Research Report – ChatGPT App Reviews: User Sentiment and Feature Insights

Group 5: Shuting Zhang, Tanvi Dinesh Nandu, William Wang, Xiaoyi Wang, Xiaoyang Li

Table of Contents

Abstract:

This study analyzes 278,337 user reviews of the ChatGPT app from the Google Play Store to uncover insights into user sentiment, app performance, and feature feedback. Using advanced Natural Language Processing (NLP) techniques, the research addresses three core areas: identifying commonly mentioned app features across sentiment categories, examining sentiment evolution across app updates, and leveraging bigram analysis to explore user feedback trends and prompt handling challenges.

The dataset, sourced from Kaggle, includes user ratings, textual reviews, app versions, and other features. Exploratory Data Analysis (EDA) provided insights into sentiment trends, ratings distribution, and app version review counts. Text preprocessing ensures data quality, involving tokenization, lemmatization, and filtering non-English content. Sentiment analysis using the VADER library highlighted nuanced feedback, incorporating emojis, slang, and punctuation. Bigram analysis revealed recurring patterns and challenges in user feedback.

Key findings include fluctuating sentiment trends, with response quality earning the highest positive sentiment and updates and prompt handling showing room for improvement. The study provides actionable recommendations to enhance user satisfaction, optimize app updates, and refine prompt handling, showcasing the value of NLP techniques in analyzing large-scale user feedback to inform app development strategies.

Introduction:

ChatGPT, developed by OpenAI, has become a widely used conversational AI application, providing advanced natural language understanding and response generation capabilities. Its popularity among diverse user groups highlights the importance of evaluating user feedback to optimize performance and address evolving expectations. App reviews, particularly those on platforms like the Google Play Store, provide a wealth of information about user experiences, feature satisfaction, and areas for improvement.

This study focuses on analyzing user reviews of the ChatGPT app, specifically addressing three research questions:

1. Which specific features or aspects of the ChatGPT app (e.g., performance, user interface, response quality) are frequently mentioned in positive, neutral and negative reviews?
2. How has user's sentiment (positive, neutral and negative reviews) on specific features evolve across different app updates?
3. How does bigram analysis reveal changes in user feedback across app versions and the challenges users face in prompt handling?

By examining these questions, the research aims to provide actionable insights for improving app updates, addressing user pain points, and enhancing overall user satisfaction.

Sentiment analysis has long been a cornerstone of Natural Language Processing (NLP). Tools such as VADER (Valence Aware Dictionary and Sentiment Reasoner) have proven effective for analyzing social media and app reviews due to their ability to process informal language, emojis, and punctuation (Hutto & Gilbert, 2014). Bigram analysis, a common text mining technique, has been used to identify recurring themes in user feedback, helping developers prioritize updates based on user needs (Jurafsky & Martin, 2019). Furthermore, FastText, a pre-trained language detection model, was utilized to filter non-English reviews, ensuring the dataset focused on English feedback. FastText's ability to enrich word vectors with subword information makes it particularly suitable for analyzing informal and diverse text data (Bojanowski, Grave, Joulin, & Mikolov, 2017).

This study utilized a dataset of app reviews sourced from Kaggle. Data preprocessing included tokenization, lemmatization, and language filtering using FastText to ensure a high-quality dataset for analysis. Exploratory Data Analysis (EDA) uncovered trends in user sentiment, ratings distribution, and feature-specific feedback across app updates. Sentiment analysis using VADER provided nuanced insights into user sentiment, while bigram analysis revealed common patterns and challenges in user interactions.

By leveraging NLP techniques, this research provides a comprehensive understanding of user sentiment and feature feedback for the ChatGPT app. The findings offer valuable recommendations for developers to address user concerns, refine app features, and guide future updates, emphasizing the critical role of user feedback in shaping app development strategies and enhancing the app experience.

Data Overview:

The dataset used in this study was sourced from Kaggle and contains 278,337 user reviews of the ChatGPT app from the Google Play Store. This dataset captures diverse user feedback across multiple app versions and geographic regions, making it a valuable resource for analyzing sentiment trends and feature-specific feedback.

Key Features: The dataset includes the following fields:

1. Review ID: A unique identifier for each review.
2. User Name: The name or pseudonym of the reviewer. (with xyz missing values)
3. Content: The textual content of the review, which provides insights into user opinions and experiences. (includes emojis)
4. Score: A numeric rating (1–5), where 1 represents the lowest satisfaction and 5 indicates the highest satisfaction.
5. Thumbs-Up Count: The number of likes a review has received, indicating its perceived helpfulness by other users.
6. Review Created Version: The version of the ChatGPT app being reviewed, allowing for trend analysis across updates. (with some missing values)
7. Date and Time: The timestamp when the review was posted, enabling time-series analysis.
8. App Version: Had the same values as *'Review Created Version'*

Dataset Characteristics

1. Size: The dataset contains 278,337 rows of user reviews, offering a large sample size for robust analysis.

2. Language: Approximately 90% of the reviews were identified as English using the FastText language detection model. Non-English reviews were excluded to maintain consistency in analysis.

3. Diversity: The reviews seem to originate from users across different countries and reflect feedback on various app versions, providing a comprehensive view of user experiences.

The dataset was leveraged to address the following use cases:

1. Sentiment Analysis: Evaluating user sentiment (positive, neutral, and negative) to identify satisfaction levels and areas for improvement.

2. Feature Feedback: Analyzing user opinions on specific features such as performance, user interface, and prompt handling.

3. Trend Analysis: Monitoring changes in sentiment and feedback across different app versions and time periods.

Data Preprocessing: To prepare the data for analysis, several preprocessing steps were applied:

1. Text Cleaning: Converting text to lowercase, removing unnecessary punctuation while retaining emojis, and eliminating excessive whitespace.

2. Tokenization and Lemmatization: Splitting reviews into individual tokens and reducing them to their base forms to improve analysis consistency.

3. Language Filtering: Using the FastText model to exclude non-English reviews while accounting for transliterated text that might remain undetected.

4. Emoji Sentiment Analysis: Extracting sentiment scores for emojis in the reviews using the VADER library. VADER's scoring system allows for detailed analysis of sentiment expressed through emojis, emoticons, and informal language. The compound score, calculated as a normalized sum of the valence scores of all text elements (including emojis, punctuation, and intensifiers), provided a single sentiment value ranging from -1 (most negative) to 1 (most positive). VADER also assigned polarity scores to each review, classifying sentiments into positive, negative, or neutral categories. This step was critical for capturing nuanced user feedback that may not have been evident in text alone.

This dataset's comprehensive nature and preprocessing enable a detailed analysis of user sentiment and feature-specific feedback, forming the foundation for this study's findings and recommendations.

Research Question 1

In research question 1, we focused on “Which specific features or aspects of the ChatGPT app (e.g., performance, user interface, response quality) are frequently mentioned in positive, neutral and negative reviews?”. To answer this question, we have divided the solution into three steps.

Methodology

1. Begin by identifying the key features to evaluate in the reviews, such as performance, user interface, response quality, prompt handling, and updates. For each feature, compile a list of associated keywords or phrases that users might use to describe it. For example, keywords for performance might include "faster," "slow," "lag," or "efficient," while for user interface, keywords might include "ui," "recording," or "voice." （Appendix 1) This dictionary serves as the foundation for analyzing and categorizing user feedback, ensuring that the sentiment analysis focuses on relevant aspects of the product or service.

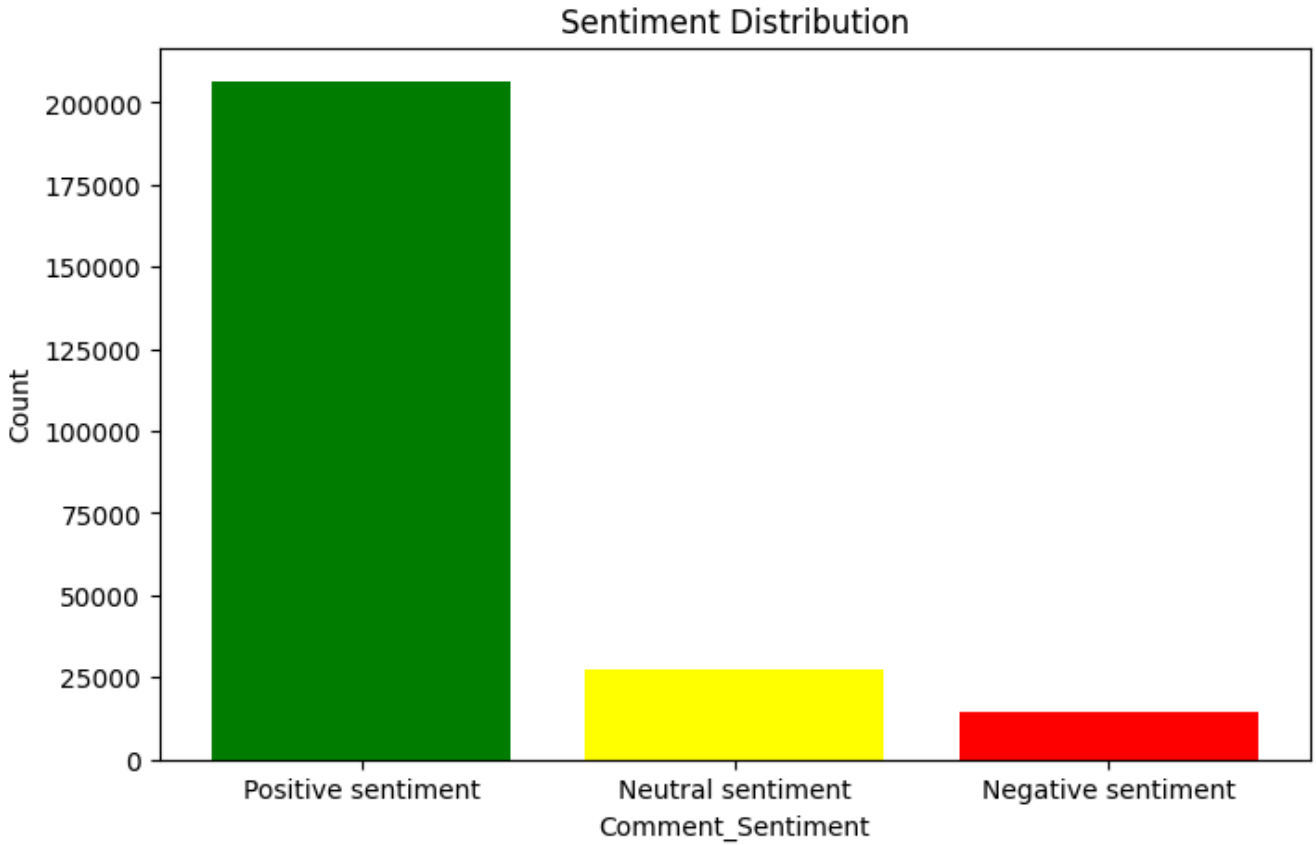
2. The second step is to analyze the text of the reviews using a sentiment analysis tool, such as Vader Sentiment. By focusing on sentences or phrases containing the identified keywords, extract sentiment scores specific to each feature. This step ensures that the analysis captures the nuanced feedback for each feature, providing insights into user opinions about distinct aspects of the product or service.

3. Finally, once the sentiment scores for all reviews are obtained, aggregate the scores for each feature to determine the overall sentiment trend. For instance, the sentiment scores for the response quality feature show the highest frequency of positive scores, it indicates that users generally view this feature most favorably. This step provides a clear, data-driven understanding of user sentiment across all features, guiding targeted enhancements and strategic decision-making.

Results

For the findings, ‘response_quality’ stands out with the highest positive sentiment count, suggesting a strong area of user satisfaction. Both prompt handling and updates have a high count of positive feedback, though they also show noticeable negative sentiments, indicating areas where improvements might be expected. ‘User_interface’ has strong positive sentiment but comparatively more neutral and negative sentiments than response quality, signaling room for enhancement. ‘Performance’ has the lowest positive sentiment count and the smallest overall feedback volume, suggesting it may be less impactful to users but also an area for improvement. While "Response Quality" has the highest total feedback, its negative sentiment proportion is relatively low compared to other features. In contrast, features like "Updates" and "Prompt Handling" have a slightly higher percentage of negative feedback, indicating these areas may require more attention to reduce dissatisfaction. Statistically, over 15,000 responses mentioned “response quality”, both “updates” and “prompt handling” are mentioned over 8,000 times. Performance is the least

discussed feature with less than 5,000 mentions. This shows that users value response quality the most, while performance could be an area to improve. Overall, 83.2% of the sentiment distribution are positive, 10.9% of the sentiment distribution are neutral, and 5.9% of the sentiment distribution are negative. That means users are very satisfied with ChatGPT, with only a small amount of negative feedback to address.



In terms of the most frequently mentioned keywords, keywords like "easy," "faster," and "accurate" received overwhelmingly positive sentiment, indicating strong satisfaction in “performance”. In “user interface”, terms such as "feel," "UI," and "responsive" had mostly positive mentions, but there were minor negative scores for "customize" and "navigation,"

suggesting room for improvement in flexibility and usability. In “prompt handling”, keywords like "generate" and "question” gained a predominantly positive sentiment. Few negative mentions were noted for "interpret" and "execution." In “updates”, while keywords like "version" and "improvement" were viewed positively, words such as "bug," "error," and "outdated" had significant negative mentions, highlighting issues with update quality and reliability.

Conclusion:

In summary, the analysis highlights that users overwhelmingly value the ChatGPT app for its response quality, which received the highest positive sentiment and the most mentions, reflecting strong satisfaction in this feature. While performance had the lowest engagement and positive sentiment, addressing its shortcomings could enhance user experience further.

Research Question 2

In research question 2, we examined the question “How has the user's sentiment (positive, neutral and negative reviews) on specific features evolved across different app updates?” We studied this question in three steps.

Step 1

First, we evaluated user sentiment trends (Positive, Neutral, and Negative) across different app versions. To achieve this goal, a systematic and rigorous approach was adopted. The following methodologies were applied.

Methodology

1. Further Data Cleaning and Preprocessing - To ensure the analysis was based on reliable and meaningful data, further data cleansing process was undertaken. We filtered Invalid Entries. Data rows with null or "Unknown" values in the appVersion column were excluded to maintain the validity of app version comparisons. Similarly, entries with missing Comment_Sentiment values were removed to ensure complete data.

The app versions were numerically sorted to preserve the natural progression of updates, enabling accurate trend visualization and analysis. Through research, we identified that 1.0.XXXX is GPT1.0 version, 1.2023.XXX is GPT3.0 version, and 1.2024.XXX reaches GPT4.0 version. Cleaning the data ensures that only valid and complete entries were analyzed, reducing noise and improving the reliability of the results.

2. Sentiment Categorization and Aggregation - The data was grouped by appVersion and Comment_Sentiment (In Dataset Characteristics we have mentioned that Sentiments were classified into three categories: Positive, Neutral, and Negative.) to compute the counts for each sentiment category. In this step, a distribution matrix was constructed to summarize the frequency of each sentiment type across all app versions.

3. Normalization and Percentage Calculations - Since the number of reviews for each app version was different, we normalized the data into percentages provided a standardized view of sentiment distributions. To enable cross-version comparison, raw sentiment counts were normalized into percentages: The count of each sentiment category was divided by the total count for the respective app version, yielding sentiment proportions. If any sentiment category was absent for a specific app version, its count was assigned a zero value to maintain consistency across versions.

4. Visualization Techniques - A stacked column chart was created to depict the proportional distribution of Positive, Neutral, and Negative sentiments for each app version.

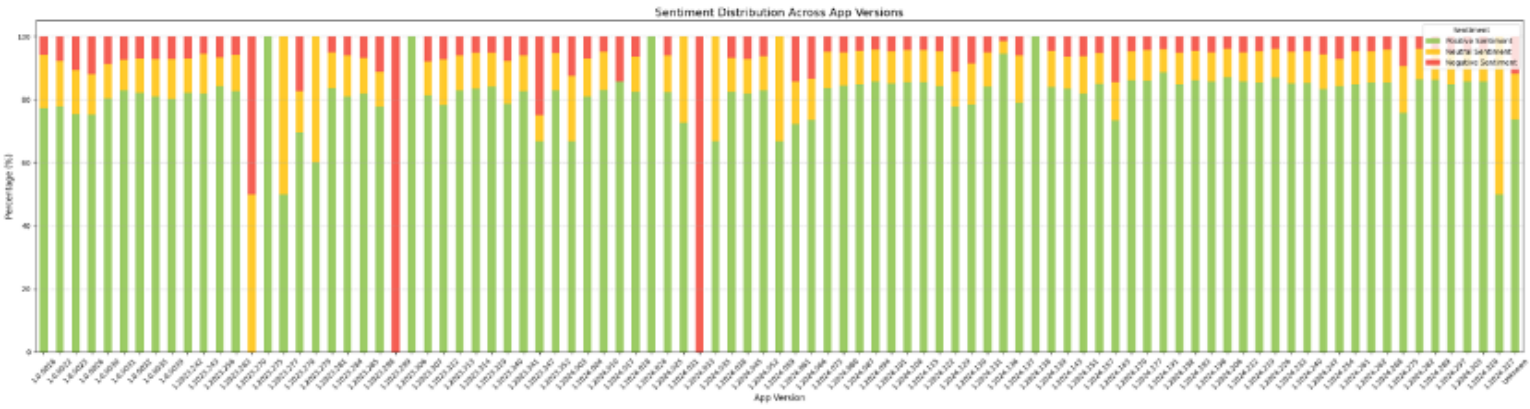
Trends in sentiment percentages over time were illustrated using line plots, with separate lines for Positive, Neutral, and Negative sentiments.

Sentiments were visually distinguished using colors—green for Positive, yellow for Neutral, and red for Negative.

5. Addressing Edge Cases and Outliers - The first graph output was not good looking since there were several versions which only had a few reviews which will lead to a “edge cases” (e.g. 0%, 100%). Thus, we made adjustments to deal with them.

- Excluding Extremes: Data points with extreme values (e.g., 0% or 100% sentiment proportions) were excluded from trend calculations to avoid skewing results.

- Validating Empty Groups: Checks were performed to ensure that sentiment distribution was not empty for any app version, flagging potential issues in data completeness.



Result

Positive Sentiment (Green): Shows a general upward trend across app updates, indicating improved user satisfaction as issues were resolved and new features were introduced.

Negative Sentiment (Red): Dominates in earlier versions but gradually declines, reflecting effective bug fixes and problem resolution in later updates.

Neutral Sentiment (Yellow): Fluctuates, possibly due to users adapting to experimental features or unclear benefits of updates.

Step 2

In Step 2, we analyzed the percentage of reviews for each sentiment category within functional areas.

Methodology: The preprocessing requirement in this step is the same as in Step 1. So the entries with missing Comment_Sentiment values or undefined functional area classifications were already excluded.

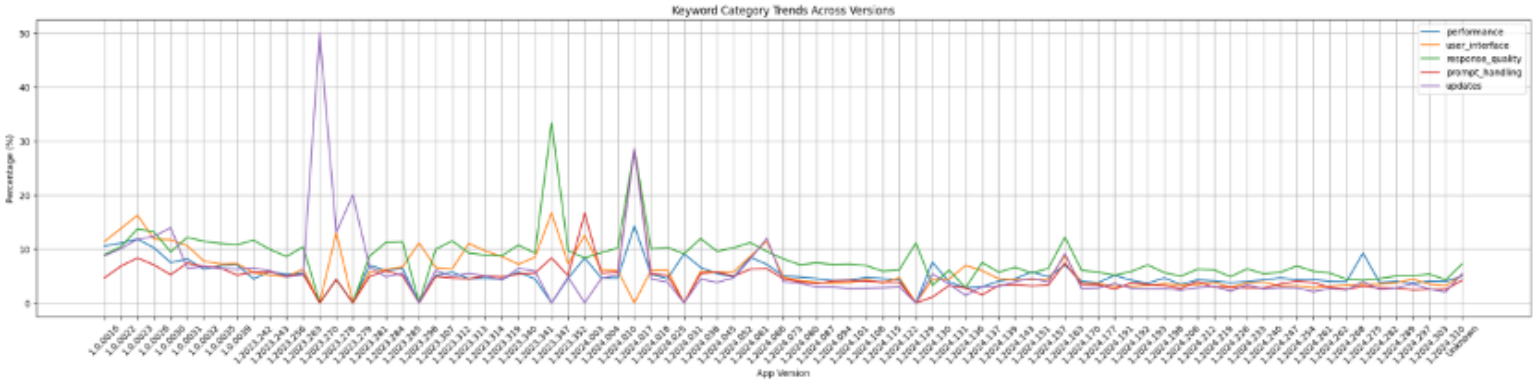
1. Categorization and calculation The reviews were grouped by key features and sentiment category - Sentiment Classification: Positive, Neutral, or Negative.

Functional Area Assignment: We used code to match reviews with keywords, so that each review was tagged with one or multi key features based on keywords we designed in Research Question 1.

The total number of reviews for each sentiment category within each functional area was calculated. Similar to step 1, to enable meaningful comparisons across functional areas, raw counts were converted into percentages.

2. Visualization To communicate the results effectively, visualization techniques were applied:

Line Graphs for Trends: Line graphs were used to plot the percentage of reviews mentioning each functional area across app versions.



The third step is to visualize the data in a different form.

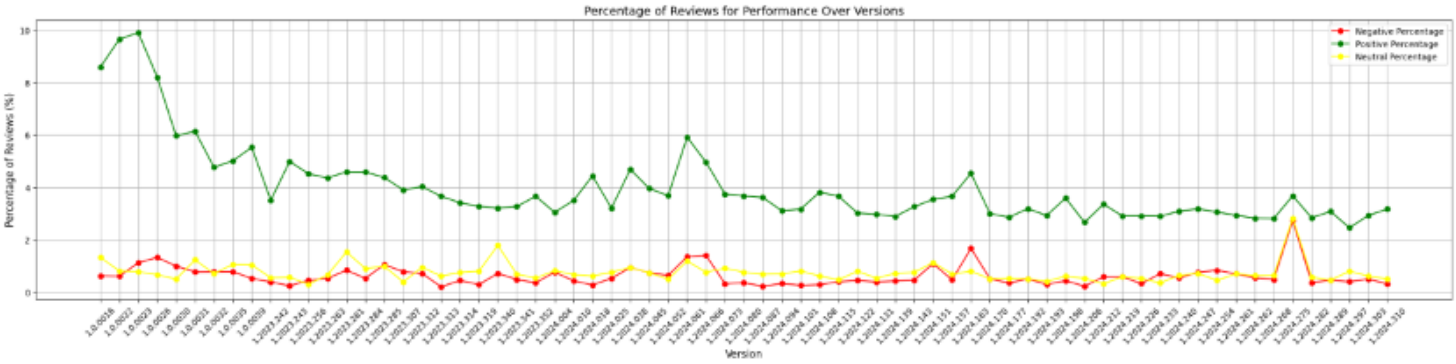
Line graphs were used to plot sentiment trends (Positive, Neutral, Negative) for each feature over app updates.

This plot is useful for showing trends over time. The changes in user sentiment for each feature (such as performance, user interface, response quality, etc.) between different versions are time series data. Line graphs are very suitable for showing how data evolves with the version number, helping to intuitively identify rising, falling or stable sentiment trends.

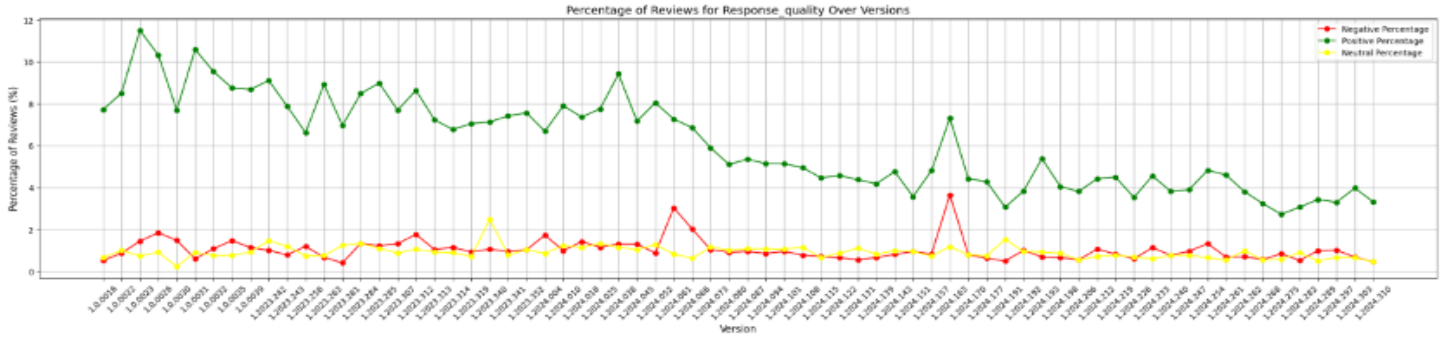
With three different lines (positive, negative, and neutral sentiment), the relative changes of three emotions can be compared at the same time. This visualization method can intuitively convey information in multiple dimensions without the need for separate analysis or comparison.

Sentiment trends for different features can be presented as separate graphs in the same chart style. This consistency facilitates cross-feature comparisons and helps development teams discover global patterns.

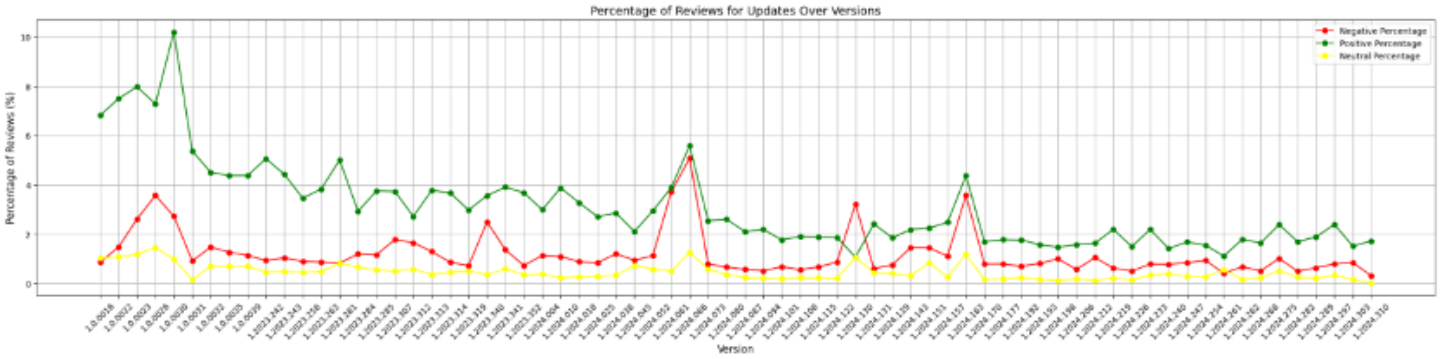
Results:



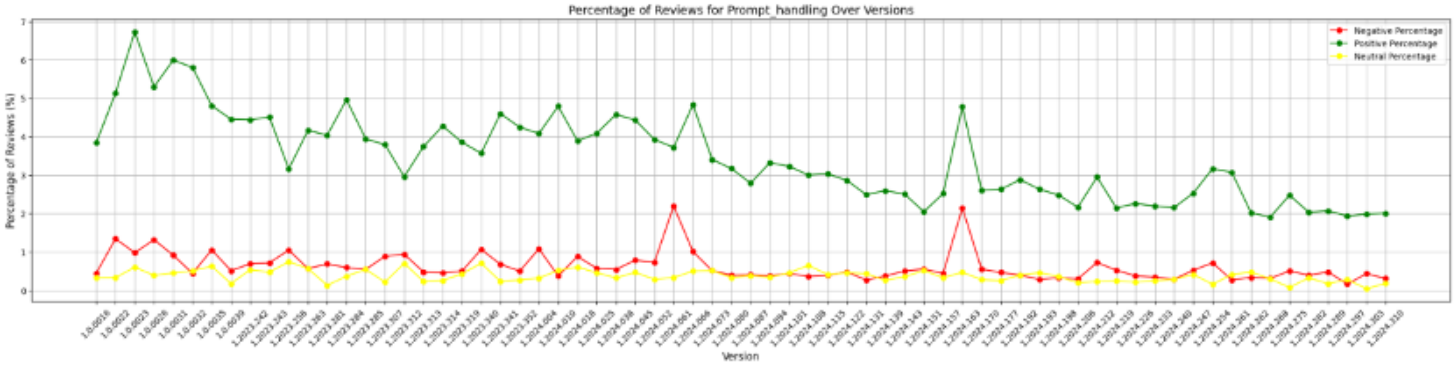
Performance: Maintains high proportions, highlighting its importance to users.



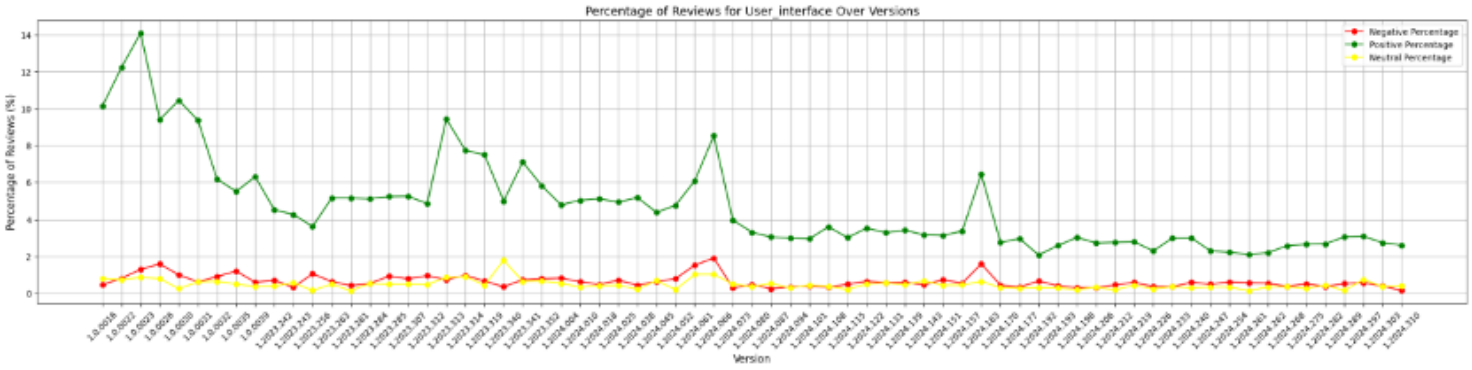
Response Quality: Consistently dominates user feedback, reflecting its critical role in the user experience.



Updates: In early versions (GPT-1), user expectations for updates were high, but in later versions (GPT-3 and GPT-4), update-related comments significantly decreased. This indicates that users view the application as mature and no longer expect frequent updates.



Prompt handling: Similarly, user satisfaction with the Prompt Handling feature has not continued to improve; negative reviews (red line) occasionally show significant peaks, indicating that individual versions may have functional issues and require further attention.



User interface: Users' satisfaction with the user interface was high in the early stage but did not improve significantly in the later stage, and there was no significant fluctuation in dissatisfaction and neutral feedback.

We observe that trends for all five features are similar. Here are the hypothesis about the Data Distribution Patterns:

- Focus Shift: As the app evolved, user attention likely shifted from basic issues (e.g., performance, interface) to more advanced functionalities like coding or writing assistance. This dilution may reduce the prominence of positive feedback for foundational features.
- Sentiment Classification Bias: Users tend to express neutral sentiments more frequently unless they feel extremely satisfied or dissatisfied. This creates an illusion of stability in Neutral and Negative sentiments while Positive sentiment visibly declines.
- Expectation-Reality Gap: As the app improves, user expectations increase. If updates fail to fully meet heightened expectations, positive feedback may drop, even if the app objectively performs better.

Conclusion:

Positive sentiment generally increased in earlier versions but began declining as expectations rose and attention shifted to advanced functionalities.

Response quality and performance remain core user concerns, showing the highest levels of feedback.

Future improvements should focus on advanced features while maintaining reliability in foundational functionalities.

Research Question 3

In research question 3, we examined the question “How does bigram analysis reveal changes in user feedback across app versions and the challenges users face in prompt handling?” Building upon the findings from earlier analyses, this research question investigates how bigram analysis can uncover changes in user feedback across app versions and highlight challenges in prompt handling. The approach is structured into four key steps:

1. Introduce the bigram analysis methodology to identify recurring patterns and themes in user feedback.

2. Compare high-frequency bigrams across different app versions to observe how user concerns and feedback evolve over time.

3. Summarize the sentiment distribution in prompt handling, offering a detailed breakdown of positive, neutral, and negative user feedback across features.

4. Identify specific challenges faced by users through the analysis of high-frequency bigrams and keywords, leading to actionable recommendations for improving prompt interactions.

This multi-step approach ensures a thorough understanding of user sentiment trends and underlying issues in both app performance and prompt interaction.

1. **Methodology**:

To analyze user feedback systematically, we employed a structured approach using Python libraries to extract insights from user comments. Common stopwords such as "the" and "and" were filtered out using Python's nltk library to focus on meaningful words.

Keyword Analysis: Tokenization was applied to break down the cleaned user comments into individual words. Frequencies of these words were then calculated to identify the most commonly used terms. By ranking keywords based on frequency, we were able to detect overarching themes and recurring issues in user feedback. The top 20 keywords provided a concise summary of user concerns.

Bigram Analysis: To delve deeper into the context of user feedback, we extracted bigrams, or two-word combinations, using the CountVectorizer from Python's scikit-learn library. This allowed us to analyze co-occurrences of words and identify specific phrases that commonly appeared in user comments. Stopwords were excluded during this process to focus on meaningful word pairs. The top 20 frequent bigrams highlighted critical patterns and user concerns.

Comparison Between Versions: To understand how user feedback evolved over time, we analyzed comments separately for two key app versions: before and after updates to version 1.2024.066 and 1.2023.313. Both keyword and bigram analyses were conducted for each period. By comparing results across versions, we identified shifts in user sentiment

and pinpointed areas where updates addressed previous concerns or introduced new challenges.

2. **Bigram Analysis of User Feedback Across Key App Updates**

This section examines user feedback across two significant app versions—1.2023.313 and 1.2024.066—to determine how user sentiments and feedback evolved after each update. By analyzing high-frequency bigrams extracted from user comments before and after the updates, we aimed to identify recurring themes in user satisfaction, emerging concerns, and persistent challenges. The analysis covered both overall and negative feedback to provide a comprehensive understanding of the app's impact and areas requiring further improvement. The results are as follows:

High frequency bigrams before and after version 1.2024.066

Bigram	Frequency	Bigram	Frequency
good app	1883	good app	6270
best app	1553	best app	5191
chat gpt	1164	nice app	3786
great app	1104	chat gpt	3102
nice app	958	great app	2915
love app	676	best ai	1871
best ai	671	love app	1780
amazing app	550	amazing app	1451
really good	455	useful app	1392
easy use	402	helpful app	1237
ai app	387	app good	1129
app good	374	really good	1114
useful app	364	easy use	1060
really helpful	330	really helpful	990
helpful app	317	ai app	973
app really	308	app helpful	946
voice chat	288	excellent app	900
web version	271	like app	822
excellent app	264	app useful	810
use app	260	app really	720

High frequency bigrams of negative comments before and after version 1.2024.066

Bigram	Frequency	Bigram	Frequency
chat gpt	103	chat gpt	285
phone number	68	try later	177
web version	65	wrong answer	159
good app	53	gives wrong	155
doesnt work	51	wrong answers	143
dont know	50	bad app	133
went wrong	50	dont know	131
wrong answers	48	good app	126
try later	46	doesnt work	125
wrong answer	46	worst app	122
gives wrong	44	stopped working	119
worst app	39	wrong information	116
use app	38	went wrong	103
app good	37	use app	98
error message	35	network error	94
error occurred	34	error occurred	93
voice chat	34	app doesnt	75
bad app	32	app good	72
wont let	32	keeps saying	71
stopped working	31		

For version 1.2023.313, the analysis revealed that positive feedback prominently featured phrases like "good app," "best app," and "great app," reflecting widespread user satisfaction with the app's usability and functionality. Other terms such as "easy use" and "amazing app" highlighted the app’s ability to meet user expectations in terms of accessibility and innovation. However, in the same version, negative feedback emphasized issues like "phone number," "doesn't work," and "web version," pointing to challenges in accessibility and integration with online features. After the update, new concerns emerged, such as "wrong answer," "stopped working," and "error occurred," indicating improvements in some areas but revealing persistent challenges in accuracy and system reliability.

The analysis of version 1.2024.066 echoed similar trends in positive feedback, with terms like "good app," "best app," and "nice app" recurring, suggesting sustained satisfaction with the app’s core features. Users continued to praise the app's functionality with phrases like "helpful app" and "really good." In contrast, the negative feedback for this version pointed to evolving user concerns. Before the update, frequent issues included "chat gpt," "phone number," and "web version," which signaled problems with integration and ease of use. Post-update, terms such as "wrong answer," "stopped working," and "error occurred" became more prominent, reflecting ongoing difficulties with response accuracy and system stability. These findings suggest that while certain issues were addressed, others persisted or intensified.

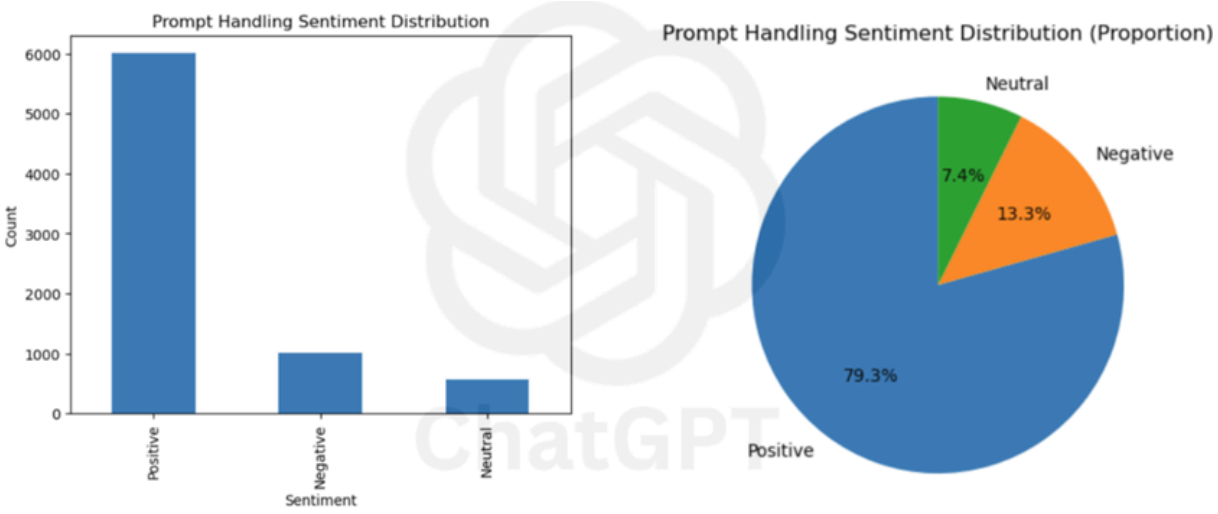
Comparing the two versions, the results highlight consistent satisfaction with core features, as indicated by the recurring positive bigrams across updates. However, shifts in negative

feedback underscore systemic challenges in areas like response accuracy and system reliability. While some earlier concerns, such as "phone number" and "web version," became less frequent in later feedback, new issues emerged, suggesting that updates introduced their own set of challenges. This underscores the importance of continuous refinement in addressing user concerns and enhancing overall app performance.

Overall, the analysis demonstrates that while updates deliver improvements in some areas, they also bring new challenges, particularly in terms of accuracy and stability. These findings provide actionable insights for guiding future app development efforts and aligning updates more effectively with user expectations.

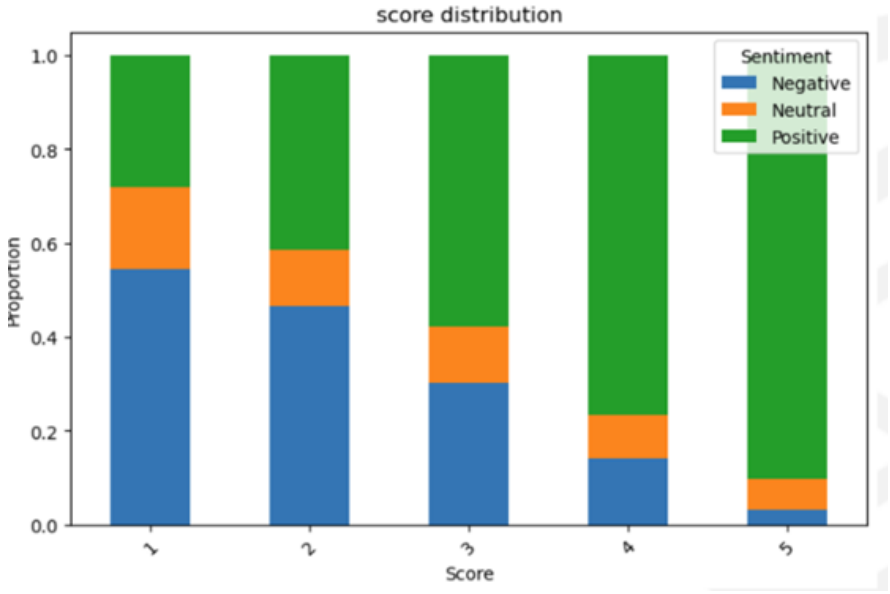
3. **Sentiment Analysis of Prompt Handling: Trends and Insights**

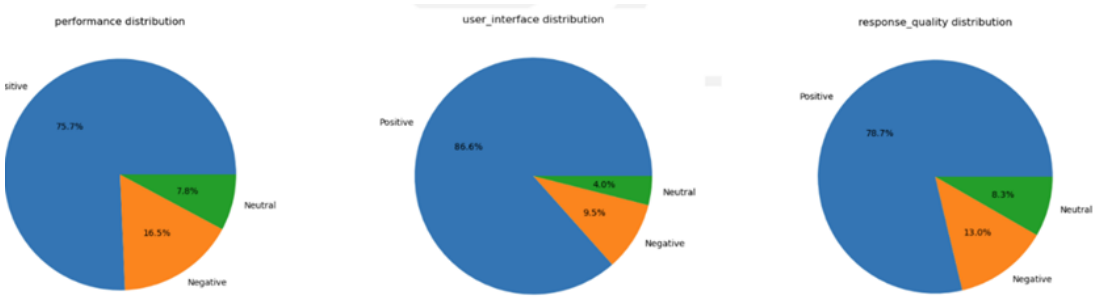
In this section, the focus shifts to exploring user sentiments specifically related to prompt handling. The analysis delves into the distribution of user feedback across various sentiment categories—Positive, Neutral, and Negative.



The first visualization, a bar chart, categorizes feedback into these three sentiment classes. It reveals a significant dominance of positive feedback, making up 79.3% of all responses. Meanwhile, negative feedback accounts for 13.3%, and neutral feedback comprises the remaining 7.4%. This distribution reflects a generally positive user experience with the system's ability to handle prompts. However, the presence of negative feedback suggests that a subset of users encounters challenges that require attention.

The subsequent analysis explores sentiment distribution across user ratings, ranging from 1 to 5. The data shows a clear correlation: higher ratings (4-5) are strongly associated with positive feedback, while lower ratings (1-2) show a significant increase in negative sentiment. This relationship underscores the system's capability to satisfy users but highlights areas of dissatisfaction among lower-rated interactions.





The final set of pie charts examines sentiment distribution across performance, user interface, and response quality. Both performance and user interface received overwhelmingly positive feedback, at 75.7% and 86.6%, respectively. These results indicate user satisfaction with these aspects of the system. However, response quality exhibited a slightly higher proportion of negative feedback (13%), hinting at areas where users perceive room for improvement in the accuracy and relevance of responses.

4. **Analysis of User Feedback on Prompt Handling Challenges**

User Interface-Related Issues

Keyword	Frequency	Bigram	Frequency
app	31	amazing app	3
question	22	answer questions	3
questions	19	app does	3
answer	19	app doesnt	2
chatgpt	17	app quite	2
ai	17	asked question	3
even	16	chat gpt	3
answers	13	chatgpt app	4
asked	13	false information	3
ask	12	input field	3
input	12	instead answering	2
use	11	mandarin written	2
good	11	paid version	2
doesnt	11	recent conversations	2
prompt	11	text input	3
one	9	view recent	2
issue	9	wont answer	2
chat	9	written sentences	3
quite	9		
responses	9		

Keyword and Bigram Frequencies for User Interface-Related Negative Comments

The analysis of user interface-related negative comments focuses on identifying recurring patterns that signal specific challenges faced by users. Key bigrams such as "mandarin written" highlight issues with Mandarin text input, reflecting significant obstacles in language support for non-English users. Similarly, the bigram "input field" suggests usability challenges within the user interface, such as poorly designed input boxes or field behavior inconsistencies. These findings emphasize the need to enhance user interface design and

functionality to cater to diverse user needs.

Response Quality-Related Issues

Keyword	Frequency	Bigram	Frequency
answer	367	answer question	56
question	279	answer questions	46
app	262	answer wrong	11
questions	247	answering questions	16
answers	168	answers questions	25
wrong	122	ask question	39
give	117	asked question	20
ask	113	chat gpt	22
chatgpt	91	correct answer	17
asked	82	doesnt answer	17
ai	81	gives wrong	14
even	80	good app	13
cant	77	question answer	25
dont	73	question asked	14
like	72	right answer	11
doesnt	72	simple question	10
good	67	waste time	10
time	64	worst app	11
response	61	wrong answer	32
gives	60	wrong answers	24

Keyword and Bigram Frequencies for Response Quality-Related Negative Comments

The analysis of response quality-related feedback uncovers critical gaps in AI performance. Prominent bigrams like "wrong answer" and "simple question" reveal that users often encounter inaccurate responses, even for straightforward prompts. These observations point to a need for refining the model's ability to comprehend and process simpler queries accurately. The data underscores the importance of enhancing response relevance and reliability to address user frustrations effectively.

Performance-Related Issues

Keyword	Frequency	Bigram	Frequency
app	46	answer hindi	4
question	24	answer questions	4
slow	20	data hai	3
answer	18	dinner scene	2
questions	17	doesnt work	3
like	14	fast answer	2
prompt	14	fastboot mode	3
doesnt	11	faster running	2
ai	11	important note	3
prompts	11	keeps crashing	3
use	11	kind slow	2
version	11	koi bhi	2
even	10	lagging times	2
chatgpt	9	message wrong	2
wrong	9	paid version	3
cant	9	running mile	2
responses	9	simple question	3
time	9	way better	2
dont	8	web version	3
giving	7	wrong answer	4

Keyword and Bigram Frequencies for Performance-Related Negative Comments

Performance feedback sheds light on system-level challenges such as speed and reliability. Notable bigrams include "fast answer," which reflects unmet expectations for quick responses, and "keeps crashing," highlighting concerns about system stability. These insights reveal that system performance issues significantly impact user satisfaction and

need prioritized attention. Improving both speed and stability is critical for delivering a smoother, more reliable user experience.

Conclusion - This analysis provides a comprehensive examination of user feedback, highlighting key challenges and insights into sentiment distribution across app versions and prompt handling performance. By comparing feedback before and after versions 1.2023.313 and 1.2024.066, the findings reveal notable improvements in user experience alongside persistent issues, particularly in areas such as response accuracy and system stability.

Sentiment analysis of prompt handling feedback shows a predominance of positive responses, reflecting overall user satisfaction. However, the presence of neutral and negative feedback underscores specific areas requiring further attention. Key challenges identified include slow response times, frequent system instability, difficulties with input functionality (e.g., language-specific issues), and inaccuracies in answering both complex and straightforward queries.

These findings emphasize the importance of addressing these challenges to enhance user experience. Future efforts should focus on improving system reliability, optimizing response accuracy, and refining user interface design to meet diverse user needs effectively. By systematically resolving these issues, the application can achieve a more seamless and satisfactory experience for its users.

Conclusion

The analysis demonstrates the power of NLP techniques in analyzing large-scale user feedback to inform app development strategies. By examining over 278,000 Android reviews of the ChatGPT app, we uncovered insights into sentiment trends, feature satisfaction, and user concerns. The findings emphasize that response quality is a core strength, consistently earning high praise, while performance, prompt handling, and updates reveal opportunities for refinement.

From research question one, we found that users frequently praise the ChatGPT app for its accurate responses, smooth performance, and user-friendly interface, while occasional criticisms focus on issues like lag, crashes, and inconsistencies in response accuracy during specific contexts.

For research question two, we identified that positive sentiment generally increased in earlier versions but began declining as expectations rose and attention shifted to advanced functionalities. Additionally, response quality and performance remain core user concerns, showing the highest levels of feedback.

In research question three, with regards to prompt handling, sentiment distribution analysis showed positive feedback, but also highlighted specific challenges, such as feedback about slow response times, system instability, input challenges, and inaccurate answers

Key takeaways include the critical role of sentiment evolution across app updates, highlighting the importance of addressing persistent issues such as **system reliability**, **response accuracy**, and **interface flexibility**. Bigram analysis provided more actionable insights into user concerns, identifying recurring challenges with prompt handling and updates.

The study also acknowledges limitations, including

1. not all non-English text was effectively removed;

2. VADER Sentiment Analyzer unable to accurately detect the accurate meaning of phrases or emojis, such as "🔥 " and "mind blowing";

3. Using custom researched lists of keywords as a crude method of text labeling; and

4. Many of the top keyword and bigrams appearances were not as meaningful.

By addressing these challenges and leveraging feedback effectively, the ChatGPT app can further optimize user satisfaction and maintain its competitive edge in the conversational AI market.

References:

1. Hutto, C., & Gilbert, E. (2014). VADER: A Parsimonious Rule-Based Model for Sentiment Analysis of Social Media Text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1), 216-225. Retrieved from https://ojs.aaai.org/index.php/ICWSM/article/view/14550

2. Jurafsky, D., & Martin, J. H. (2019). *Speech and Language Processing* (3rd ed.). Pearson. Retrieved from https://web.stanford.edu/~jurafsky/slp3/

3. Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching Word Vectors with Subword Information. *Transactions of the Association for Computational Linguistics*, 5, 135–146. Retrieved from https://aclanthology.org/Q17-1010/

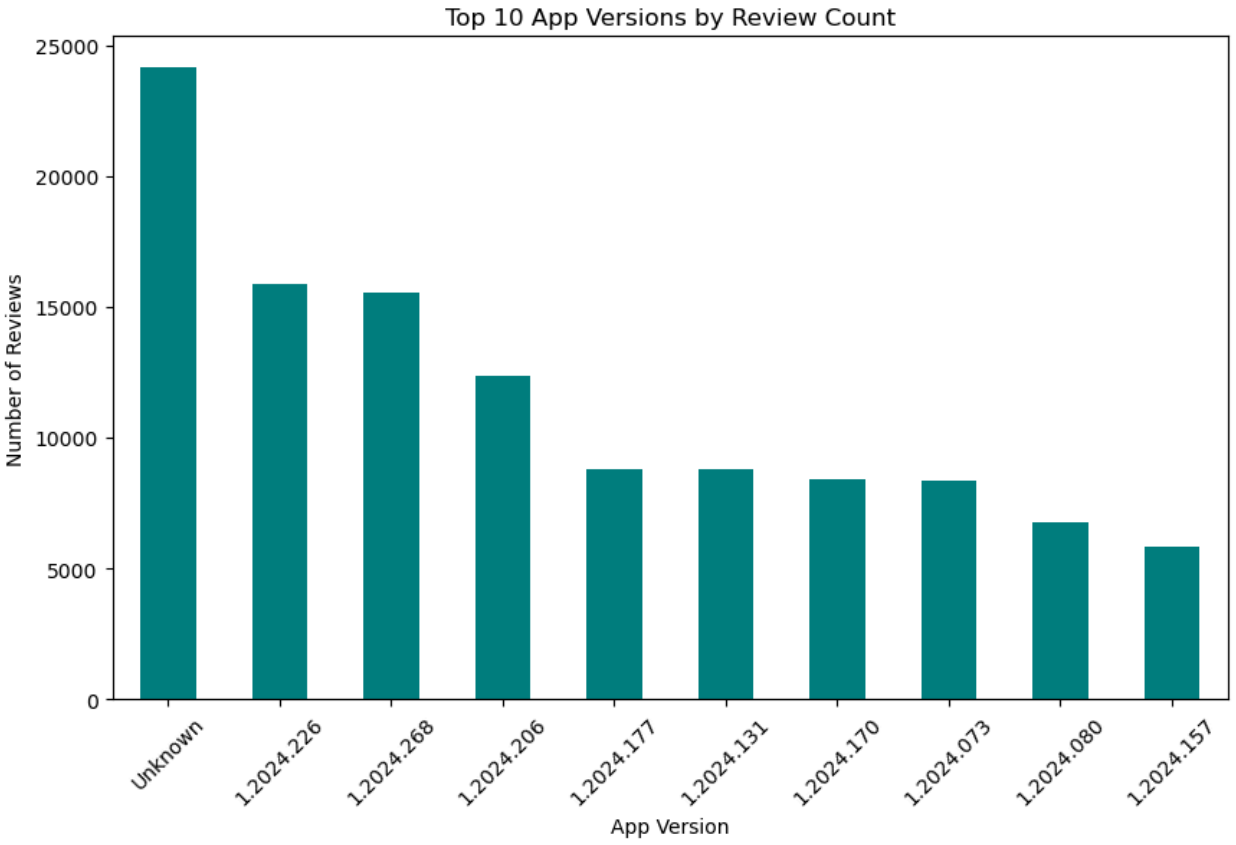
4. Dataset Source - Kumar, A. (2024). *ChatGPT Reviews [Daily Updated]*. Kaggle. Retrieved from Kaggle https://www.kaggle.com/datasets/ashishkumarak/chatgpt-reviews-daily-updated

Appendix

1. Snippet of source data (kaggle):

reviewId id	userName reviewer	content review	score rating	thumbsUpCount likes	reviewCreatedV... app version	at date	appVersion app version
278337 unique values	255900 unique values	good 5% nice 2% Other (265335) 94%	1 5	0 1193	[null] 9% 1.2024.268 6% Other (242288) 86%	2023-07-25 2024-11-28	[null] 9% 1.2024.268 6% Other (242288) 86%
42ad3464-5158-485f-a745-51058db2bdc3	Pavia Akter	that's amazing ,its help me ,and i am happy to used it	5	0	1.2024.324	2024-11-28 16:20:45	1.2024.324
d638be9c-18a1-42e6-b363-aae43c0ef53b	Yubraj Singh	it's really Good 👍	5	0	1.2024.324	2024-11-28 16:18:37	1.2024.324
273b8d07-a3db-4ac0-9166-9a043c8e3eb8	Hemanth 1219	very very very excellent 👌	4	0	1.2024.324	2024-11-28 16:17:55	1.2024.324
2b2244bb-1e0e-408e-bc8b-d5cddd8849de	RITESH Das	The best ai app	1	0	1.2024.324	2024-11-28 16:17:16	1.2024.324
e263b34a-545e-4ade-b423-c1ce8dd322a7	Aashiya Khan	it's very helpful	3	0	1.2024.324	2024-11-28 16:17:12	1.2024.324
f6161614-58c2-4397-9473-35703dc8d387	Moiz	it is good and helps me alot in tasks	4	0	1.2024.324	2024-11-28 16:17:06	1.2024.324
685a86cf-5588-408f-873c-9c951bc10f66	Sidhu moose wala	very good app	4	0	1.2024.324	2024-11-28 16:16:57	1.2024.324
c14eddfd-2ac2-48e3-a326-c754dfa2a396	Tanbir Ahmed saikat	nc apo	5	0	1.2024.324	2024-11-28 16:16:52	1.2024.324
c452c5ff-e151-4fe9-adb7-abadcffb6558	Chiku	one of my favorite friend	5	0	1.2024.310	2024-11-28 16:16:34	1.2024.310
aa91a3ea-24e3-44f9-80bf-ffc2c5142c5b	Khadiga Mohamed	جميل ادي ادي ادي	5	0	1.2024.310	2024-11-28 16:16:00	1.2024.310
2527d2c8-c4fe-4182-8b16-a77a17799340	Kenes Syiem	Thanks for the apps it was great to use and whatever I want to search it i found very helpful thank ...	5	0	1.2024.317	2024-11-28 16:15:51	1.2024.317

2. Data Visualization - to see the number of reviews in each version.



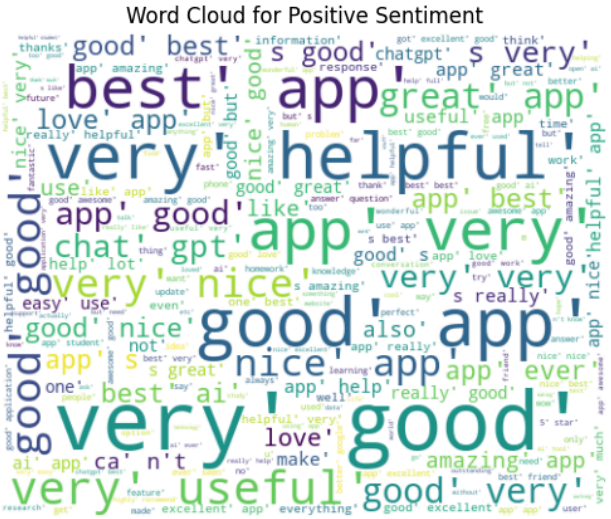
3. Cleaned and Processed Data Dictionary

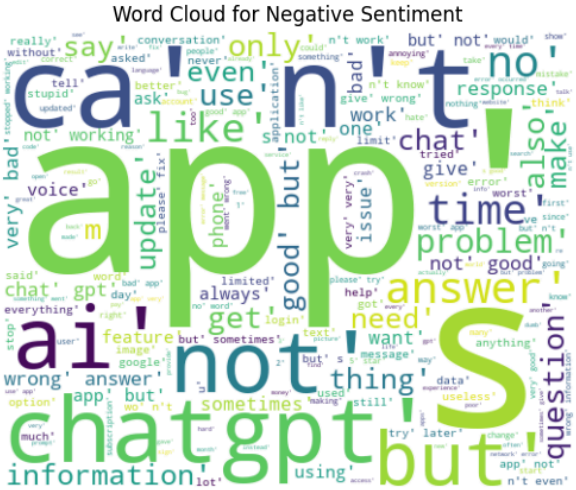
- **reviewId -** Unique Review ID
- **userName -** Username of the reviewer
- **comment -** The textual content of the review
- **score -** star rating that the reviewer posted (1 = lowest, 5 = highest)
- **date_time -** date and time of the review posted
- **appVersion** - The version of the app being reviewed
- **date** - date of the review
- **time** - timestamp of the review
- **Score_sentiment** - Rating >= 4: Positive; Rating == 3: Neutral; Rating <= 2: Negative
- **tokens** - Words in their original form without any lemmatization or stemming
- **Cleaned_tokens** - undergone additional preprocessing for analysis
- **Text Sentiment** - pos, neu, neg and standardized compound score derived from Sentiment Analyzer for text only
- **Emoji Sentiment** - pos, neu, neg and standardized compound score derived from Sentiment Analyzer for emoji only
- **Comment Sentiment** - pos, neu, neg and standardized compound score derived from Sentiment Analyzer for the overall comment

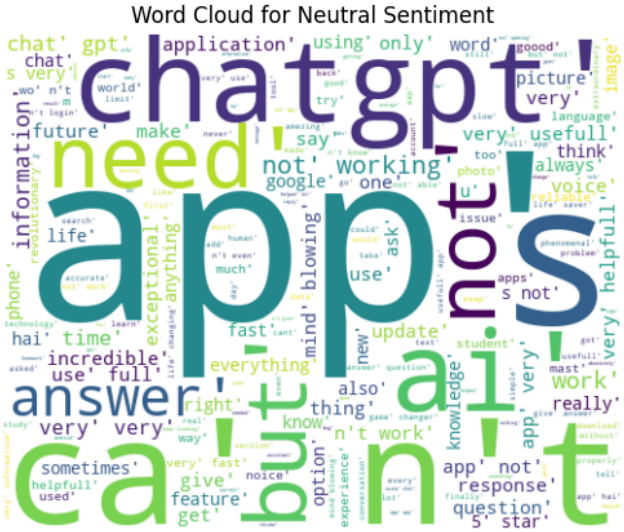
4. Code snippet of allocating relevant keywords to each feature.

```python
features = {'performance': ['performance','accurate',
'faster', 'freeze', 'slow',
'crash','loading','delay','smooth','easy','speed','responsive'
,'lag','frozen','hang','buffering','glitch'],

  'user_interface':['ui','notification','alert','voice','dicp
ate','recording','recognition','userinterface','responsive','d
ropdown','customize','style','ux','interface', 'intuitive',
'design','friendly',
'navigation','buttons','button','menu','icons','theme','visual
','look','feel','front','dark mode'],

  'response_quality': ['correct', 'response', 'language',
'accuracy',
'answer','output','insightful','incorrect','irrelevant','misle
ading','ambiguous','inaccurate','vague','informative','detaile
d','understandable','understand'],

    'prompt_handling':['prompt', 'question', 'input',
'command','generate','clarify','interpret','execution','instru
ction','query','prompt
engineering','misinterpret','retry','feedback','understands'],

   'updates': [' fix', 'update', 'updates','updated','bug',
'error','improvement','beta','agreed','improved','version','mo
dify','modified','resolve','resolved','outdated','patch','supp
ort','issues']}
```

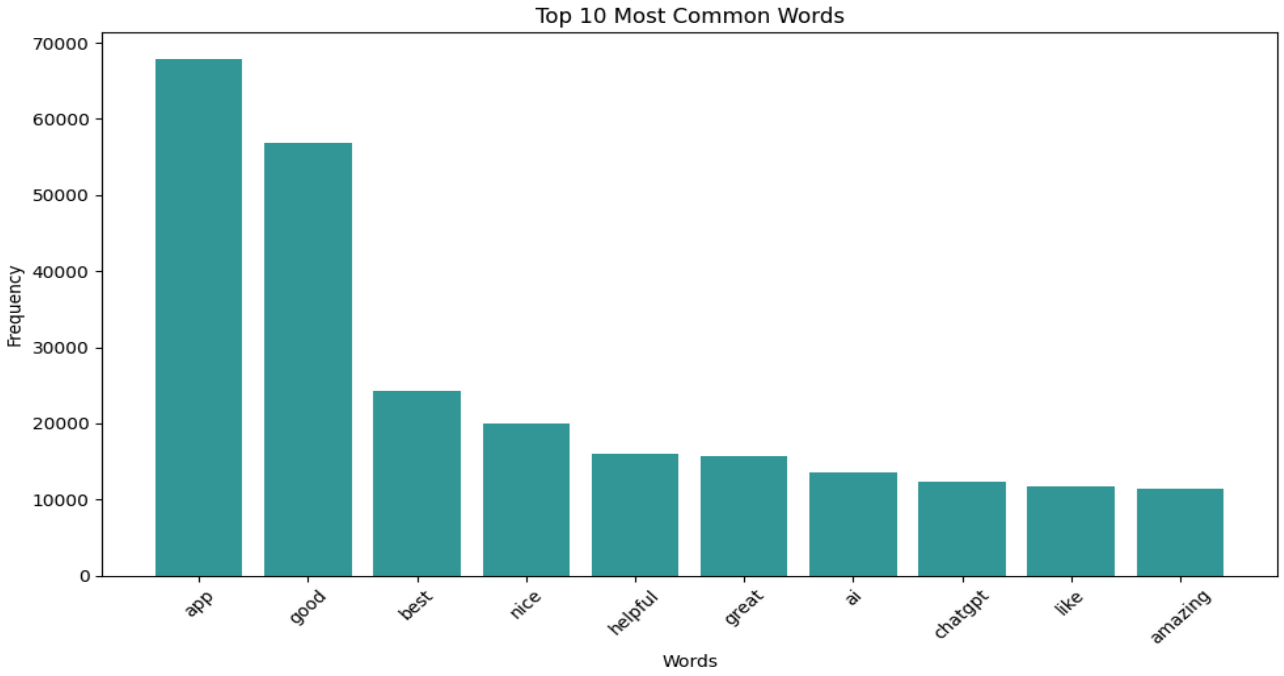
5. Data Visualization - Wordclouds for positive, negative and neutral sentiments:



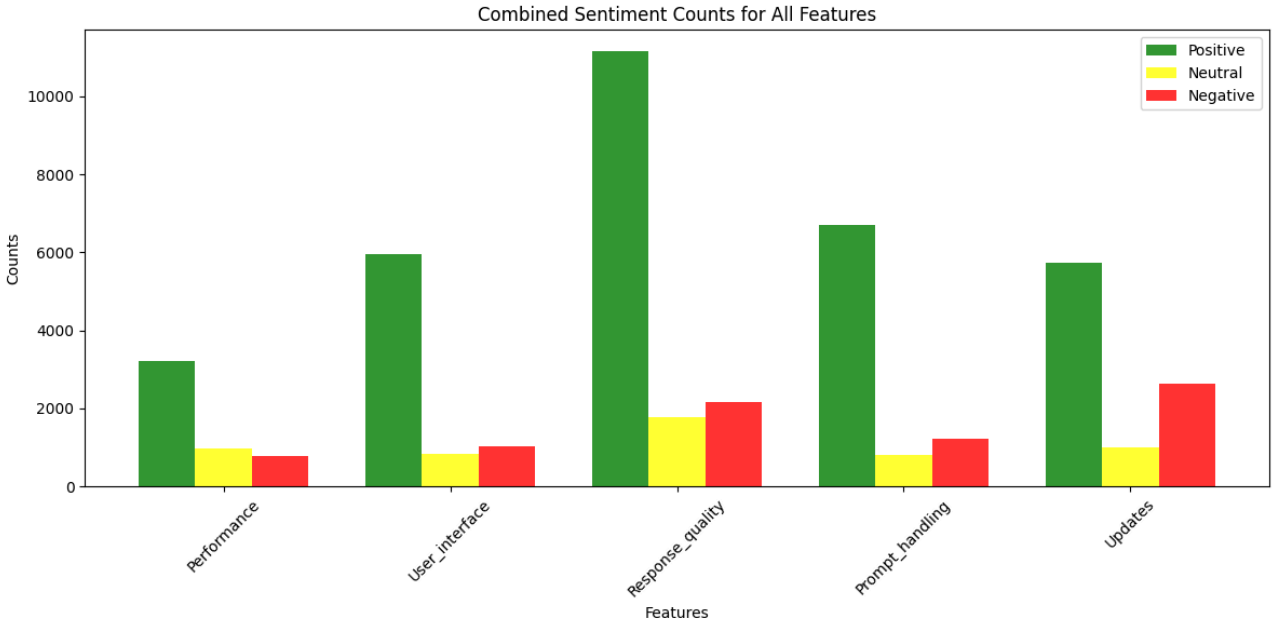


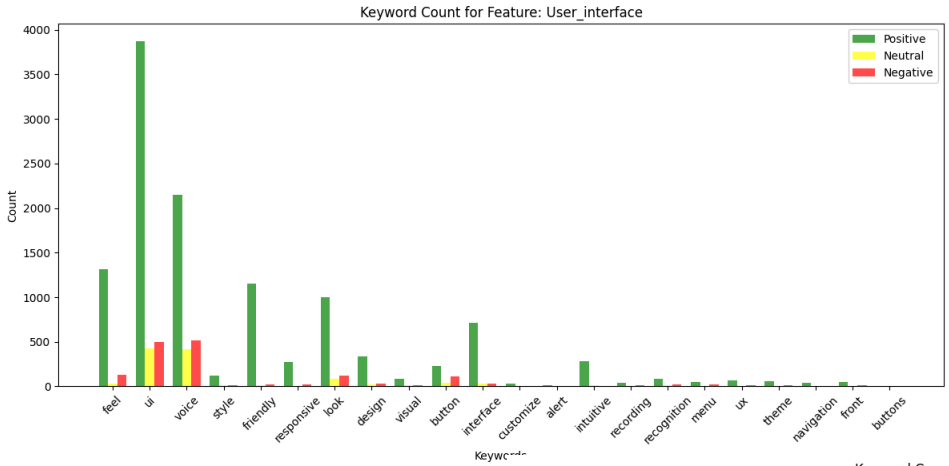


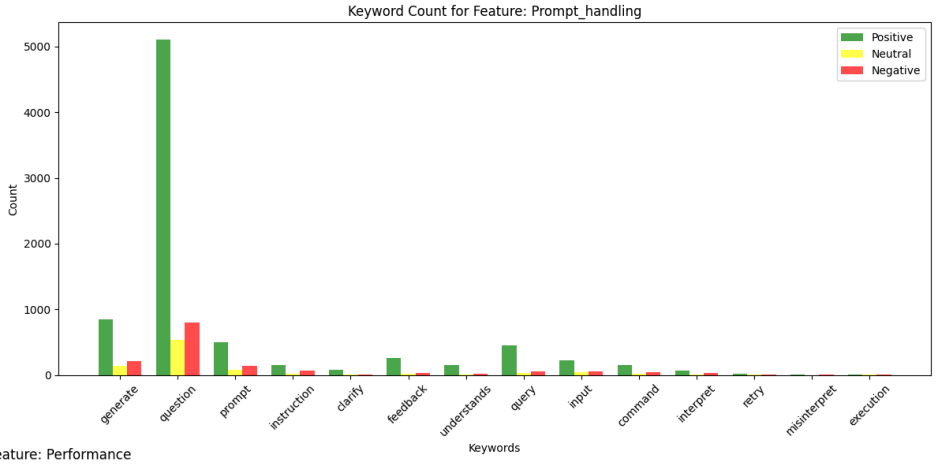
6. Data Visualization - Top 10 Most Common Words:

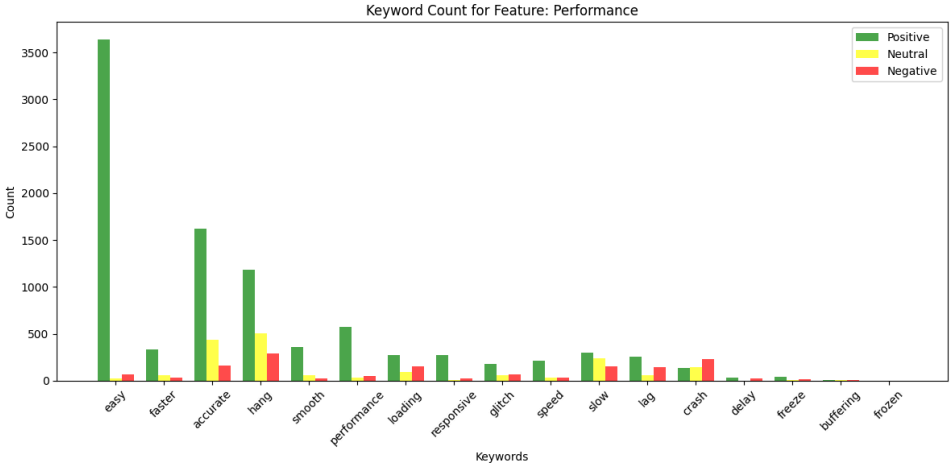


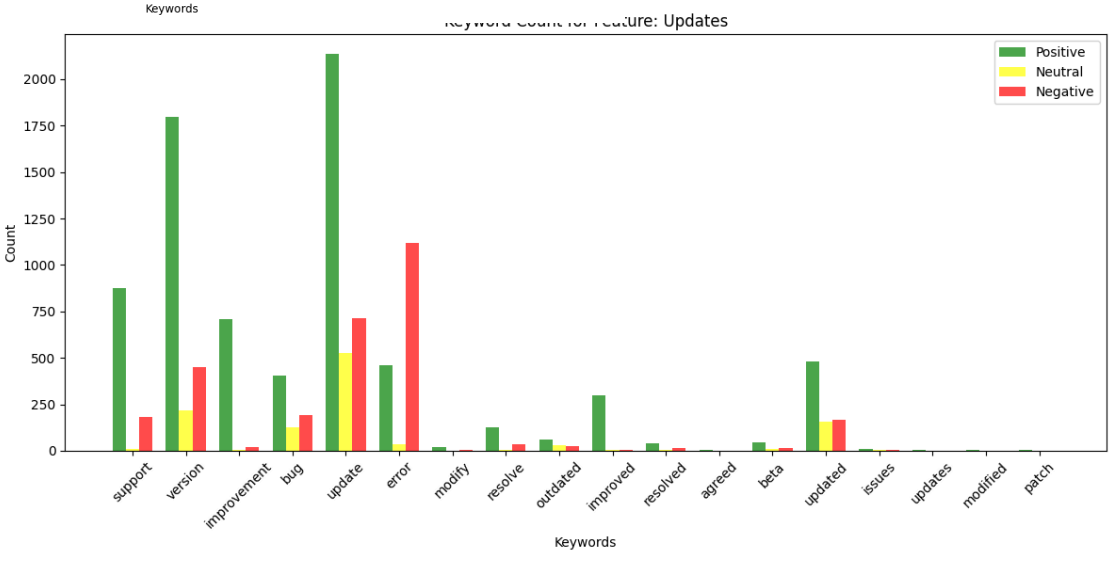
7. Data Visualization of selected features and its sentiment trend:




Keyword Count for Feature: User_interface
Positive
Neutral
Negative
Count
feel
ui
voice
style
friendly
responsive
look
design
visual
button
interface
customize
alert
intuitive
recording
recognition
menu
ux
theme
navigation
front
buttons


Keyword Count for Feature: Prompt_handling
Positive
Neutral
Negative
Count
generate
question
prompt
instruction
clarify
feedback
understands
query
input
command
interpret
retry
misinterpret
execution
Keywords


Keyword Count for Feature: Performance
Positive
Neutral
Negative
Count
easy
faster
accurate
hang
smooth
performance
loading
responsive
glitch
speed
slow
lag
crash
delay
freeze
buffering
frozen
Keywords


Positive
Neutral
Negative
Count
support
version
improvement
bug
update
error
modify
resolve
outdated
improved
resolved
agreed
beta
updated
issues
updates
modified
patch
Keywords