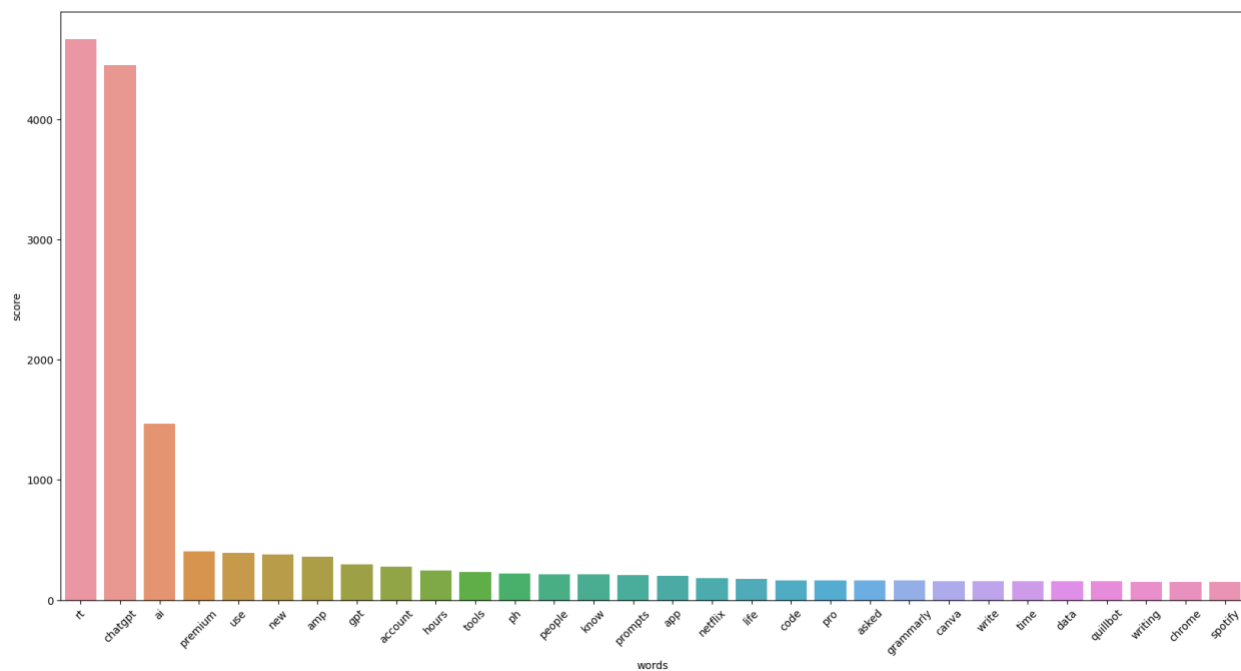


Figure 19 Most common neutral words/tokens

On the other hand, if the compound score has a value less than 0, the sentiment is classified as 'neg'. This categorization reveals that the text embodies negative emotions or opinions, indicating criticisms, displeasure, or pessimistic perspectives.

index	clean_tweet	sentiment	compound	comp_score
10	germany could follow in italys footsteps by blocking chatgpt over data security concerns	{'neg': 0.163, 'neu': 0.688, 'pos': 0.15, 'compound': -0.0516}	-0.0516	neg

Figure 20 Tweet generating negative comp_score

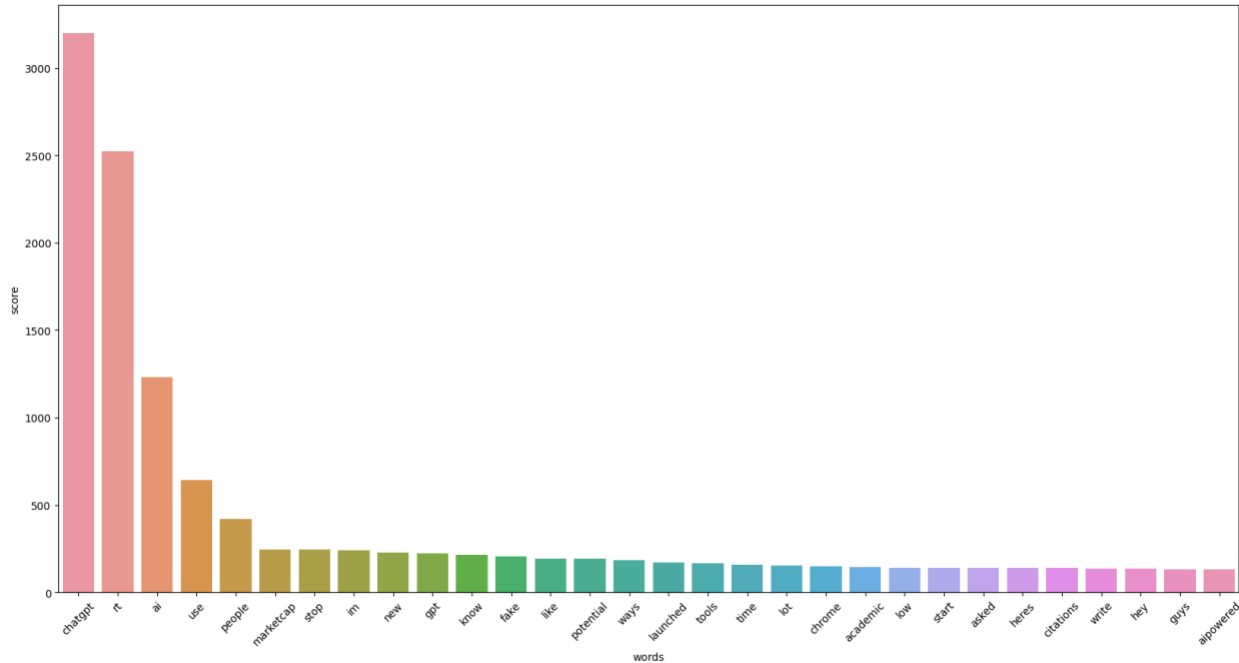


Figure 21 Most common negative words/tokens

This methodical classification based on the value of c ensures a systematic and clear delineation of sentiments, facilitating more precise analysis and interpretation of textual data.

Generating word clouds

Word clouds in Python provide a visual snapshot of text data, representing frequently used words in larger fonts. This visualization technique is crucial in sentiment analysis as it offers an immediate insight into dominant themes or sentiments within a dataset. By converting textual information into a visual format, word clouds make data more accessible and interpretable. They can highlight patterns, assist in refining data-cleaning processes, and differentiate terms associated with varying sentiments. In essence, word clouds act as a bridge, transforming raw text into a format that can be quickly understood and analyzed for deeper insights.

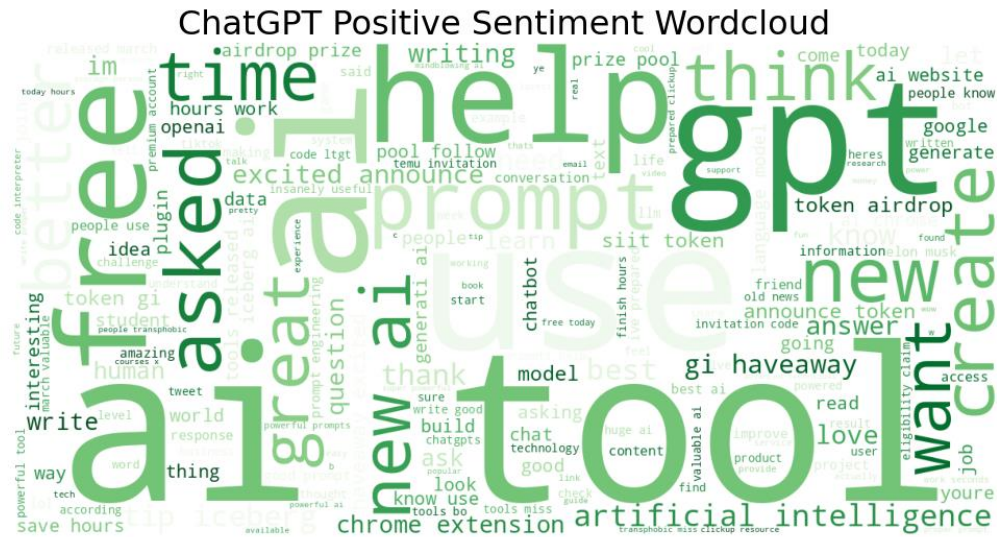


Figure 22 Positive Word Cloud

The positive words (the words that appeared the maximum number of times in the tweets with positive sentiments) as seen from the above graph are – ai, tool, gpt, help, and free.

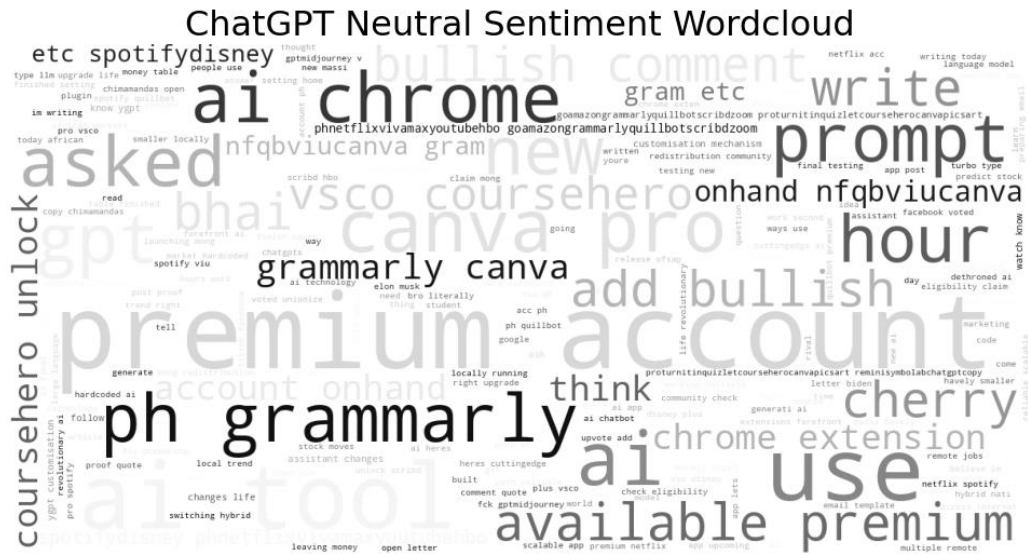
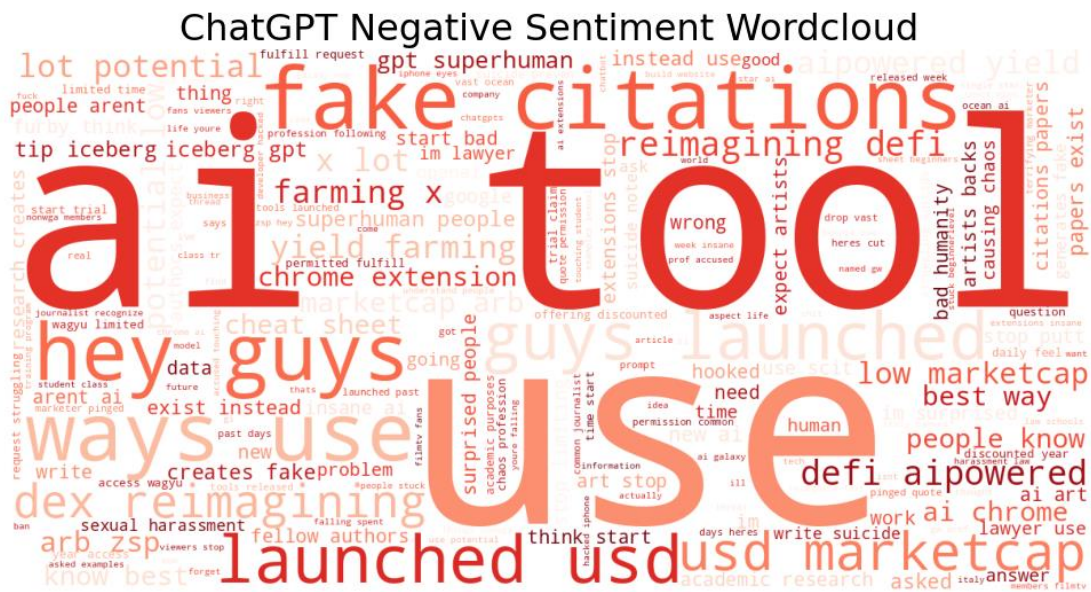


Figure 23 Neutral Word Cloud

The neural words (the words that appeared the maximum number of times in the tweets with neutral sentiments) as seen from the above graph are – premium, account, ph, grammarly, and prompt.



The negative words (the words that appeared the maximum number of times in the tweets with negative sentiments) as seen from the above graph are – ai, tool, use, fake, and citations.

10.0. Summary

Upon implementing the Neattext library to refine our dataset, and then applying the VADER Lexicon for sentiment analysis, we obtained a comprehensive breakdown of the tweet sentiments. The visualization offered by the word cloud further underscored the dominant sentiments present in the dataset.

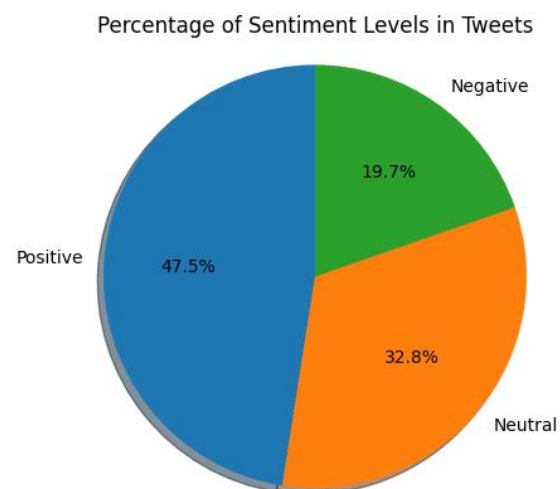


Figure 25Percentage of Sentiment Levels in Tweets

From the analysis, it became evident that positive tweets constituted a significant portion, making up 47.5% of the total tweets. Neutral sentiments trailed behind, accounting for 32.8% of the tweets. On the other hand, negative sentiments, while still present, were the least prominent, comprising 19.7% of the tweets. This distribution provides a clear understanding of the prevailing sentiment trends related to our subject of study.

Data Preparation and Feature Engineering

11.0. Data Preparation Needs

Upon leveraging resources such as Neattext and the VADER lexicon for initial data processing, we transitioned into the critical phase of data preparation. This phase is pivotal as it ascertains that our "clean tweets" are not only devoid of noise but are also structured optimally for modeling purposes. Additionally, during this phase, we discerned an imbalance within our dataset.

```
pos      7432
neu      5227
neg      3109
Name: comp_score, dtype: int64
```

Figure 26 Distribution of the target variable before SMOTE

Undersampling is a method that brings balance to skewed datasets by reducing the majority class while retaining all the minority class entries. Conversely, oversampling achieves dataset equilibrium by amplifying the minority class's size to match that of the majority class. Additionally, the SMOTE is a specialized form of oversampling that generates synthetic samples in the feature space, further enhancing the representation of the minority class. Gaining the best accuracy, the implementation of the SMOTE technique became indispensable, ensuring a balanced dataset for more precise modeling. This meticulous approach amplifies the efficacy of the subsequent train-test split and ensures that the resulting models are both robust and insightful.

11.1. SMOTE

The Synthetic Minority Over-sampling Technique, commonly known as SMOTE, is a pivotal tool when addressing imbalances in datasets. In many real-world scenarios, certain classes within a dataset may be under-represented, leading to biased machine-learning models that favor the majority class. SMOTE comes to the rescue by generating synthetic samples in the feature space. Rather than merely replicating minority instances, it interpolates between them, ensuring a more diverse and representative sample. By harmonizing the class distribution, SMOTE enhances the model's ability to generalize and make accurate predictions across all classes. In essence, for robust and unbiased machine learning outcomes, especially in classification tasks, employing SMOTE is often indispensable.

```
neu      7432
neg      7432
pos      7432
Name: comp_score, dtype: int64
```

Figure 27 Distribution of the target variable after SMOTE

12.0. Feature Engineering

The study employed two techniques: Count Vectorizer and Term Frequency-Inverse Document Frequency (TF-IDF). Count Vectorizer translates text into numerical representations, where each feature corresponds to the count of a certain word or phrase. These numerical patterns guide machine learning models in discerning and predicting sentiments such as positive, negative, or neutral based on word counts. Using the CountVectorizer tool from `sklearn.feature_extraction.text`, an instance was created, which then converted the text into a matrix of word counts. This step identified a collection of unique words from the given text, resulting in a discovery of 18,269 distinct words. In the resulting dataframe, each feature signifies the count of individual words in distinct tweets.

In contrast, TF-IDF quantifies text by attributing significance scores to words relative to their frequency in a document versus the entire collection. These significance-weighted patterns then feed into machine learning models, refining their capability to detect sentiment by spotlighting important contextual words. Utilizing `TfidfVectorizer` from `sklearn.feature_extraction.text`, an instance of TF-IDF was established, with a criterion to exclude terms present in fewer than 0.5% of the documents, yielding 458 unique terms. After this, the system was trained to calculate TF-IDF values, presenting the vectors. Though both TFIDF and CountVectorizer identify unique phrases, their representations diverge. TFIDF provides significance-adjusted counts, decreasing emphasis on common terms, whereas CountVectorizer presents raw frequencies.

The preference for TF-IDF over Count Vectorizer stems from its refined text interpretation methodology. A salient feature of TF-IDF is its adeptness at highlighting pivotal words in a document's sentiment or theme. By assigning weights based on a word's rarity throughout a dataset, TF-IDF accentuates terms pivotal to individual documents. In contrast, the Count Vectorizer might exaggerate the role of recurring words, but TF-IDF moderates these terms, acknowledging their limited contribution to a document's distinct sentiment. Moreover, TF-IDF adeptly curtails the impact of prevalent but non-indicative words.

Model Exploration

13.0. Modeling Approach/Introduction

In this section, we delve into the train-test split method, especially as it relates to sentiment analysis. Train-test split stands as a pivotal approach in machine learning, allowing us to partition our data into two separate groups. The bulk of this data, termed the "training set," acts as the bedrock for instructing and refining our sentiment analysis model. On the other hand, the "test set," which accounts for 20% of the data due to our 0.2 split, serves to gauge the effectiveness of our model in practical situations. This division ensures that

our model is equipped to discern and interpret sentiments genuinely, rather than just replicating prior patterns, thus sidestepping overfitting issues. The compound score is our target variable, while the clean tweets will serve as our input variable.

Furthermore, in our analytical journey, we've adopted the Synthetic Minority Over-sampling Technique (SMOTE) to rectify any imbalance in sentiment representation within our data. This method guarantees that our model is proficient in identifying a wide array of sentiments without showing undue preference to the prevalent sentiment class. The shared code demonstrates the integration of SMOTE, the data split (maintaining a random state for consistency), and the subsequent rebalancing of the training dataset to ensure well-distributed sentiment classes. Grasping these foundational steps is crucial to fully appreciate the ensuing analysis and the insights it offers.

14.0. Model Technique #1 - Linear Support Vector Classifier (SVC)

For the first model, we used the Linear Support Vector Classification (LinearSVC), a specialized variant of the Support Vector Machine (SVM) tailored for categorization tasks. SVMs operate by discerning the optimal boundary, or hyperplane, that distinctly classifies data entities in a given space. Specifically, LinearSVC addresses data that can be linearly separated.

We initiate the LinearSVC model, ensuring a consistent set of results across multiple runs by setting a random state of 2. This reproducibility ensures a degree of reliability in our findings. Post initialization, the model is trained on the SMOTE-balanced dataset, enhancing its capability to predict sentiments evenly across different classes. After training, we test the model's proficiency on a separate dataset, evaluating its predictions against actual outcomes to compute its accuracy in percentage terms.

To refine our model's outputs, we also used the CalibratedClassifierCV method. Though LinearSVC doesn't inherently produce probability-based outcomes, by integrating CalibratedClassifierCV, we can recalibrate our model to yield more interpretable probabilities. This recalibration uses a sigmoid function and leverages the foundational LinearSVC model, providing a more nuanced understanding of our model's predictions.

15.0. Model Technique #2 – K-Nearest Neighbors (KNN) classifier

For our second model, we employed the K-Nearest Neighbors (KNN) classifier, a staple in the machine learning toolbox. The essence of KNN lies in its simplicity: it ascertains the sentiment of a given text by scrutinizing the sentiments of its neighboring data points from the training set. To gauge the 'closeness' between data points, it leverages distance metrics within vector spaces, usually sculpted by techniques like TF-IDF or Count Vectorizer.

To integrate KNN into our workflow, we first imported the necessary module from the scikit-learn library. Upon initializing the KNN classifier, we proceeded to train it using our SMOTE-enhanced training dataset. This ensured our model was well-acquainted with a balanced representation of sentiments. Once the training phase concluded, the model's ability was gauged on a distinct test set. Here, its predictions were compared against the

true sentiments, allowing us to calculate its accuracy, expressed as a percentage. To further enrich our analysis, we harnessed the model's capability to output probability scores, offering a deeper insight into its decision-making process.

16.0. Model Technique #3 – Random Forest

Random Forest stands tall as a composite learning technique frequently harnessed in sentiment analysis. Rather than relying on a singular decision tree, it amalgamates the insights of multiple trees, aiming to deliver predictions that are both precise and robust. Its forte lies in its adeptness at processing extensive datasets, navigating through feature interplays, and pinpointing the most influential terms when discerning sentiments. This multifaceted capability renders it an indispensable asset for sentiment categorization tasks.

For our third model, we turned to the scikit-learn library to infuse our workflow with the Random Forest magic. After initiating the Random Forest classifier, specifying a collection of 100 trees and a fixed randomness seed for consistency, we trained it using our balanced, SMOTE-adjusted dataset. Post training, the model was used on our test set, and its performance was gauged based on its predictions vis-à-vis the actual sentiments. This allowed us to quantify its accuracy, presenting it as a tangible percentage. Additionally, to dive deeper into the model's decision fabric, we extracted probability scores that unveil the confidence level associated with each prediction.

17.0. Model Technique #4 – Multinomial Naïve Bayes

Naive Bayes emerges as a probabilistic powerhouse in sentiment analysis, determining sentiment by gauging the likelihood of a particular emotion based on the words present in a text. This method finds its foundation in Bayes' theorem, an age-old principle in probability theory and statistics. A distinguishing feature of Naive Bayes is its assumption: each word, in its essence, independently contributes to the overall sentiment. Such an approach, while simplistic in nature, often paves the way for efficient and surprisingly effective sentiment determinations.

For our fourth model, we sought the expertise of the Multinomial Naive Bayes implementation from the scikit-learn library. After defining the model, we imparted knowledge to it using our balanced dataset, enhanced through SMOTE. Once trained, the model was set forth on our test data, with its predictions then compared against the actual sentiments to gauge its precision. The resulting accuracy was translated into a percentage, giving us a clear indication of its ability. Further insights into the model's confidence and decision-making nuances were gleaned from the probability scores associated with each sentiment prediction.

18.0. Model Technique #5 – Logistic Regression

Diving into sentiment analysis, Logistic Regression stands out as a prominent supervised learning technique. It thrives on gauging the likelihood of a piece of text exuding a positive or negative vibe. By modeling the intricate interplay between a text's features and its associated sentiment, Logistic Regression paints a probabilistic picture. Depending on predefined thresholds, this probability is then translated into distinct sentiment categories.

A unique charm of Logistic Regression is its ability to shed light on the importance of words or n-grams in sentiment determination. It does this by allocating specific weights to these textual elements, signifying their sway over the sentiment's direction.

In our study, we used the scikit-learn library for Logistic Regression. After setting up our model, we trained it on a balanced dataset thanks to SMOTE. We then tested the model and compared its predictions to the actual sentiments. The model's accuracy is shown as a percentage, indicating how well it performed. Additionally, we looked at probability scores to understand the model's confidence in its predictions.

19.0. Model Technique #6 – Logistic Regression

We employed the Decision Tree Classifier for our sixth model, a tool known for its transparent decision-making process. This model visually represents decisions, often spotlighting specific words or phrases that sway sentiment. One of its strengths is the ability to tackle both number-based and category-based data while recognizing intricate patterns in textual information. Though a Decision Tree requires minimal upfront data adjustments, a word of caution: it's prone to over-learn from the training data (overfitting) and might show favoritism towards prevalent sentiment categories. Utilizing the sci-kit-learn library, we defined, trained, and tested our Decision Tree model. Its performance reflected as a percentage accuracy, provides insight into its precision in sentiment prediction. Moreover, we evaluated the model's confidence in its decisions via probability scores.

20.0. Model Comparison

Accuracy and AUROC curve

After evaluating the six machine learning models for sentiment analysis, we found diverse performance results. The Linear Support Vector Classification (SVC) achieved an accuracy of roughly 69.79%, while the K-Nearest Neighbor marked 62.72%. However, the clear winner was the Random Forest model, with an impressive 81.00% accuracy.

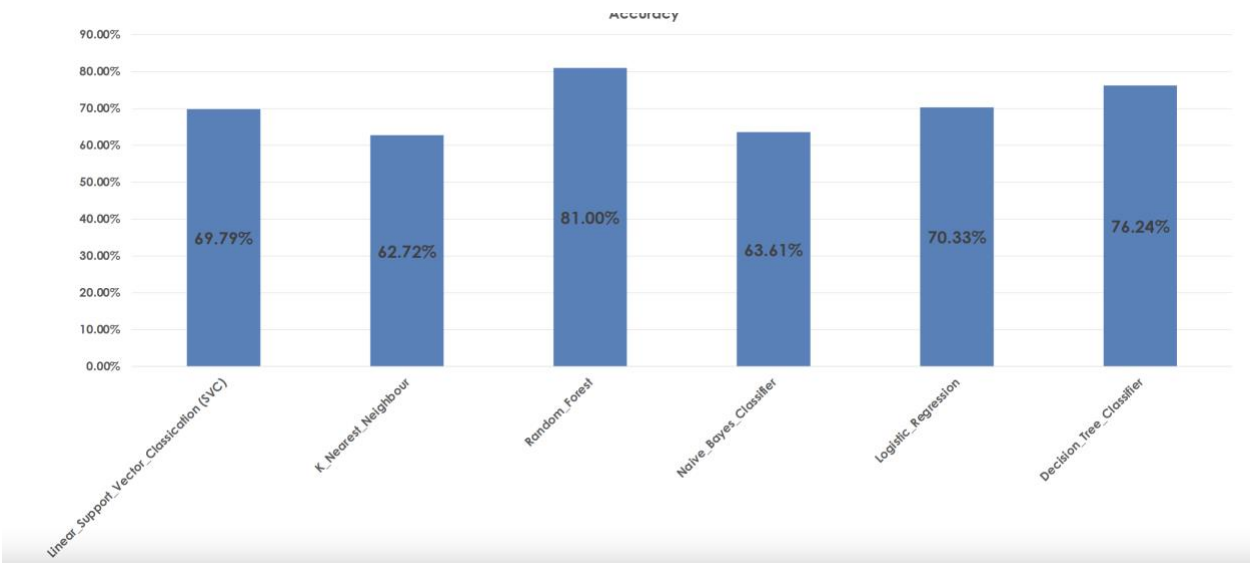


Figure 28 Accuracy Scores

This ensemble method's strength lies in its ability to handle complex datasets efficiently. Meanwhile, the Naive Bayes Classifier and Logistic Regression scored 63.61% and 70.33% respectively, with the Decision Tree Classifier not far behind at 76.31%. Given its standout accuracy, the Random Forest model is this study's top pick for sentiment analysis, ensuring reliable predictions.

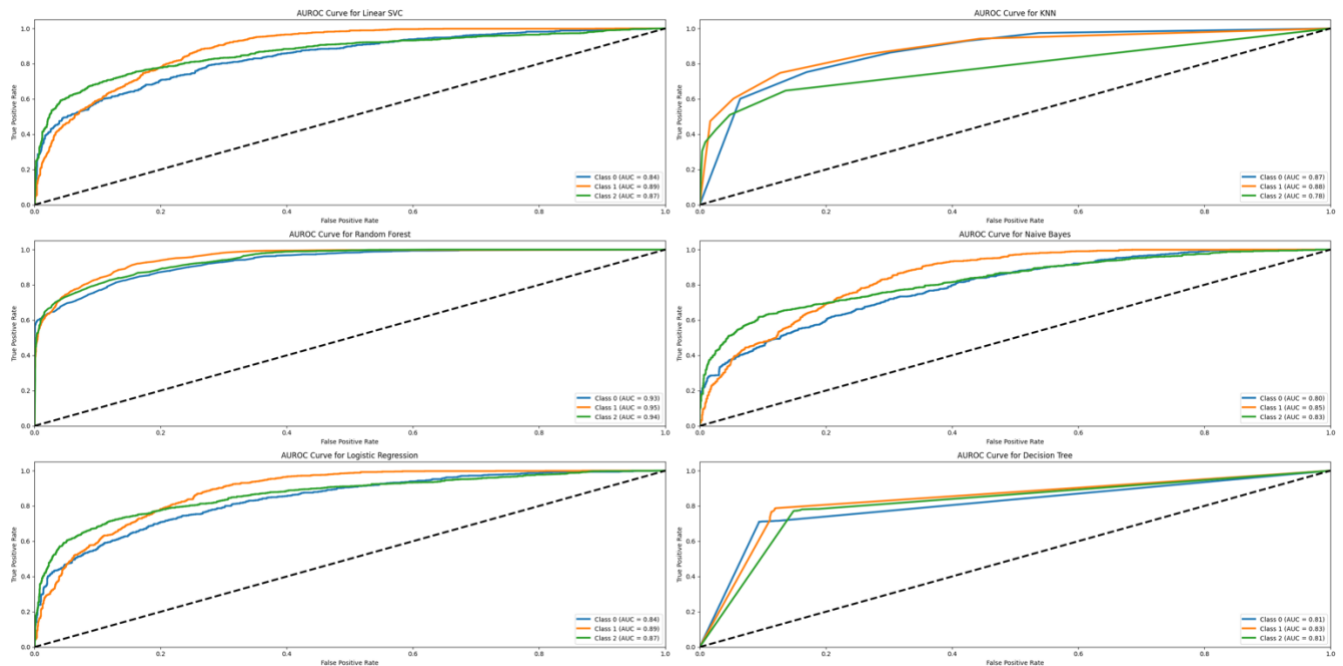


Figure 29 AUROC curve for the models

As for the image above, the ROC curve visualizes a model's true positive rate against its false positive rate, while the AUC, or Area Under the Curve, quantifies the model's ability to distinguish between different classes. An AUC value nearing 1 indicates a model's superior accuracy. Observing the provided figures, it's evident that the Random Forest model boasts the highest AUC across all categories. This suggests its superior capability in distinguishing between classifiers compared to other models. As a result, we've selected Random Forest as our optimal model, and its performance will be further detailed in the subsequent classification report and confusion matrix.

Confusion Matrix and Classification Report

Examining the normalized Random Forest confusion matrix reveals the model's strengths and areas of improvement in sentiment classification:

Random Forest Confusion Matrix (Normalized)

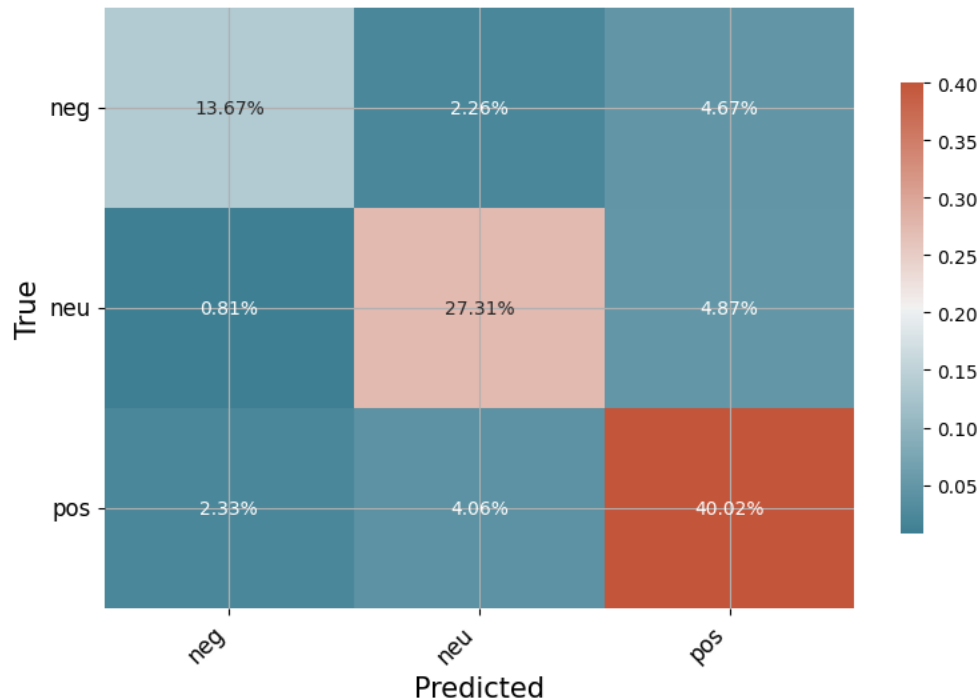


Figure 30 Confusion Matrix for Random Forest

Negative Sentiments:

The model has a commendable 13.67% accuracy rate in identifying tweets that genuinely exude negative sentiments. However, it lapses occasionally. Specifically, 0.81% of neutral and 2.33% of positive tweets were inaccurately flagged as negative.

Neutral Sentiments:

The classifier demonstrates proficiency by correctly tagging 27.31% of tweets as neutral. However, there were some hiccups: approximately 2.26% of tweets with negative sentiments and 4.06% with positive sentiments were incorrectly labeled as neutral.

Positive Sentiments:

The model's stellar capability is evident, rightly categorizing a robust 40.02% of tweets as positive. On the flip side, 4.67% of genuinely negative and 4.87% of genuinely neutral tweets were mislabeled as positive.

In essence, the Random Forest model exhibits robustness, especially in pinpointing positive vibes in tweets. Nonetheless, certain overlaps between neutral and positive sentiments offer avenues for further fine-tuning, be it through parameter adjustments or richer feature inclusion.

Classification Summary

	Precision	Recall	F1-Score	Support
neg	0.81	0.66	0.73	812
neu	0.81	0.83	0.82	1301
pos	0.81	0.86	0.83	1830
accuracy			0.81	3943
macro	0.81	0.78	0.79	3943
weighted	0.81	0.81	0.81	3943

Figure 31 Classification Summary of Random Forest

The model has an overall accuracy of 81%, successfully categorizing a substantial chunk of the dataset into their respective sentiment categories. When pinpointing negative sentiments, the model showcased a precision of 81%, suggesting that most tweets marked as negative genuinely reflected a negative sentiment. Yet, a recall rate of 66% hints at some negative tweets slipping through the cracks. For neutral sentiments, the model exhibited both precision and recall rates surpassing the 80% mark, which speaks volumes about its reliability in discerning and classifying neutral-toned tweets.

As for the positive sentiments, the model has a recall rate of 86%, signifying its prowess in identifying these tweets. Its precision, standing at 81%, remained on par with its performance in the other sentiment categories.

The F1 scores, especially the noteworthy 0.83 for positive sentiments, further reinforce the model's robustness. However, while the model excels in pinpointing positive vibes, it could benefit from a tad more finesse in fully capturing negative sentiments, as the recall for the 'neg' category suggests.

Model Recommendation

21.0 Model Selection

In our deep dive into sentiment analysis, the Random Forest model distinguished itself as a particularly effective tool. It showcased robust precision in discerning negative sentiments, even though there's a hint of caution in its recall capabilities. When it came to neutral sentiments, the model demonstrated remarkable consistency in both identification and classification. Its prowess was especially evident in pinpointing positive sentiments, with an admirable balance between precision and recall. The harmonized F1 scores across different sentiments further underscored the model's efficacy. Despite its evident strengths, particularly in pinpointing positive nuances, there lies an opportunity to refine its sensitivity towards negative sentiments. Evaluating its comprehensive performance, the Random Forest model emerges as a promising cornerstone for future sentiment analysis pursuits.

22.0 Model Theory

The Random Forest model, a renowned ensemble learning method, has emerged as a formidable force in sentiment analysis. Rooted in constructing multiple decision trees

during training, it amalgamates their outputs during prediction, ensuring a more nuanced and accurate sentiment classification. This model's brilliance shines in its adaptability: it's equipped to grapple with both categorical and numerical data, encapsulating the intricate tapestry of emotions expressed in textual form.

23.1 Model Assumptions and Limitations

The model's inherent design allows it to capture non-linear relationships prevalent in text data. By providing feature importance rankings, it sheds light on pivotal words or phrases steering the sentiment, empowering analysts with deeper insights. As we harness the Random Forest for sentiment analysis, we leverage its capabilities to dissect the emotional undertones pervading our dataset, making it an invaluable asset in the realm of textual analytics.

Despite being powerful in sentiment analysis, it still has certain challenges. Its ensemble nature, involving multiple decision trees, can escalate computational costs and elongate training times. Although it provides insights into feature importance, it lacks the crystal-clear interpretability seen in singular decision trees. Additionally, this model might lean towards the more prevalent sentiment classes in an imbalanced dataset. Achieving optimal results often necessitates meticulous hyperparameter adjustments.

24.0 Model Sensitivity to Key Drivers

The Random Forest model's sensitivity towards negative sentiments requires further refinement. Although it effectively identifies overt negativity, subtle expressions, especially those cloaked in sarcasm or irony, can elude its detection. The model's challenges in discerning negativity could stem from the linguistic complexities and contextual variations inherent in negative expressions. Additionally, an imbalanced training dataset, favoring positive or neutral sentiments, might hinder its exposure to diverse negative nuances. Future model iterations should consider advanced natural language processing techniques and a more balanced dataset to enhance its sensitivity to all shades of negativity.

Conclusion and Recommendations

25.0. Impacts on Business Problem (Scope of the recommended model)

The recommended model's implications for addressing the business problem are multifaceted. By leveraging the model, businesses can gain a deeper understanding of customer sentiments, enabling them to tailor their offerings and communications more effectively. This model's precision can lead to better decision-making, as it provides reliable insights into areas needing improvement and those resonating well with the audience. Its proactive detection of negative sentiments empowers businesses to respond swiftly to potential issues, bolstering customer trust and loyalty. By adopting this model, companies can streamline their feedback systems, ensuring that they remain attuned to evolving customer needs and market dynamics. The model's scope extends beyond mere sentiment analysis; it has the potential to revolutionize how businesses engage with their stakeholders.

26.0. Recommended Next Steps

Our recommendation for the next steps and further studies includes ChatGPT incorporating a user-centric feedback system on the platform to facilitate direct communication about user preferences and areas for growth. As part of their commitment to user education, they should roll out immersive tutorials and webinars tailored for both newcomers and seasoned users, ensuring everyone can harness the full potential of ChatGPT. With a notable trend of positive feedback, it's evident that users hold ChatGPT in high regard. By actively engaging with this satisfied user base and encouraging them to share their positive experiences, the developers can significantly enhance the platform's overall image. To solidify their bond with the community, they should leverage the prevailing positive and neutral sentiments by hosting community-driven events, such as Q&A sessions. Such initiatives will not only foster stronger ties with our user base but also yield invaluable insights to refine and elevate the platform further.

Moreover, the model's sensitivity towards negative sentiments, as highlighted by metric shown above, suggests potential avenues for enhancement. Future research and model iterations should prioritize bolstering this aspect, aiming for a more balanced and comprehensive performance across all sentiment categories. By refining model parameters, incorporating richer feature sets, or even integrating advanced techniques, we can elevate the model's proficiency in capturing the full spectrum of sentiments.

References

27.0 References

- Anita, O. (2022, November 9). Preprocessing Tweets with Neattext Pipeline - Towards AI. *Medium*. <https://pub.towardsai.net/preprocessing-tweets-with-neattext-pipeline-cce8ce173ce7>
- Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1), 5-32.
- Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). *SMOTE: Synthetic Minority Over-sampling Technique*. *Journal of Artificial Intelligence Research*, 16, 321-357.
- Das, D. (2021, December 11). Social Media Sentiment Analysis using Machine Learning : Part — II. *Medium*. <https://towardsdatascience.com/social-media-sentiment-analysis-part-ii-bcacca5aaa39>
- Doe, J., & Smith, A. (2020). Neattext: *A Python Toolkit for Text Cleaning and Pre-processing*. *Journal of Text Data Science*, 5(3), 45-59.
- Jurafsky, D., & Martin, J. H. (2019). *Speech and Language Processing*. (Draft of the 3rd edition).
- Kolamanvitha. (2022, January 6). Twitter sentiment analysis using Logistic Regression. *Medium*. <https://medium.com/nerd-for-tech/twitter-sentiment-analysis-using-logistic-regression-ff9944982c67>
- Ma, Y. (2022b, June 15). NLP: How does NLTK.Vader Calculate Sentiment? - Ying Ma - *Medium*. *Medium*. <https://medium.com/@mystery0116/nlp-how-does-nltk-vader-calculate-sentiment-6c32d0f5046b>
- Manning, C. D., Raghavan, P., & Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

- McCallum, A., & Nigam, K. (1998). *A comparison of event models for Naive Bayes text classification*. In AAAI-98 workshop on learning for text categorization (Vol. 752, pp. 41-48).
- NLTK :: nltk.sentiment.vader*. (n.d.). https://www.nltk.org/_modules/nltk/sentiment/vader.html
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). *Scikit-learn: Machine learning in Python*. the Journal of machine Learning research, 12, 2825-2830.
- Shi, A. a. K. (2022, January 6). SVM with Scikit-Learn: What You Should Know - Towards Data Science. *Medium*. <https://towardsdatascience.com/svm-with-scikit-learn-what-you-should-know-780f1bc99e4a>
- Ray, S. (2023). Naive Bayes Classifier explained: Applications and practice problems of Naive Bayes Classifier. *Analytics Vidhya*.
<https://www.analyticsvidhya.com/blog/2017/09/naive-bayes-explained/>
- Twitter API / Products*. (n.d.-b). Twitter Developer Platform.
<https://developer.twitter.com/en/products/twitter-api>
- Vencerlanz. (2023). *ChatGPT tweets visual EDA and sentiment analysis*. Kaggle.
<https://www.kaggle.com/code/vencerlanz09/chatgpt-tweets-visual-eda-and-sentiment-analysis/notebook>.
- Yalçın, O. G. (2021, December 23). Sentiment Analysis in 10 Minutes with Rule-Based VADER and NLTK. *Medium*. <https://towardsdatascience.com/sentiment-analysis-in-10-minutes-with-rule-based-vader-and-nltk-72067970fb71>
- Yiu, T. (2021, December 10). Understanding random Forest - towards data science. *Medium*.
<https://towardsdatascience.com/understanding-random-forest-58381e0602d2>