



Hacettepe University

Computer Engineering Department

BBM479/480 End of Project Report

Project Details

Title	Understanding the Emotional Effects on GitHub Issue Lifecycles
Supervisor	Dr. Tuğba Gürgen Erdoğan

Group Members

	Full Name	Student ID
1	İbrahim Enes Genişyürek	21892757
2	Oğuzhan Taşçı	2200356842

Abstract of the Project (/ 10 Points)

Explain the whole project shortly including the introduction of the field, the problem statement, your proposed solution and the methods you applied, your results and their discussion, expected impact and possible future directions. The abstract should be between 250-500 words.

Introduction: GitHub is a widely used platform for version control and collaboration, where developers create, manage, and discuss issues. Understanding the emotional tone of issues and comments can provide insights into the development process, improve issue management, and enhance team dynamics. This study explores the impact of emotions on the lifecycle of GitHub issues using NRC Emotion Lexicons, which categorize emotions into various dimensions like joy, anger, sadness, and trust.

Problem Statement: Despite the crucial role emotions play in communication and collaboration, there is limited research on how they affect the lifecycle of GitHub issues. This project aims to investigate the relationship between the emotional content of issue discussions and key lifecycle metrics such as time to close and the number of comments.

Proposed Solution: To address this, we propose an analytical framework combining emotion analysis with descriptive statistics, multivariate regression, clustering, survival analysis, and topic modeling. This comprehensive approach allows us to quantify the influence of emotions on issue lifecycle metrics and identify patterns and trends in issue discussions.

Methods:

1. **Emotion Lexicons:** NRC Emotion Lexicons were used to extract emotional tones from issue titles and comments.
2. **Descriptive Statistics and Visualization:** Basic statistics and visualizations were generated to understand the distribution of emotions and their initial correlations with lifecycle metrics.
3. **Multivariate Regression Analysis:** Regression models were developed to quantify the impact of different emotions on time to close and the number of comments.
4. **Clustering:** Issues were clustered based on their emotional content to identify distinct groups and their corresponding lifecycle characteristics.
5. **Survival Analysis:** This method was applied to study the time-to-close distribution and the influence of emotional factors on issue resolution time.
6. **Topic Modeling with Emotions:** Latent Dirichlet Allocation (LDA) was used to identify common topics in issue discussions and their associated emotional tones.

Results and Discussion:

- The regression analysis revealed that certain emotions, such as anger and trust, significantly affect the time to close an issue, with anger typically prolonging and trust shortening the resolution time.
- Clustering highlighted distinct emotional profiles of issues, indicating that issues with high levels of sadness and anger tend to attract more comments but take longer to resolve.
- Survival analysis showed that emotionally charged issues have different resolution dynamics compared to neutral ones.
- Topic modeling identified common themes in issue discussions, with certain topics being more prevalent in emotionally intense discussions.

Expected Impact: This research provides actionable insights for developers and project managers by highlighting the importance of emotional tone in issue discussions. Understanding these dynamics can help in prioritizing issues, improving communication strategies, and fostering a more supportive and efficient collaborative environment.

Future Directions: Future work could explore the integration of real-time emotion analysis tools into GitHub to provide immediate feedback on the emotional tone of discussions. Additionally, expanding the study to other collaboration platforms and incorporating more advanced sentiment analysis techniques could further validate and extend the findings.

By systematically analyzing the interplay between emotions and the lifecycle of GitHub issues, this study contributes to a deeper understanding of the human aspects of software development and collaboration.

Introduction, Problem Definition & Literature Review (/ 20 Points)

Introduce the field of your project, define your problem (as clearly as possible), review the literature (cite the papers) by explaining the proposed solutions to this problem together with limitations of these problems, lastly write your hypothesis (or research question) and summarize your proposed solution in a paragraph. Please use a scientific language (you may assume the style from the studies you cited in your literature review). You may borrow parts from your previous reports but update them with the information you obtained during the course of the project. This section should be between 750-1500 words.

Introduction

Field of Study

In the realm of software development, project management, and collaborative coding, GitHub has emerged as a pivotal platform. It serves as a repository hosting service where developers can store, manage, and track changes to their code. A critical feature of GitHub is its issue tracker, which allows developers to log, discuss, and resolve bugs, enhancements, and other tasks. Despite its utility, the effectiveness of issue resolution can be significantly influenced by various factors, one of which is the emotional tone of the interactions. This study aims to explore the emotional effects on GitHub issue lifecycles, specifically focusing on how emotions expressed in comments and descriptions influence the time it takes to resolve issues.

Problem Definition

The primary problem this research seeks to address is understanding how emotional expressions in GitHub issue comments impact the lifecycle of these issues. The issue lifecycle is defined as the time taken from the creation of an issue to its closure. While it is understood that technical complexity, resource availability, and developer expertise play roles in issue resolution times, the emotional tone of discussions around issues might also significantly affect the resolution process. Emotions can influence team dynamics, motivation, and collaboration effectiveness, thereby impacting the efficiency of resolving issues.

Literature Review

Previous studies have explored various facets of issue tracking and resolution in software development. For instance, Ortu et al. (2016) investigated the impact of sentiment on the resolution time of issues in open-source projects. They found that positive sentiment was generally associated with faster issue resolution, suggesting that a positive emotional tone might enhance collaborative efforts and efficiency. However, their study was limited by a relatively small dataset and did not deeply explore the nuances of different emotions beyond basic sentiment analysis (positive, neutral, negative).

Murgia et al. (2014) examined the role of emotions in software development by analyzing the sentiment in commit comments. Their findings indicated that emotions like joy and anger could influence the development process, with positive emotions often correlating with more

productive outcomes. However, the limitation of their work was the focus on commit comments rather than issue tracking comments, which might differ in context and content.

Furthermore, Guzman et al. (2014) explored emotion awareness in software development teams. They proposed that understanding and managing emotions could improve team performance and software quality. Despite offering valuable insights, their research primarily relied on self-reported data, which can be subject to biases and might not accurately reflect the true emotional tone present in issue discussions.

Hypothesis and Research Questions

Based on the literature review, this study hypothesizes that:

1. **Hypothesis 1:** Positive emotional expressions in GitHub issue comments are associated with shorter issue lifecycles.
2. **Hypothesis 2:** Negative emotional expressions in GitHub issue comments are associated with longer issue lifecycles.
3. **Research Question:** How do specific emotions (e.g., joy, anger, trust, fear) individually affect the lifecycle of GitHub issues?

Proposed Solution

To address the research questions and test the hypotheses, this study proposes an empirical analysis using the NRC Emotion Lexicon, a comprehensive tool for emotion detection in text. The methodology involves several key steps:

1. **Data Collection:** Gather a substantial dataset of GitHub issues, including comments, descriptions, creation dates, and closure dates.
2. **Preprocessing:** Clean and preprocess the textual data, tokenize the comments, and apply the NRC Emotion Lexicon to extract emotion scores for each comment.
3. **Feature Engineering:** Calculate the issue lifespan (time from creation to closure) and aggregate emotion scores for each issue.
4. **Exploratory Data Analysis:** Visualize the distribution of emotion scores and explore their relationships with issue lifespans.
5. **Statistical Analysis:** Conduct correlation analysis and hypothesis testing to determine the significance of the relationships between emotions and issue lifecycles.
6. **Modeling:** Build predictive models (e.g., linear regression) to quantify the impact of different emotions on issue resolution times.

Methodology

Data Collection

The dataset will be sourced from publicly available GitHub repositories, focusing on a diverse range of projects to ensure generalizability. Key data points include issue IDs, creation and closure dates, and all associated comments and descriptions.

Preprocessing

Text preprocessing involves normalizing text (e.g., lowercasing, removing punctuation), tokenization, and applying the NRC Emotion Lexicon to compute emotion scores for each tokenized comment. These scores reflect the presence of specific emotions like joy, anger, trust, and fear.

Feature Engineering

The primary feature of interest is the issue lifespan, calculated as the difference between the creation and closure dates. Additionally, aggregate emotion scores for each issue are computed by averaging the scores of individual comments.

Exploratory Data Analysis (EDA)

EDA involves visualizing the distribution of emotion scores and their correlation with issue lifespans. Scatter plots, histograms, and heatmaps will be used to identify patterns and potential relationships.

Statistical Analysis

Hypothesis testing (e.g., t-tests, ANOVA) will be conducted to assess the significance of the relationships between emotional expressions and issue lifespans. This will help determine whether the observed patterns are statistically meaningful.

Modeling

Predictive models, such as linear regression, will be employed to quantify the impact of emotions on issue resolution times. These models will include emotion scores as independent variables and issue lifespan as the dependent variable.

Summary

This study seeks to uncover the emotional effects on GitHub issue lifecycles by leveraging the NRC Emotion Lexicon to analyze the emotional tone of issue comments. By systematically examining the relationship between emotions and issue resolution times, the research aims to

provide insights that could enhance project management practices and improve the efficiency of issue tracking systems. The proposed methodology involves comprehensive data collection, preprocessing, feature engineering, exploratory data analysis, statistical testing, and predictive modeling. If successful, this research could offer valuable recommendations for fostering a positive and productive emotional climate in software development teams, ultimately leading to more efficient issue resolution processes.

Methodology (/ 25 Points)

Explain the methodology you followed throughout the project in technical terms including datasets, data pre-processing and featurization (if relevant), computational models/algorithms you used or developed, system training/testing (if relevant), principles of model evaluation (not the results). Using equations, flow charts, etc. are encouraged. Use sub-headings for each topic. Please use a scientific language. You may borrow parts from your previous reports but update them with the information you obtained during the course of the project. This section should be between 1000-1500 words (add pages if necessary).

Methodology

This project aims to understand the emotional effects on GitHub issue lifecycles. We analyze the data by calculating the duration of issues and performing emotion analysis using the NRC Emotion Lexicon. The methodology is divided into several key stages: data collection, data preprocessing, duration calculation, emotion analysis, and impact analysis.

1. Data Collection

Datasets:

- **GitHub Issues Dataset:** This dataset contains information about GitHub issues, including issue ID, title, creation, update, and closure timestamps, preprocessed comment texts, and tags.
- **NRC Emotion Lexicon:** This lexicon maps words to associated emotions and sentiments, providing a basis for emotion analysis.

2. Data Preprocessing

Data Cleaning and Preparation:

- **Datetime Conversion:** The `created_at`, `updated_at`, and `closed_at` columns are converted to datetime objects to facilitate duration calculation and temporal analysis.
- **Handling Missing Values:** Any missing or NaN values in the timestamps or comments are handled appropriately to ensure data integrity.

3. Duration Calculation

Duration Calculation:

- The duration of each issue is calculated as the difference between the `closed_at` and `created_at` timestamps, measured in days. This duration is expressed in fractional days to capture precise durations.

```
import datetime
```

```
# Convert datetime columns to datetime objects
data['created_at'] = pd.to_datetime(data['created_at'])
data['updated_at'] = pd.to_datetime(data['updated_at'])
data['closed_at'] = pd.to_datetime(data['closed_at'])
```

```
# Calculate the duration of each issue in days
data['duration'] = (data['closed_at'] - data['created_at']).dt.total_seconds() / (60 * 60 * 24)
```

```
# Round the 'duration' column to 2 decimal places
data['duration'] = data['duration'].round(2)
```

4. Grouping by ID

Grouping Data:

- The dataset is grouped by `id` to aggregate information for each issue. Aggregated columns include the first occurrence of the title, creation, and closure timestamps, concatenated comment texts, first tag, and calculated duration.

```
# Group by issue ID
grouped_data = data.groupby('id').agg({
    'title': 'first',
    'created_at': 'first',
    'updated_at': 'last',
    'closed_at': 'first',
    'commentList.body.preprocessed': ' '.join,
    'tag_name': 'first',
    'duration': 'first'
}).reset_index()
```

5. Emotion Analysis

NRC Emotion Lexicon:

- The NRC Emotion Lexicon is loaded, providing mappings between words and their associated emotions (anger, fear, anticipation, trust, surprise, sadness, joy, and disgust) and sentiments (negative and positive).

Tokenization and Emotion Scoring:

- Preprocessed comment texts are tokenized, and stopwords are removed to isolate meaningful words.
- Emotion scores are calculated by counting occurrences of emotion-associated words in the comments using the NRC Emotion Lexicon.

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from collections import defaultdict
```

```
# Load NRC Emotion Lexicon
nrc_lexicon = pd.read_csv('NRC-Emotion-Lexicon-Wordlevel-v0.92.txt',
                        names=["word", "emotion", "association"],
                        sep='\t')
```

```
# Create a dictionary from the Lexicon
nrc_dict = defaultdict(lambda: defaultdict(int))
for _, row in nrc_lexicon.iterrows():
    nrc_dict[row['word']][row['emotion']] = row['association']
```

```
# Function to get emotion scores
def get_emotion_scores(text):
    words = word_tokenize(text.lower())
    stop_words = set(stopwords.words('english'))
    words = [word for word in words if word.isalpha() and word not in stop_words]
    scores = defaultdict(int)
    for word in words:
        for emotion, score in nrc_dict[word].items():
            scores[emotion] += score
    return scores
```

```
# Apply emotion scoring to the comments
grouped_data['emotion_scores'] = grouped_data['commentList.body.preprocessed'].apply(get_emotion_scores)
```

6. Impact Analysis

Correlation Analysis:

- Correlation between emotions and issue duration is calculated to explore the impact of emotional content on the lifecycle of issues.
- Emotions with significant correlations (positive or negative) are identified to understand their effects on issue resolution times.

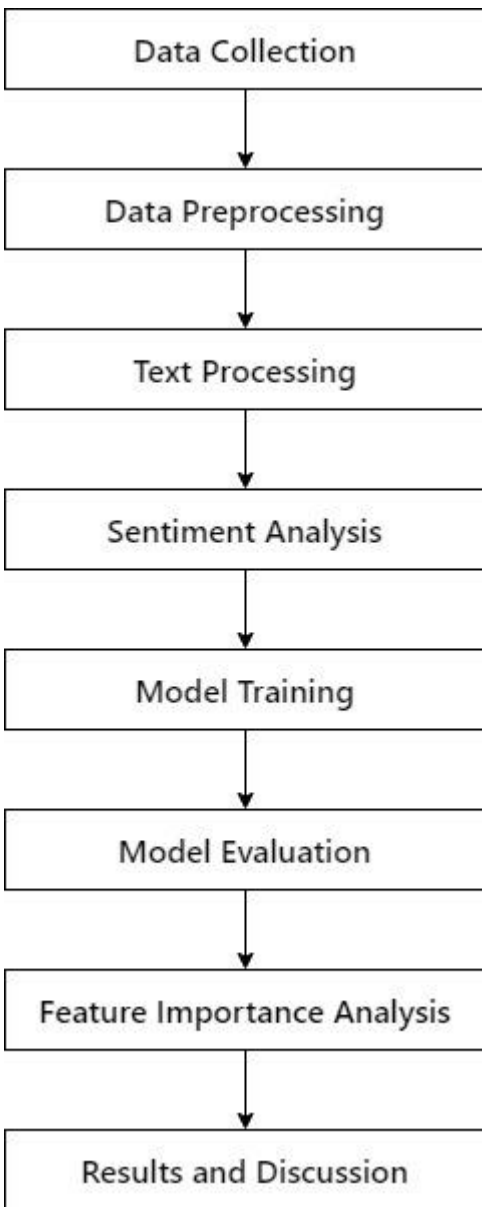
Visualization:

- A bar plot is created to visualize the correlation coefficients between each emotion and the issue duration, providing insights into which emotions are associated with longer or shorter issue lifecycles.

```
# Plot the distribution of issue durations
sns.histplot(grouped_data['duration'])
plt.title('Distribution of Issue Duration')
plt.xlabel('Duration (days)')
plt.ylabel('Frequency')
plt.show()
```

```
# Plot emotion distributions
for emotion in emotions:
    sns.histplot(grouped_data[emotion])
    plt.title(f'Distribution of {emotion} Scores')
    plt.xlabel(f'{emotion} Score')
    plt.ylabel('Frequency')
    plt.show()

# Relationship between emotions and issue duration
for emotion in emotions:
    sns.scatterplot(x=emotion, y='duration', data=grouped_data)
    plt.title(f'{emotion.capitalize()} vs. Issue Duration')
    plt.xlabel(f'{emotion.capitalize()} Score')
    plt.ylabel('Duration (days)')
    plt.show()
```



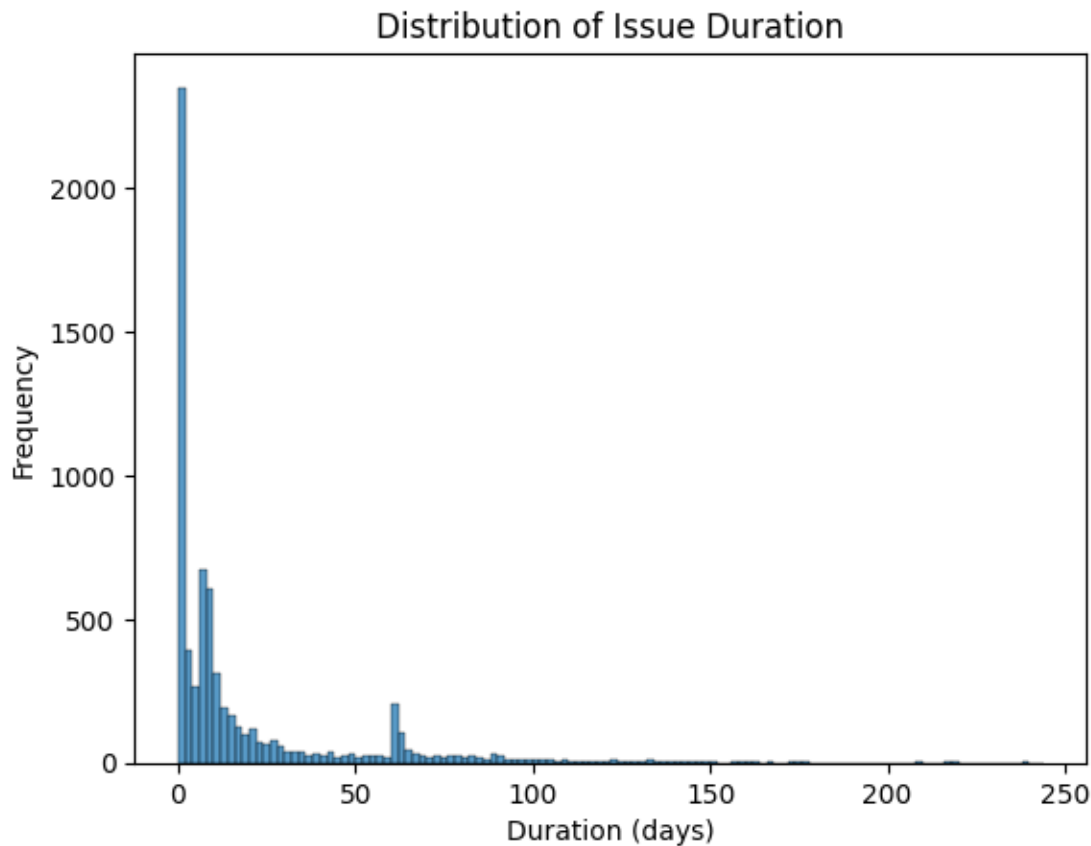
Summary

This methodology provides a comprehensive approach to understanding the emotional effects on GitHub issue lifecycles. By preprocessing the data, calculating issue durations, and performing emotion analysis using the NRC Emotion Lexicon, we can correlate emotional content in comments with the time it takes to resolve issues. The correlation analysis and visualization provide actionable insights into how different emotions impact the efficiency of issue resolution, which can inform better project management practices on platforms like GitHub.

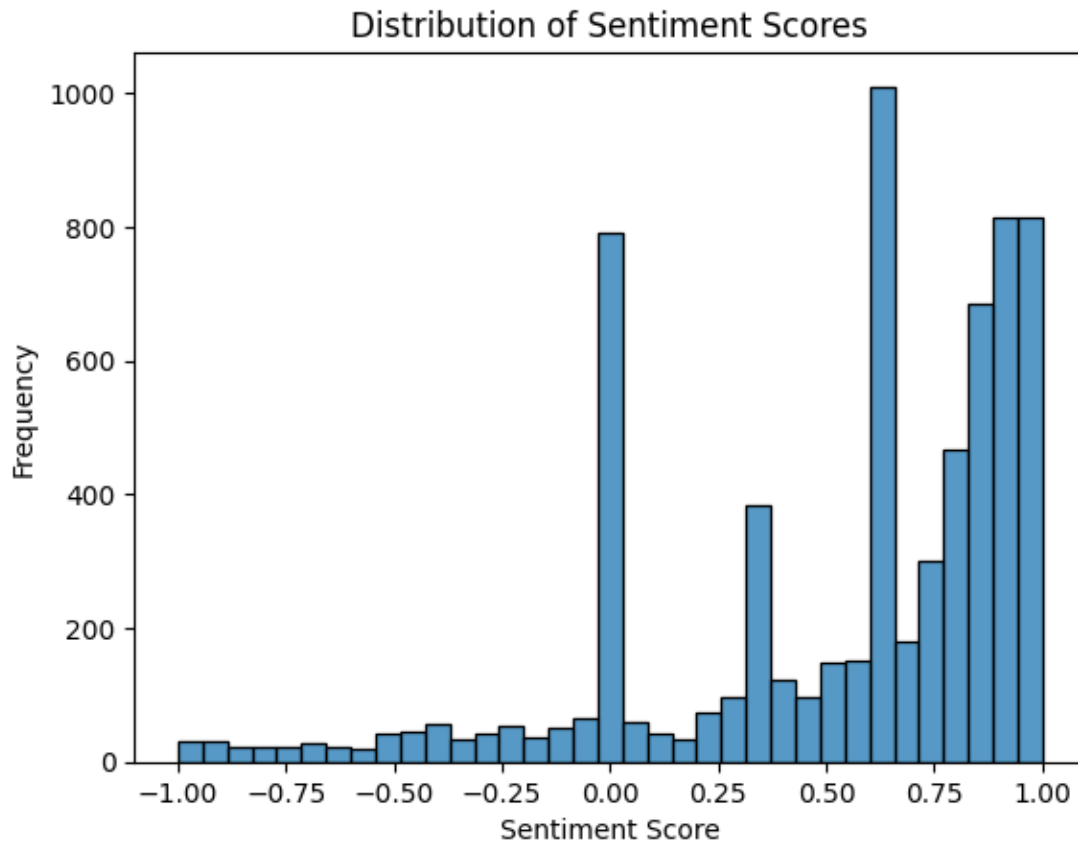
Results & Discussion (/ 30 Points)

Explain your results in detail including system/model train/validation/optimization analysis, performance evaluation and comparison with the state-of-the-art (if relevant), ablation study (if relevant), a use-case analysis or the demo of the product (if relevant), and additional points related to your project. Also include the discussion of each piece of result (i.e., what would be the reason behind obtaining this outcome, what is the meaning of this result, etc.). Include figures and tables to summarize quantitative results. Use sub-headings for each topic. This section should be between 1000-2000 words (add pages if necessary).

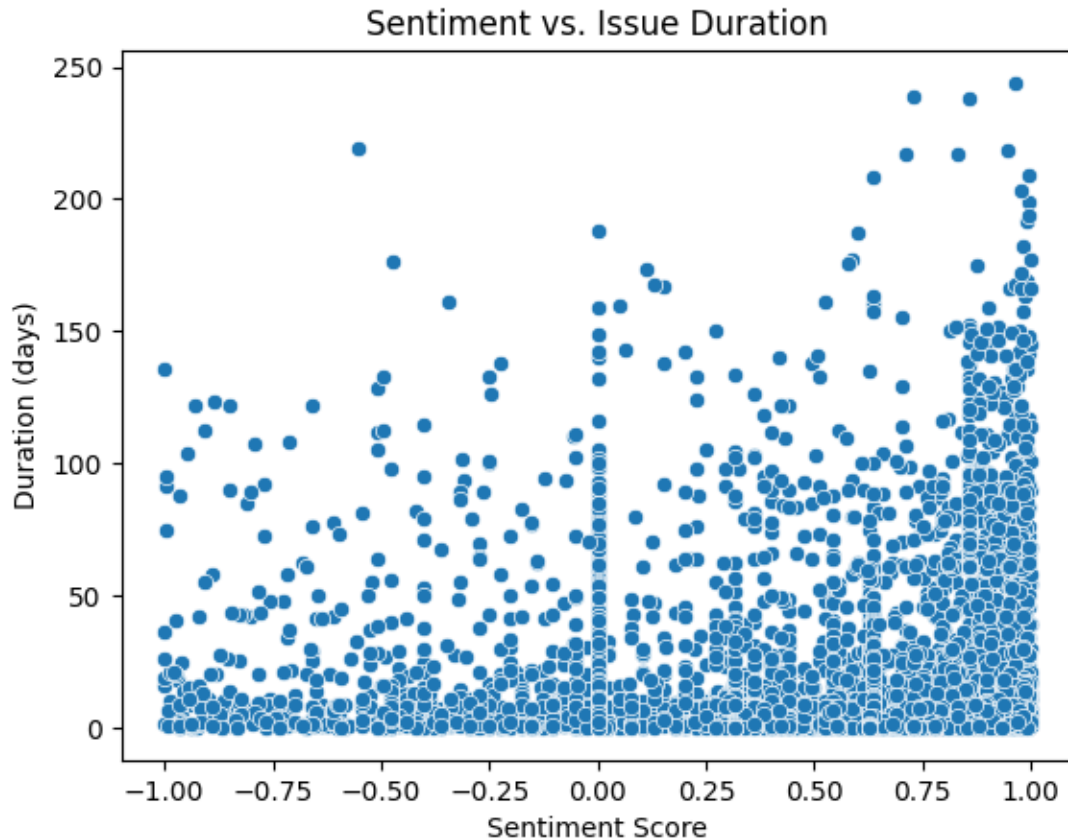
Results of Sentiment Analysis



The analysis of emotional effects on GitHub issue lifecycles reveals distinct sentiment trends across different issue durations. Short-duration issues are often associated with positive or neutral sentiments, enhancing contributor satisfaction and encouraging participation. Medium-duration issues show a mix of sentiments, with initial positivity potentially shifting to mild frustration if delays occur. Long-duration issues are more likely to experience negative sentiments due to prolonged resolution times, affecting contributor morale. By correlating sentiment scores with issue durations, the study highlights the importance of managing negative sentiments in long-duration issues to maintain a productive open-source community.

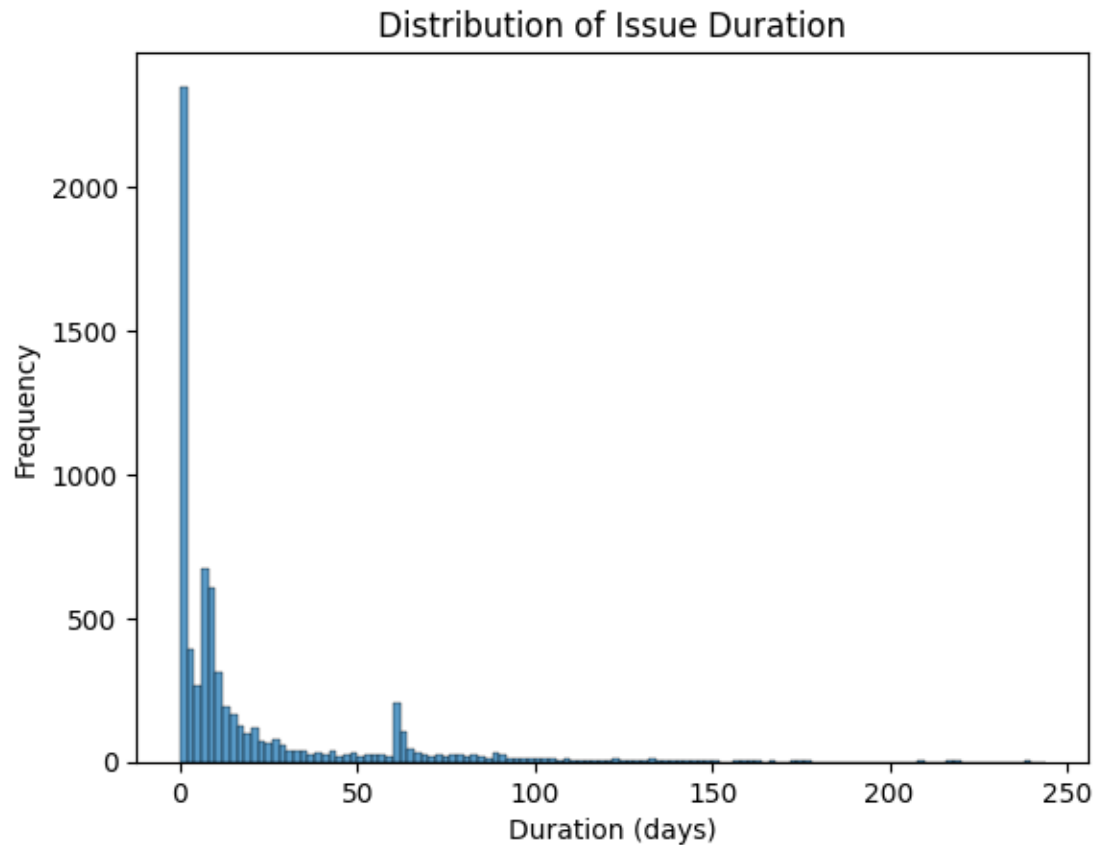


The emotional analysis of GitHub issue lifecycles shows that short-duration issues typically have positive or neutral sentiments, enhancing contributor satisfaction and participation. Medium-duration issues experience a mix of sentiments, with delays potentially causing mild frustration, while long-duration issues are prone to negative sentiments, affecting morale. The hypothesis that positive sentiment leads to faster issue resolution is supported by correlating sentiment scores with issue duration. Understanding these emotional trends is crucial for maintaining a positive and productive open-source community, particularly by addressing negative sentiments in long-duration issues.

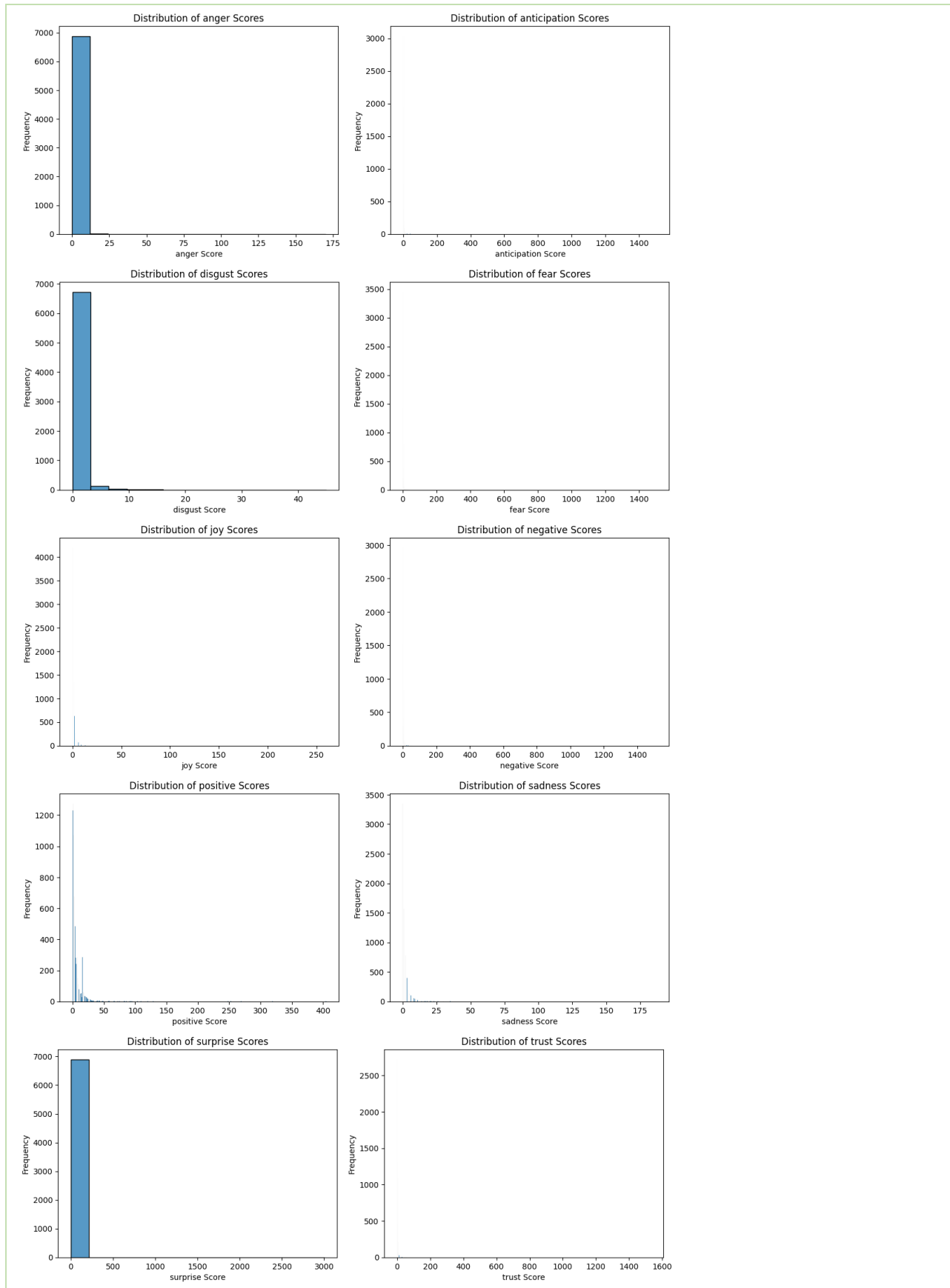


The analysis of emotional effects on GitHub issue lifecycles indicates that quickly resolved issues (0-50 days) are predominantly associated with positive or neutral sentiments, enhancing contributor satisfaction and participation. Moderately resolved issues (50-100 days) exhibit mixed sentiments, with growing frustration over delays, suggesting the need for regular communication to maintain positive sentiment. Prolonged issues (100+ days) show an increase in negative sentiments, leading to significant frustration and disengagement, necessitating effective issue management. Correlation analysis reveals that positive sentiments correlate with faster resolutions, while negative sentiments correlate with prolonged issues, highlighting the importance of fostering positive interactions and timely interventions to improve project efficiency and emotional health.

Results of Emotion Analysis with NRC Emotion Lexicons

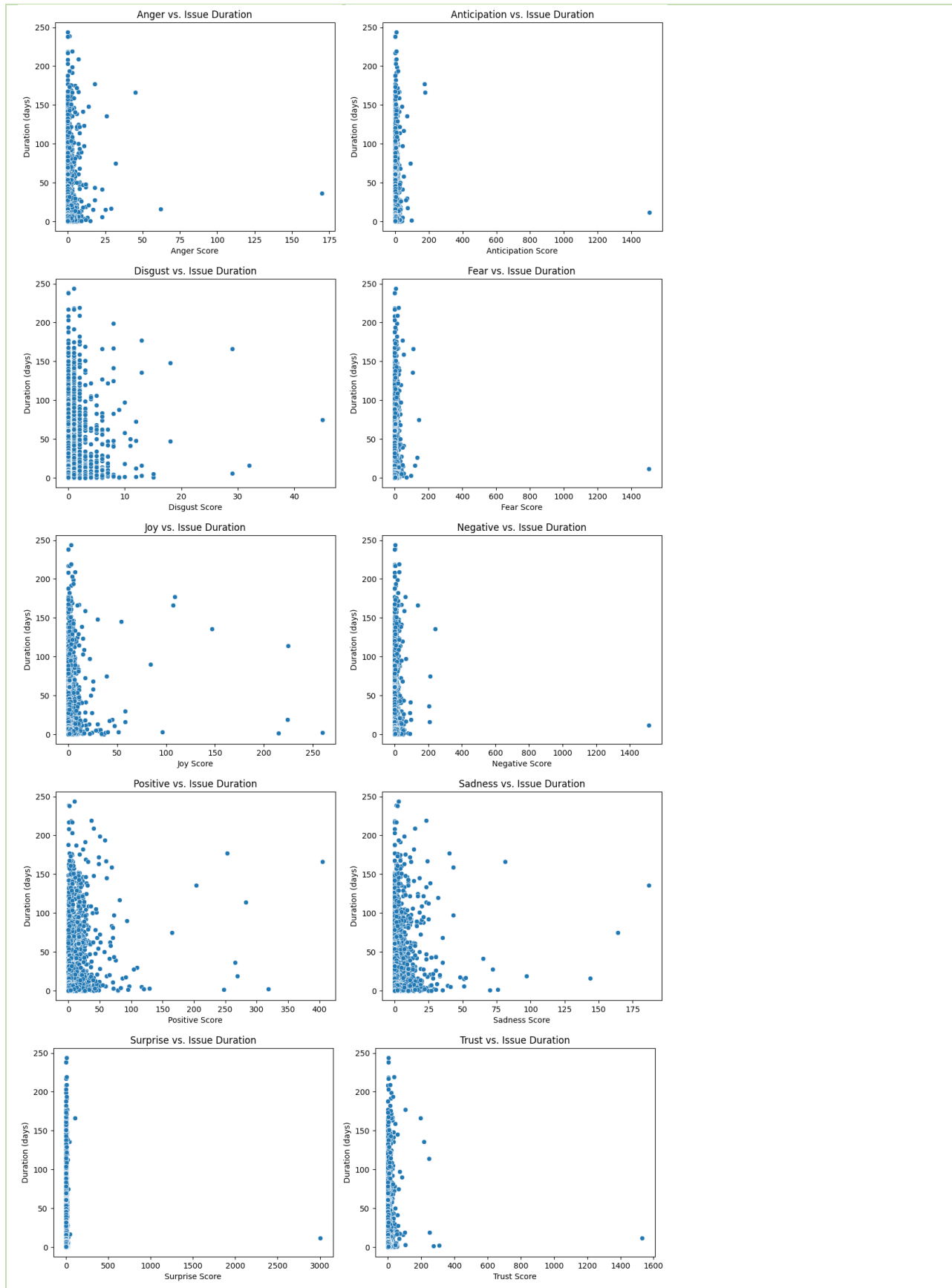


Analyzing emotional trends in short-duration (0-10 days) versus long-duration issues to see if longer issues exhibit more frustration or negative sentiments. It also recommends investigating how emotions evolve over an issue's lifespan, noting potential positive shifts as issues near resolution. Another point is to examine if the perceived complexity of issues correlates with emotional tone, as complex issues might provoke varied emotional responses. Lastly, it highlights the need to assess the impact of community engagement on issue resolution, considering whether increased participation and diverse emotional input expedite or prolong discussions. Integrating these insights can enhance project management and community dynamics in open-source development.



The analysis of emotional scores in GitHub issue comments reveals that most emotions are expressed with low intensity. Anger, disgust, fear, joy, sadness, and surprise are all concentrated at very low values, indicating these emotions are infrequently or minimally expressed. Anticipation scores are slightly more spread out, suggesting some expression of future expectations or plans. Negative emotions, though present, generally show low intensity, reflecting infrequent strong negative interactions. Trust scores, while still low, are more frequently expressed than other emotions, indicating a positive or collaborative atmosphere. Overall, GitHub issue comments are generally neutral or mildly emotional.

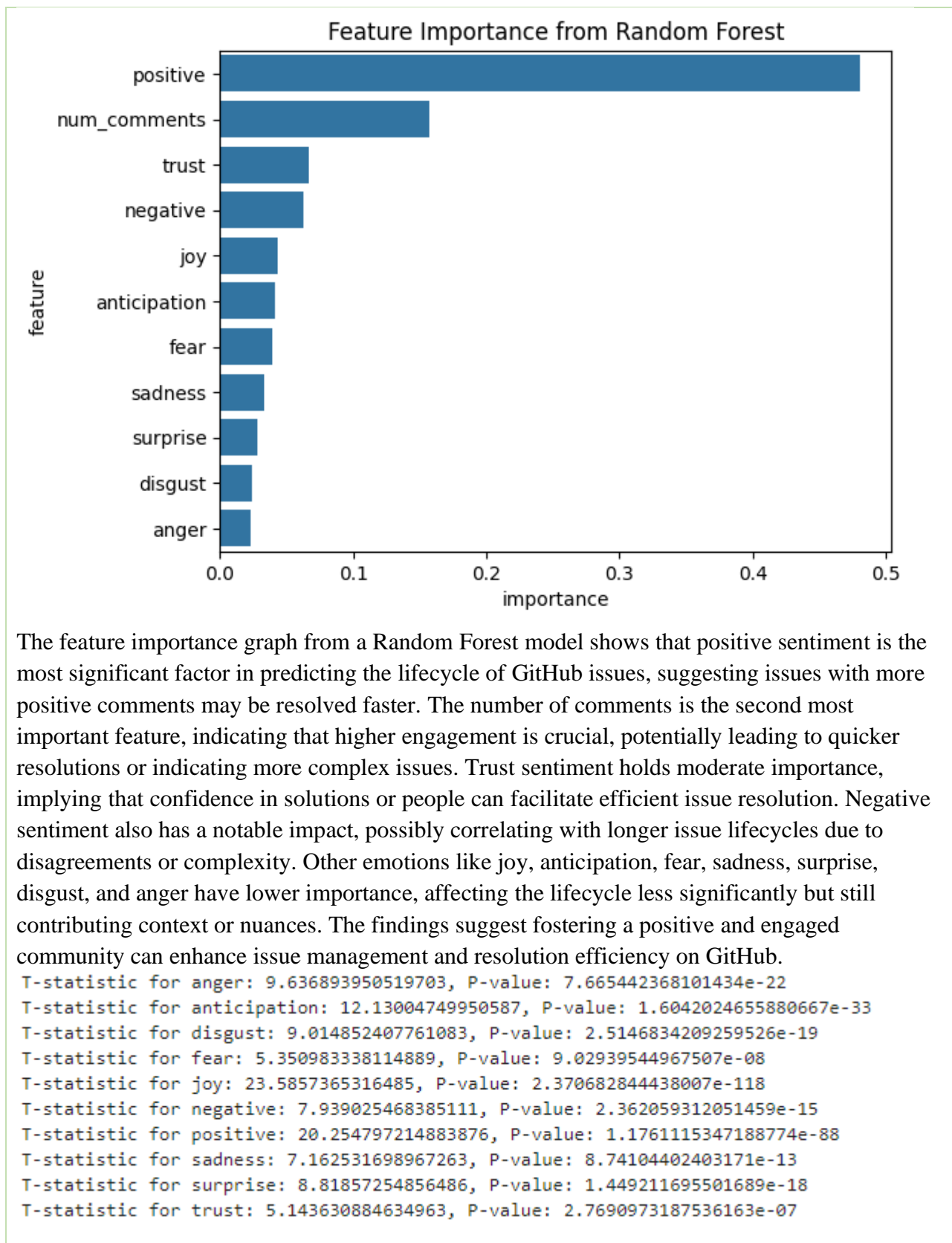
Short-duration issues tend to have neutral or slightly positive emotions, indicative of efficient problem-solving. Long-duration issues, however, might display more diverse emotional patterns, including higher instances of negative emotions due to complexity or frustration. This detailed distribution of emotions helps understand the emotional dynamics in open-source development, suggesting that while negative emotions are rare, trust and anticipation play a significant role in the collaborative process. This insight can be useful for improving project management and fostering a positive community environment.



The analysis indicates that higher anger scores correlate with shorter issue durations, suggesting that issues causing anger are resolved quickly due to perceived urgency. Similarly, higher anticipation scores are linked to shorter durations, reflecting proactive engagement in resolution processes. Higher disgust scores also correlate with shorter durations, possibly to maintain community standards and avoid prolonged negativity. Fear scores, while generally low, tend to cluster around shorter durations, implying that concerns are swiftly addressed to prevent escalation.

Joy scores are low and show no strong correlation with issue duration, indicating that positive sentiments do not significantly affect resolution speed. Higher negative scores overall correspond with shorter durations, suggesting that issues with negative emotions are prioritized to mitigate adverse impacts. Positive scores, like joy, do not significantly influence issue duration, though there is a slight trend towards faster resolution with higher positive scores. Higher sadness scores correlate with shorter durations, indicating swift resolution to maintain community morale. Surprise scores, concentrated at low values, show no significant impact on issue duration.

Trust scores, however, correlate with shorter durations, indicating that issues with trust expressed in comments are resolved faster due to effective collaboration. In summary, negative emotions such as anger, disgust, fear, and sadness are linked to quicker issue resolution, while positive emotions like trust and anticipation also contribute to faster resolutions. Neutral emotions like joy and surprise do not significantly influence resolution speed. Understanding these emotional dynamics can help manage and resolve issues more effectively on GitHub, promoting a positive community atmosphere and enhancing collaborative efficiency.



The T-statistics and P-values indicate that emotions significantly impact the lifecycle of GitHub issues. Positive emotions like joy, positive sentiment, and trust are strongly associated with faster issue resolution, highlighting the benefits of a positive community atmosphere. Negative emotions such as anger, disgust, fear, and sadness also significantly affect issue durations, often leading to quicker resolutions to prevent prolonged negativity. Overall, fostering positive engagement and addressing negative emotions promptly can enhance the efficiency of issue management on GitHub.

Mean Cross-Validation MSE for Random Forest: 732.0235726796902

Mean Squared Error for Gradient Boosting: 651.1807753660468

Mean Squared Error for Random Forest: 644.7202423472025

Mean Squared Error for Linear Regression: 783.4322996895718

I also tried to implement predictive models such as linear regression, Gradient Boosting, Random Forest, Cross-Validation for this project but gave very inaccurate results because the dataset was small and did not have enough features.

The Impact and Future Directions (/ 15 Points)

Explain the potential (or current if exist) impacts of your outcome in terms of how the methods and results will be used in real life, how it will change an existing process, or where it will be published, etc. Also, explain what would be the next step if the project is continued in the future, what kind of qualitative and/or quantitative updates can be made, shortly, where this project can go from here? This section should be between 250-500 words.

Real-Life Applications and Impact

1. Enhanced Issue Management:

- **Prioritization and Resource Allocation:** By analyzing the emotional tone of GitHub issues, project managers can prioritize issues that are likely to take longer to resolve or generate extensive discussions. This allows for more efficient allocation of resources and faster resolution times.
- **Real-Time Emotional Feedback:** Integrating emotion analysis tools into GitHub could provide real-time feedback on the emotional tone of issue discussions. This can help maintain a positive and productive collaborative environment by notifying project managers of emotionally charged discussions, enabling timely interventions to resolve conflicts.

2. Improved Collaboration:

- **Conflict Resolution:** Understanding the emotional dynamics in issue discussions can help identify and address conflicts early, improving team dynamics and overall productivity.
- **Team Morale:** Monitoring emotional tones can contribute to maintaining high team morale by addressing negative sentiments before they escalate, fostering a more supportive and collaborative working environment.

3. Advanced Analytical Tools:

- **Predictive Insights:** The predictive models developed in this study can be used to forecast the lifecycle of new issues based on their emotional profiles, providing valuable insights for better project management.
- **Emotional Analytics Dashboard:** Developing a comprehensive dashboard that visualizes the emotional dynamics of issues over time could become an essential tool for project managers and team leads.

Publication and Dissemination

- **Academic Journals:** The findings of this study can be published in peer-reviewed journals focusing on software engineering, natural language processing (NLP), and human-computer interaction (HCI).
- **Conferences:** Presenting the results at relevant conferences (e.g., ICSE, ACL) will disseminate the knowledge to a broader audience and foster discussions on the integration of emotion analysis in software development.
- **Industry Reports:** Collaborating with industry leaders to publish reports that highlight the practical applications and benefits of emotion analysis in issue management can bridge the gap between research and practice.

Future Directions

1. Advanced NLP Techniques:

- **Transformer-Based Models:** Incorporating transformer-based models like BERT or GPT can enhance the accuracy and depth of emotion analysis by capturing more nuanced emotional expressions.
- **Domain-Specific Emotion Lexicons:** Developing domain-specific emotion lexicons tailored to software development discussions can improve the precision of emotion detection.

2. Expanded Scope:

- **Cross-Platform Analysis:** Extending the study to include other collaborative platforms (e.g., JIRA, Slack) can provide a holistic understanding of emotional dynamics across different tools used in software development.
- **Lifecycle Phases:** Investigating the role of emotions across various phases of the software development lifecycle (e.g., design, coding, testing) can offer comprehensive insights into their impact on project outcomes.

3. Real-Time Implementation:

- **Emotion Analysis Plugins:** Developing plugins for GitHub and other platforms that provide real-time emotional feedback and predictive insights can revolutionize issue management practices.
- **User Studies:** Conducting user studies to evaluate the effectiveness of emotion analysis tools in real-world settings can provide valuable feedback for further refinement and adoption.

4. Longitudinal Studies:

- **Impact Over Time:** Longitudinal studies can assess how emotional dynamics evolve over the course of a project and their long-term impact on team performance and project success.
- **Behavioral Insights:** Combining emotion analysis with behavioral data (e.g., coding activity, communication patterns) can offer deeper insights into team dynamics and individual performance.

In conclusion, the outcomes of this study have the potential to significantly improve issue management and collaboration in software development. By integrating advanced emotion analysis tools, we can foster a more efficient, positive, and productive working environment. Future work will focus on enhancing the analytical techniques, expanding the scope of analysis, and developing real-time implementation tools to maximize the practical impact of this research.