Employee Sentiment Analysis Report

# Approach and Methodology

This section describes the approach taken in the analysis. The methodology involves setup and data collection, and specific tasks including: sentiment labeling, exploratory data analysis (EDA), sentiment scoring of employee emails, ranking employees based on sentiment scores, identifying flight risk, and building a predictive model. In the setup and data collection phase, a module was created to automate the dataset file downloading process, but due to authentication access, it was unsuccessful. Future works will try to fix this module.

The dataset includes 2191 records with 4 columns. The “Subject” column contains the subject of each email, the “body” column contains the message, “date” contains the date sent and “from” contains the sender.

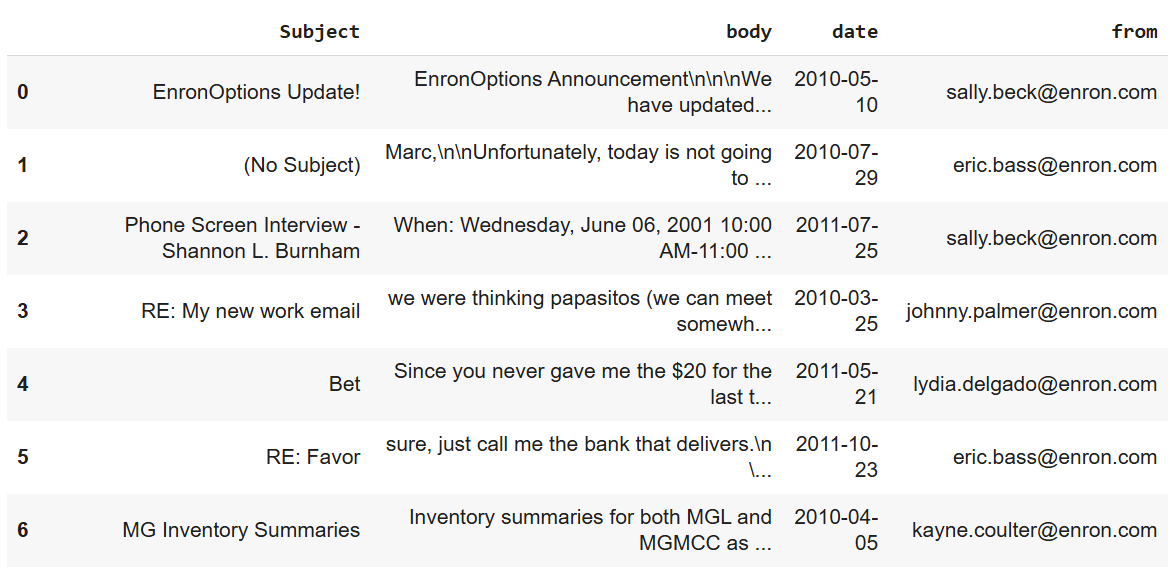


Figure 1: Overview of the dataset

The methodology, results and key findings of each task will be shown below.

## Sentiment Labeling

**Objective:** Label each employee message with one of three sentiment categories: Positive, Negative, or Neutral.

**Methodology**: Use an LLM to label the “body” column of the DataFrame as Positive, Negative, or Neutral, and add the labels as a new column to the DataFrame.

* The transformers library was successfully installed, and a sentiment analysis pipeline using the distilbert-base-uncased-finetuned-sst-2-english model was set up.
* A Python function *get\_sentiment* was created to utilize the sentiment analysis pipeline and return sentiment labels ('Positive', 'Negative', or 'Neutral') for input text.
* The *get\_sentiment* function was applied to the 'body' column of the DataFrame, generating a series of sentiment labels.
* A new column named 'sentiment' was added to the DataFrame, and it was populated with the generated sentiment labels.
* A manual review of a random sample of 20 rows indicated that the LLM's sentiment labels generally aligned with the content, and the accuracy was deemed acceptable for this general sentiment analysis task.

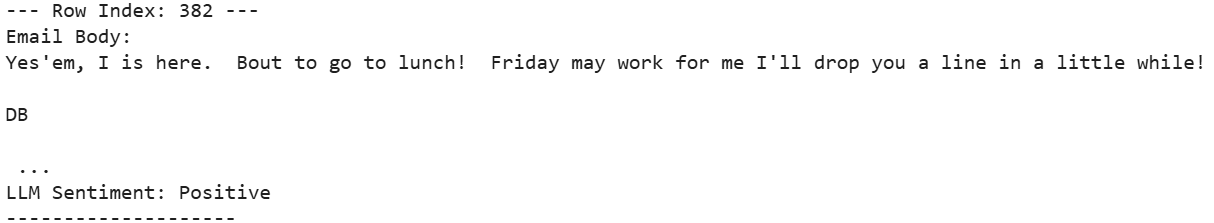


Figure 2: Sample of a record being labeled “Positive”

## Exploratory Data Analysis (EDA)

**Objective:** Understand the structure, distribution, and trends in the dataset through thorough exploration.

**Methodology**: Perform exploratory data analysis (EDA) on the provided DataFrame, focusing on data structure, sentiment distribution, and trends over time. Include visualizations to effectively communicate findings.

* Check the number of records, data types, and identify missing values in the DataFrame: The dataset contains 2191 records with no missing values in the analyzed columns (“Subject”, “body”, “date”, “from”, “sentiment”).
* Analyze sentiment distribution: The dataset is imbalanced, there are 1191 records being labeled “Negative”, 1000 being labeled “Positive” and none labeled “Neutral”. The sentiment analysis reveals a predominance of negative sentiment in the emails.

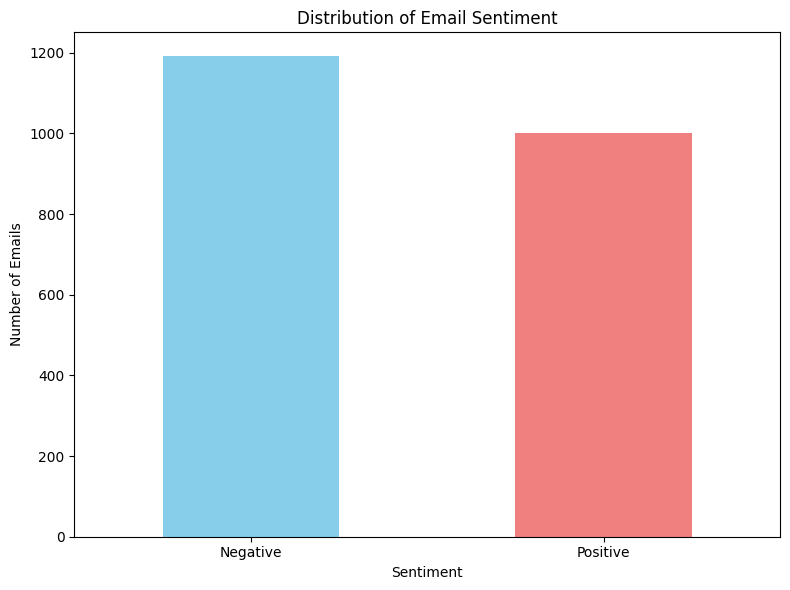


Figure 3: Distribution of Email Sentiment

* Analyze trends overtime: Sentiment distribution over time shows fluctuations in the number of emails per sentiment category across different years. The number of emails labeled “Negative” decreased while that of “Positive” increased slightly.

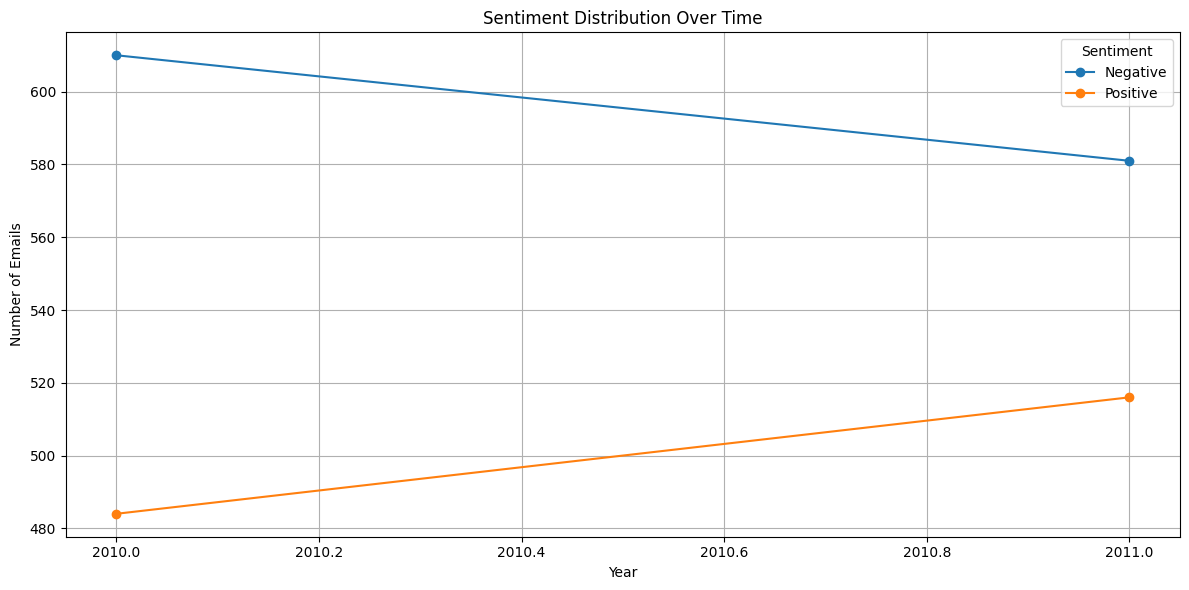


Figure 4: Sentiment Distribution Over time

* Analysis of frequent senders: identified individuals like Lydia Delgado, John Arnold, and Sally Beck as top contributors by email volume.

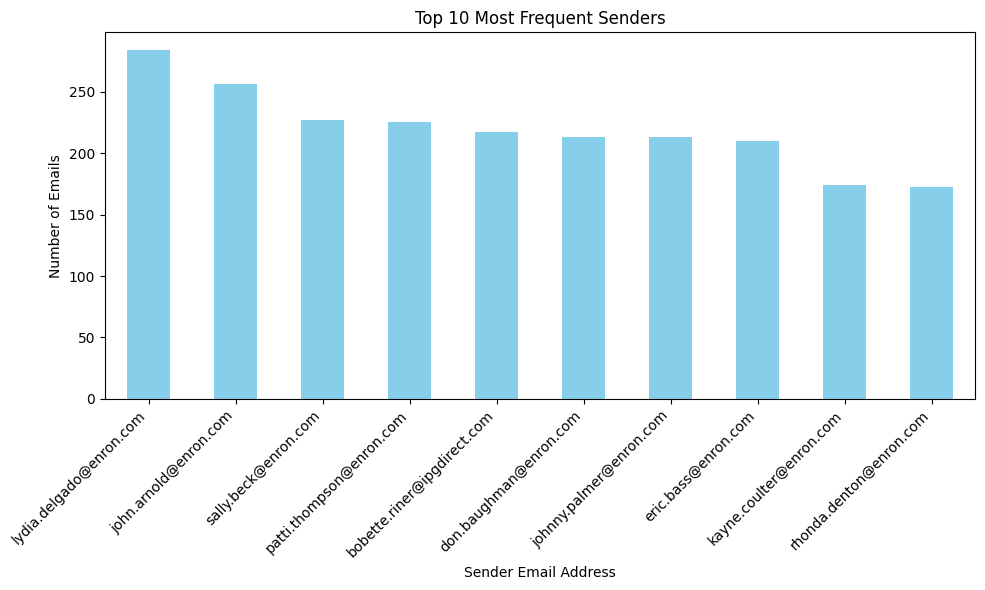


Figure 5: Top 10 most frequent senders

* Frequent subjects for negative sentiment include "(No Subject)" and "Re:", along with specific topics like "Master Power Contracts".

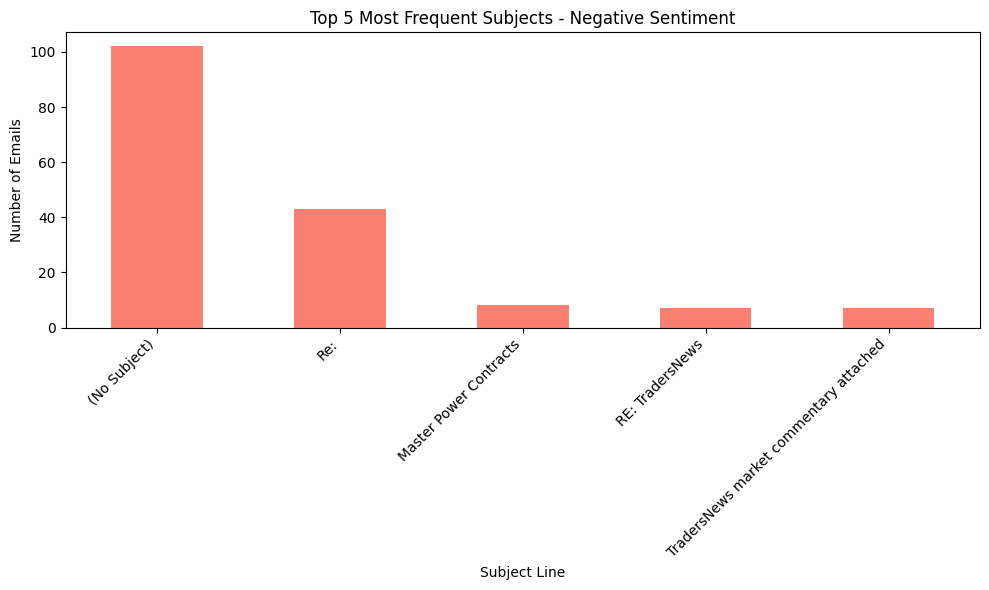


Figure 6: Top 5 most frequent subjects for emails with negative sentiment

* Frequent subjects for positive sentiment also include "(No Subject)" and "Re:", but also positive indicators like "Congratulations".

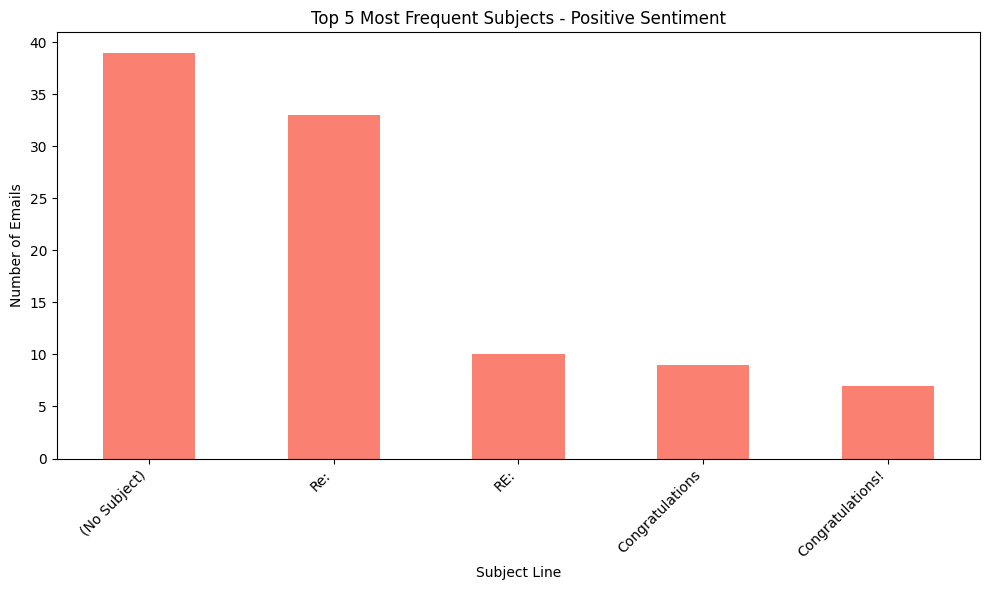


Figure 7: Top 5 most frequent subjects for emails with positive sentiment

## Employee Score Calculation

**Objective:** Compute a monthly sentiment score for each employee based on their messages.

**Methodology**: Calculate a monthly sentiment score for each employee based on their messages, assigning +1 for Positive, -1 for Negative, and 0 for Neutral sentiment, and aggregate these scores on a monthly basis for each employee, ensuring the score resets at the beginning of each new month.

* Sentiment Score Assignment: A numerical score was assigned to each email based on its sentiment label derived from the LLM analysis. Positive sentiment emails were assigned a score of +1. Negative sentiment emails were assigned a score of -1. Neutral sentiment emails were assigned a score of 0. This score was added as a new column named *sentiment\_score* to the DataFrame.
* Extracting Month and Year: To facilitate monthly aggregation, the year and month were extracted from the date column and added as new columns, year and month, respectively.
* Grouping and Aggregation: The DataFrame was then grouped by the employee's email address (from), the year (year), and the month (month). Within each of these groups, the *sentiment\_score* was summed. This sum represents the total sentiment score for a given employee in a specific month and year.
* Monthly Score Reset: The grouping by year and month inherently ensures that the sentiment score is calculated independently for each month. As a result, the accumulated sentiment score for an employee naturally resets at the beginning of each new month and year combination.

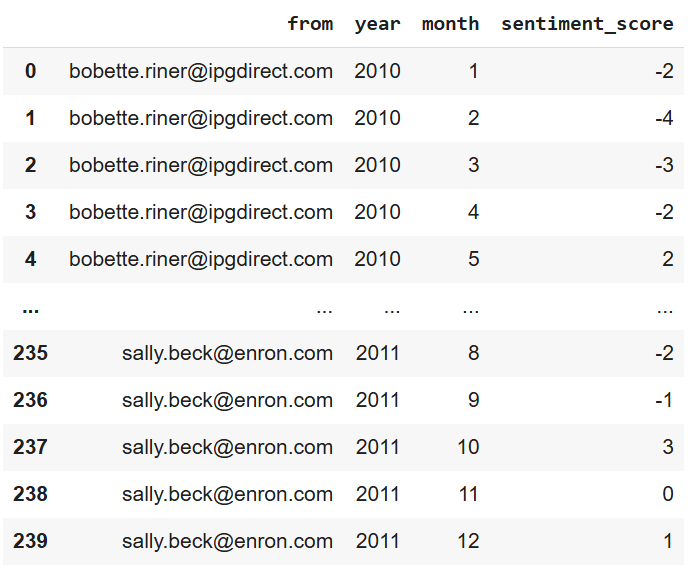


Figure 8: Final DataFrame calculating the monthly sentiment score of each employee

## Employee Ranking

**Objective:** Generate ranked lists of employees based on their monthly sentiment scores.

**Methodology**: Rank employees based on their monthly sentiment scores, identifying the top three positive and top three negative employees for each month, and present the rankings.

* **Monthly Sentiment Score Calculation:** As detailed in the "Employee Score Calculation" section, a numerical sentiment score was assigned to each email (+1 for Positive, -1 for Negative, 0 for Neutral). These individual scores were then aggregated by employee, year, and month to obtain a single monthly sentiment score for each employee.
* **Sorting for Ranking:** The DataFrame containing the monthly sentiment scores was sorted. The primary sorting was by year and then by month in ascending order. Within each month, the scores were sorted in descending order, placing the highest (most positive) sentiment scores at the top and the lowest (most negative) sentiment scores at the bottom.
* **Identifying Top Positive Employees:** For each month, the top three employees with the highest positive sentiment scores (scores greater than 0) were identified from the sorted data.

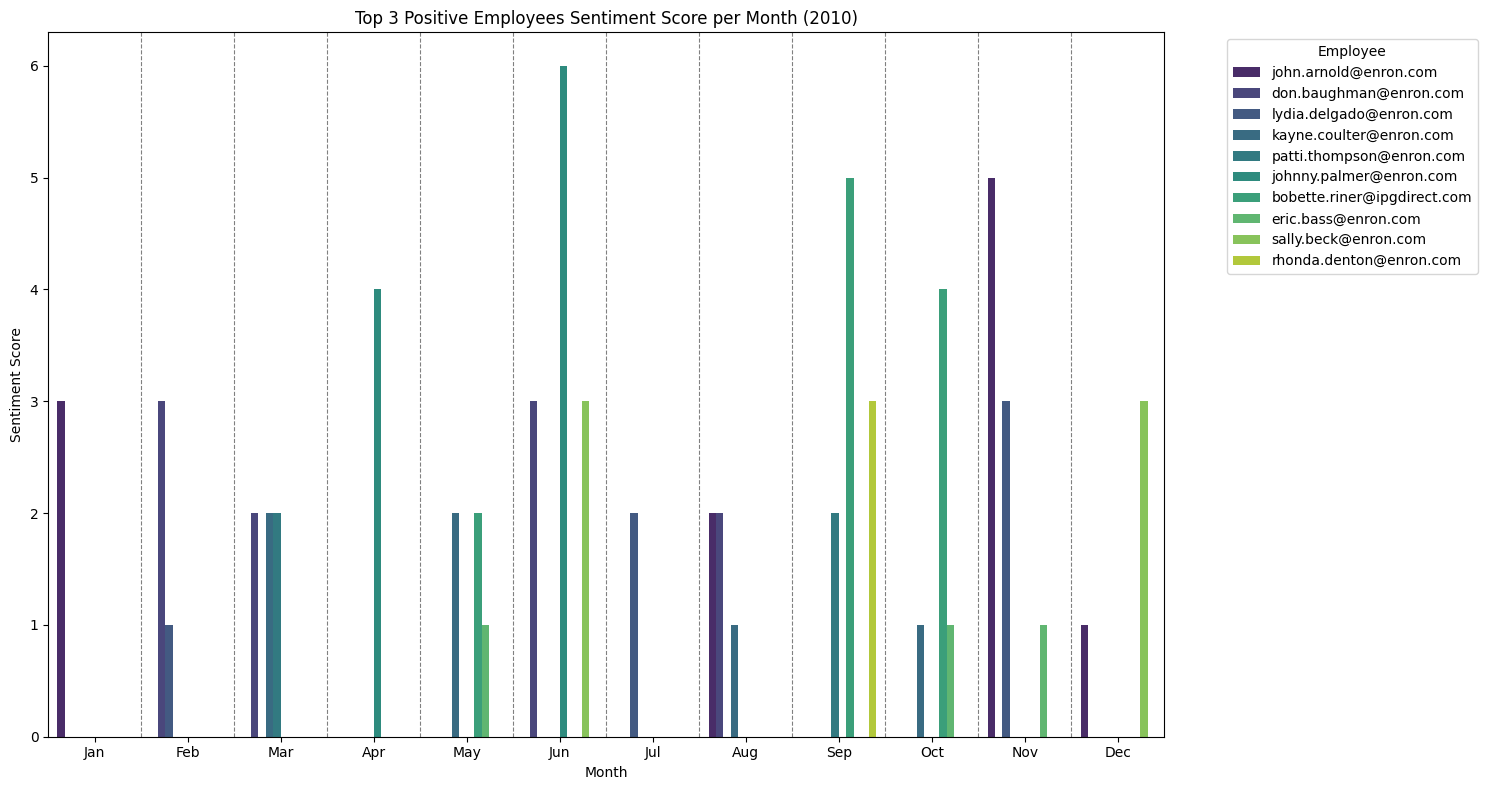


Figure 9: Top 3 employees with positive sentiment score of each month in 2010

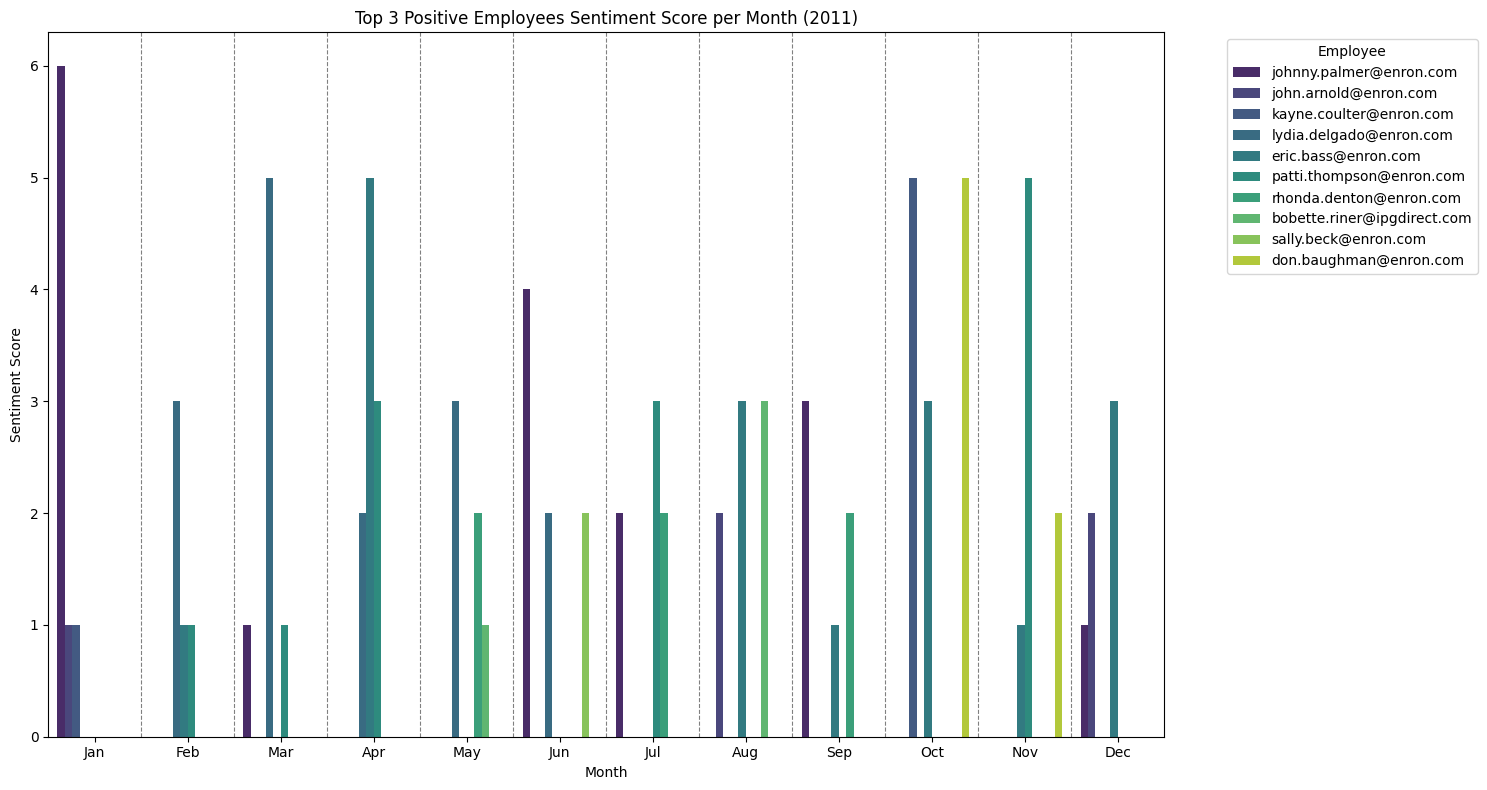


Figure 10: Top 3 employees with positive sentiment score of each month in 2011

* **Identifying Top Negative Employees:** For each month, the top three employees with the lowest (most negative) sentiment scores (scores less than 0) were identified from the sorted data.

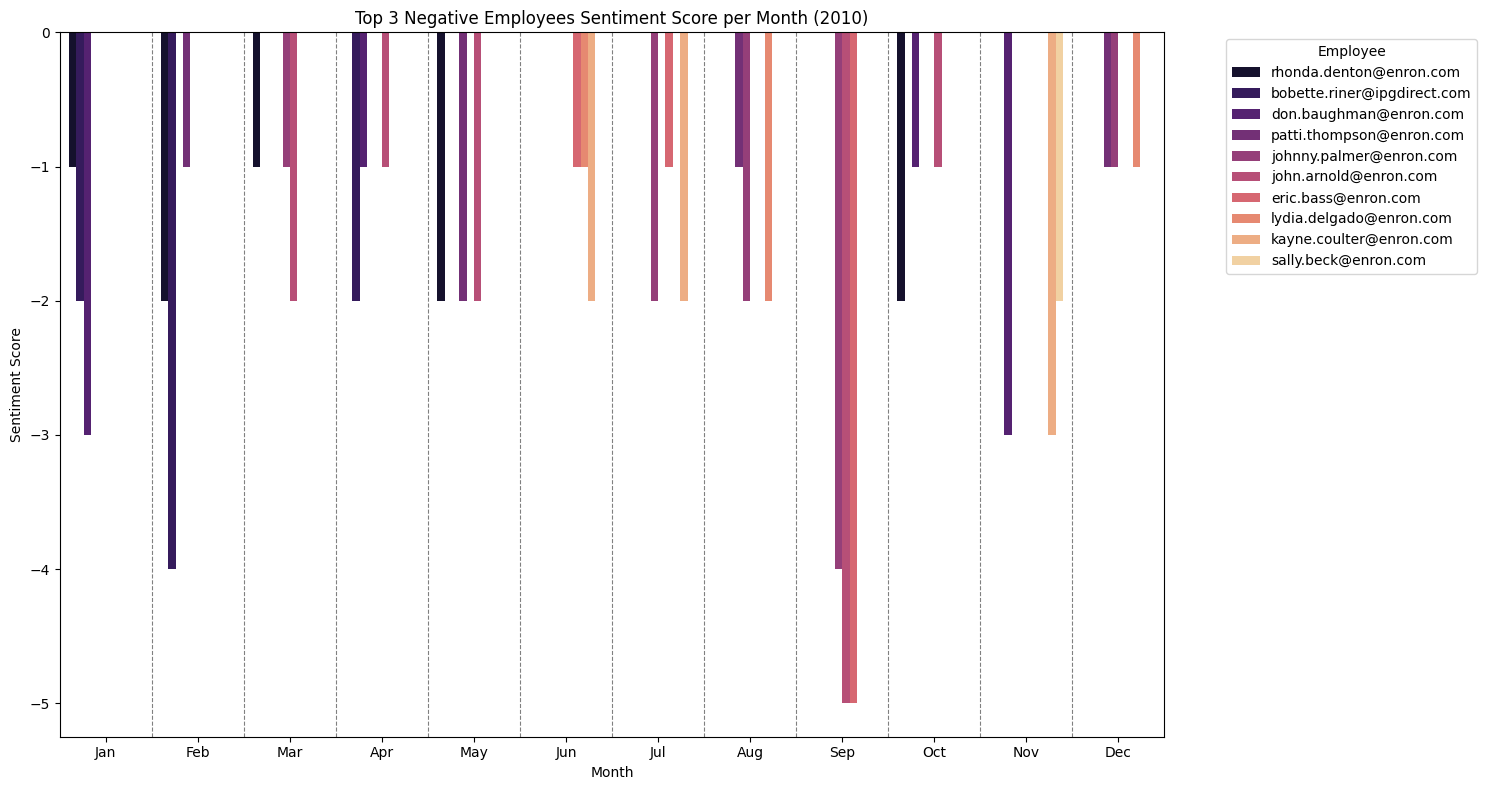


Figure 11: Top 3 employees with negative sentiment score of each month in 2010

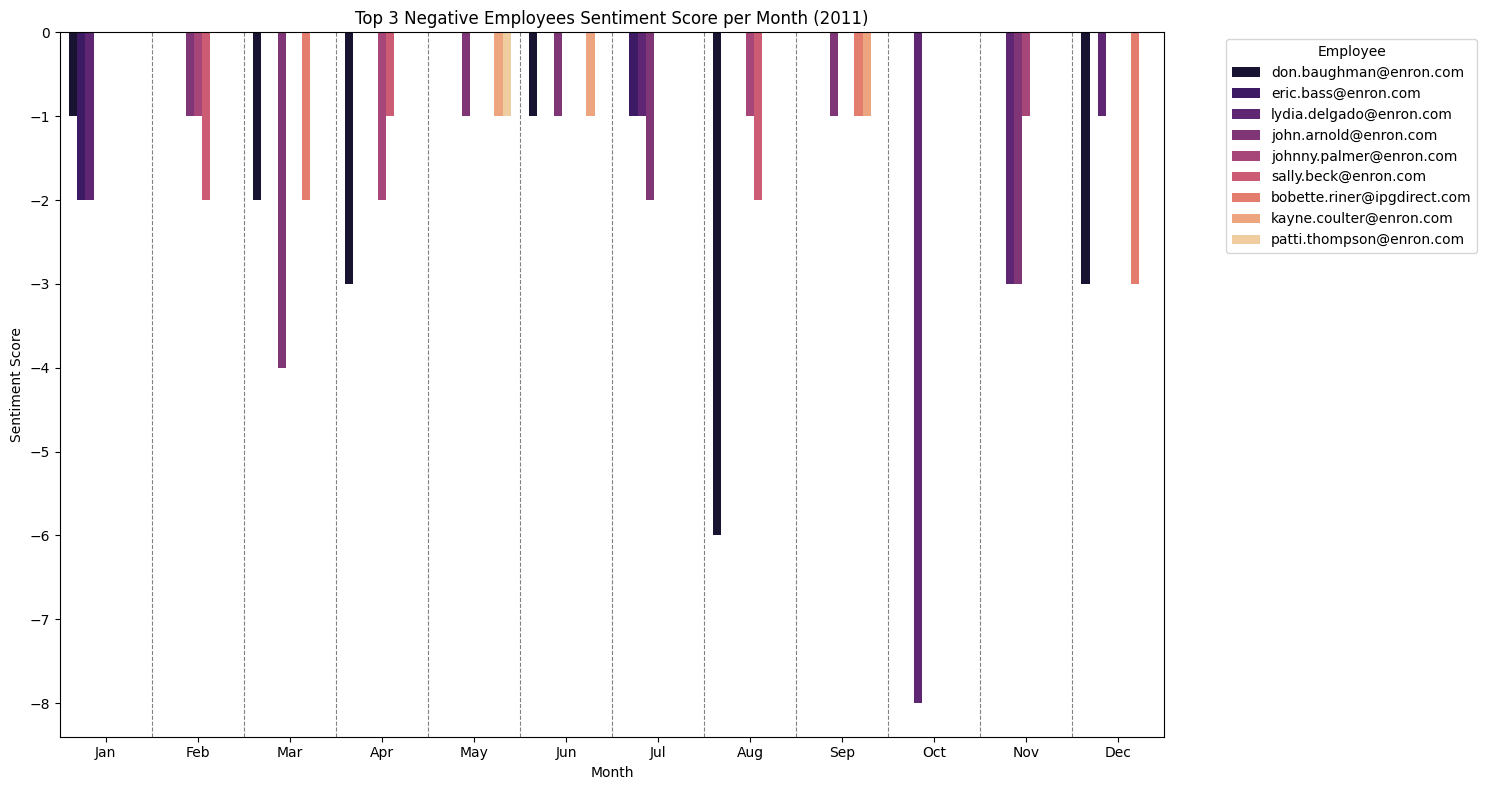


Figure 12: Top 3 employees with negative sentiment score of each month in 2011

This process ensured that the rankings for each month were independent and reflected the sentiment performance of employees within that specific month.

## Flight Risk Identification

**Objective:** Identify employees who are at risk of leaving based on their monthly sentiment scores. This task is critical for identifying potential issues in employee engagement and retention.

**Methodology:** Identify employees who are at risk of leaving based on their monthly sentiment scores. A Flight risk is any employee who has sent 4 or more negative mails in the span of 30 days (irrespective of the score). The 30-day period is rolling count of days, irrepective of months. Extract a list of these employees. Ensure that this flagging process is robust.

* Negative Email Sentiment: Create another dataframe containing only emails classified with a 'Negative' sentiment were considered for this analysis.
* Rolling 30-Day Window: A rolling 30-day window was applied to each employee's negative emails. For each negative email sent, the analysis looked back 30 days from the date of that email.
* Threshold for Negative Emails: An employee was flagged as at risk if they sent 4 or more negative emails within any of these rolling 30-day windows.
* Calculating Rolling Negative Count: The *rolling\_negative\_count* was calculated by first sorting each employee's negative emails by date. Then, for each email, the number of negative emails sent by that same employee in the preceding 30 days was counted.
* Identifying At-Risk Employees: Any employee who, at any point in time, had a *rolling\_negative\_count* of 4 or more was identified as an at-risk employee. The final list of at-risk employees includes all unique individuals who met this criterion at least once.

## Predictive Modeling

**Objective:** Develop a linear regression model to analyze sentiment trends and predict sentiment scores using a variety of independent variables that may influence sentiment scores. Due to the ambiguity the objective of the task using sentiment scores, whether of each email or the monthly sentiment score), we create two different models for both cases.

**Methodology:** Develop two different linear regression models to analyze sentiment trends and predict sentiment scores in two cases using a variety of independent variables that may influence sentiment scores such as message length, word count, average word count, etc.

**Case 1: Develop a linear regression model to analyze sentiment trends and predict sentiment score of each message**

* **Model Development Process:**

Feature Selection: The following features were selected as independent variables based on the assumption that they might influence the sentiment expressed in emails:

* + message\_length: The number of characters in the email body.
  + word\_count: The number of words in the email body.
  + monthly\_message\_count: The total number of emails sent by an employee within a specific month. These features represent quantifiable aspects of communication volume and verbosity, which could potentially correlate with sentiment.

Data Splitting: The dataset was split into training and testing sets. A 75/25 ratio was used, with 75% of the data allocated for training the model (X\_train, y\_train) and 25% for evaluating its performance on unseen data (X\_test, y\_test). This split helps in assessing the model's generalization ability.

Model Used: A Linear Regression model from the sklearn.linear\_model library was used. Linear regression is a simple yet powerful algorithm that models the linear relationship between independent variables and a dependent variable.

* **Model Evaluation Results:**

The trained linear regression model was evaluated on the testing set using the following metrics:

* The Mean Squared Error (MSE) of the model on the test set is approximately 0.9334.
* The Root Mean Squared Error (RMSE) of the model on the test set is approximately 0.9662.
* The R-squared (R2) value of the model on the test set is approximately 0.0528, indicating that about 5.28% of the variance in sentiment scores is explained by the model.
* **Model Coefficients and Intercept:**
  + Intercept: The intercept of the model is approximately -0.0126. This is the predicted sentiment score when all independent variables are zero. In this context, where zero values for message length, word count, and monthly message count are not typical of meaningful communication, the intercept's interpretation in isolation is limited.
* message\_length Coefficient**:** The coefficient for message\_length is approximately -0.0032, suggesting a weak negative relationship with sentiment. This coefficient indicates the change in predicted sentiment score for a one-unit increase in message length, holding other features constant. The negative value suggests a slight tendency for longer messages to be associated with more negative sentiment.
* word\_count Coefficient: The coefficient for word\_count is approximately 0.0177, suggesting a weak positive relationship with sentiment. This coefficient shows the change in predicted sentiment score for a one-unit increase in word count, holding other features constant. The positive value suggests that messages with more words tend to be associated with slightly more positive sentiment.
* monthly\_message\_count Coefficient:  The coefficient for monthly\_message\_count is approximately 0.0028, suggesting a weak positive relationship with sentiment. This coefficient represents the change in predicted sentiment score for a one-unit increase in the monthly message count, holding other features constant. The positive value suggests that a higher frequency of messages might be slightly associated with more positive sentiment.

The linear regression model trained on the selected features provides some limited insight into the factors influencing email sentiment. While the coefficients suggest weak linear relationships between the features and sentiment (with word count and monthly message count showing a slight positive association, and message length a slight negative association), the low R-squared value indicates that these features alone are not strong predictors of sentiment. A significant portion of the variability in email sentiment remains unexplained by this model. This suggests that other factors not included in this analysis, or potentially non-linear relationships, play a more dominant role in determining email sentiment.

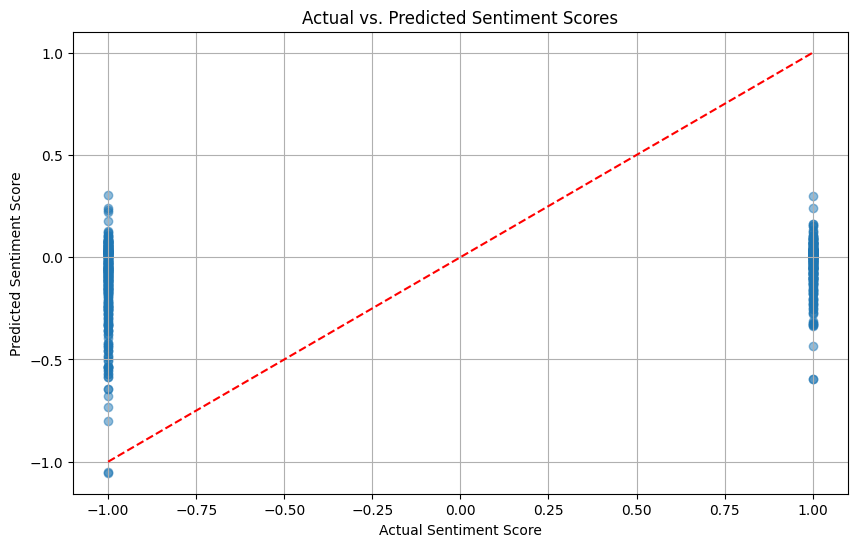


Figure 13: Plot depicting Actual Sentiment Score and Predicted Sentiment Score

This plot shows that the predicted sentiment scores from the first Linear Regression model do not align well with the actual sentiment scores. The points are tightly clustered near -1 and +1 on the x-axis, but the predicted scores mostly center near 0, indicating severe underfitting. The model fails to capture the polarity of sentiment, producing predictions that are too close to neutral.

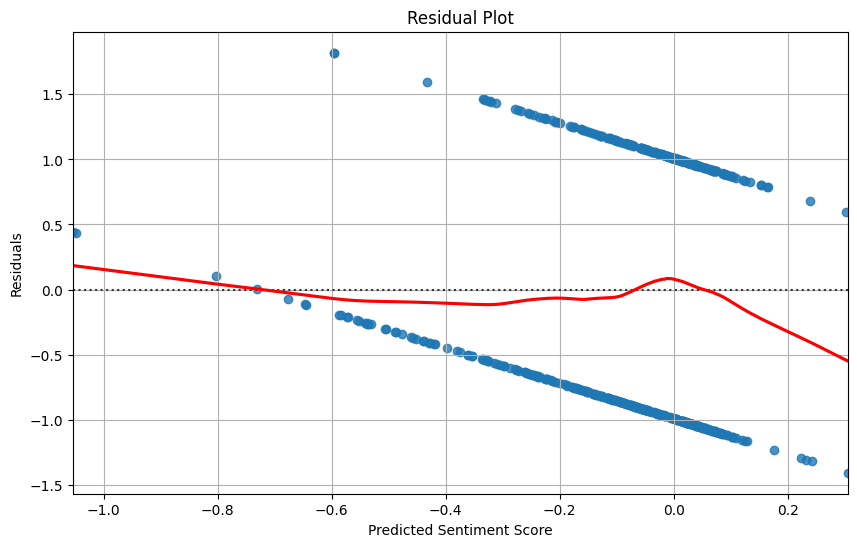


Figure 14: Residual plot

The residual plot shows a clear pattern instead of random scatter, suggesting that the model’s errors are systematic and not random. Many residuals are far from zero, and the red smoothing line deviates notably from the horizontal axis. This confirms that the Linear Regression model does not adequately fit the data, violating the assumption of homoscedasticity.

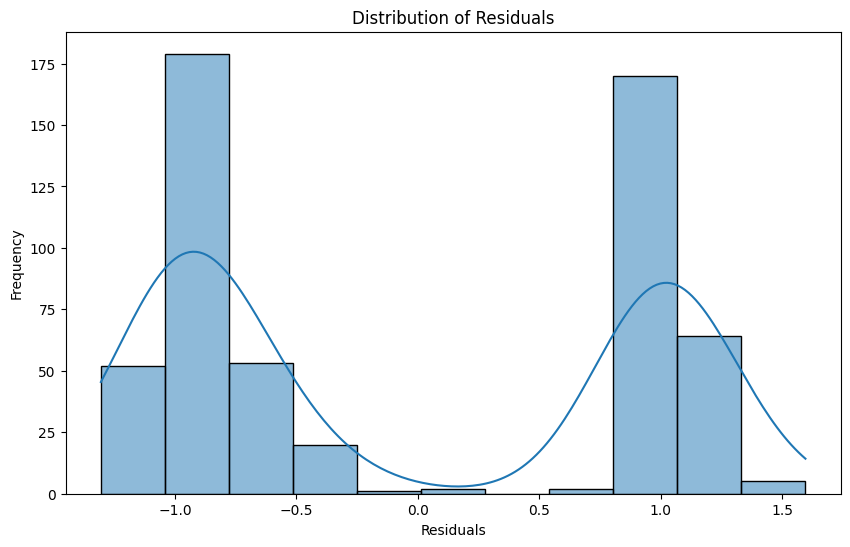


Figure 15: Distribution of Residuals

The residuals are bimodally distributed, with peaks around -1 and +1. This indicates that the model consistently overestimates negative scores and underestimates positive scores. The lack of a normal, bell-shaped residual distribution further supports that the Linear Regression model is inappropriate for this dataset and is failing to capture its underlying structure.

Based on the plots and the low R-squared value, the linear regression model developed with these specific features does not appear to be suitable for accurately predicting sentiment scores or analyzing sentiment trends effectively. The model's performance is quite poor, and a significant portion of the sentiment variability remains unexplained. The fundamental purpose of a linear regression model is to predict a continuous numerical output. The sentiment scores, -1 and 1, are not a continuous range. They are discrete, distinct categories. This type of problem is known as a classification problem, not a regression problem.

**Case 2: Develop a linear regression model to predict monthly sentiment scores using monthly aggregated features.**

* **Model Development Process:**

Feature Selection: Based on the monthly aggregated data, the following features were selected as independent variables to predict the monthly\_sentiment\_score:

* + monthly\_message\_count: The total number of emails sent by an employee in a given month.
  + average\_message\_length: The average character length of emails sent by an employee in a given month.
  + average\_word\_count: The average number of words in emails sent by an employee in a given month. These features were chosen as they represent quantifiable aspects of monthly communication volume and verbosity that might correlate with overall monthly sentiment.

Data Preparation: Before splitting, the selected features and the target variable (monthly\_sentiment\_score) were checked for missing values and appropriate data types. No missing values were found, and the data types (int64 and float64) were suitable for regression.

Data Splitting: The monthly aggregated dataset was split into training and testing sets using the train\_test\_split function from sklearn.model\_selection. A 75/25 ratio was used, with 75% of the data allocated for training and 25% for testing. A random\_state of 42 was set for reproducibility.

Model Used: A LinearRegression model from the sklearn.linear\_model library was chosen for this task. Linear regression models the linear relationship between the independent variables and the dependent variable.

* **Model Evaluation Results:**

The performance of the trained linear regression model was evaluated on the testing set (X\_test, y\_test) using common regression metrics:

* The MSE of 7.6642 represents the average squared difference between the actual and predicted monthly sentiment scores. This value is relatively high given the range of possible monthly sentiment scores, indicating a significant average prediction error.
* The RMSE of 2.7684 provides the average magnitude of the prediction errors in the units of the monthly sentiment score. This means, on average, the model's predictions are off by about 2.77 units, which is substantial considering the sentiment scores are sums of +1 and -1 values.
* The R2 value of -0.0428 is very low and negative. A negative R2 indicates that the model performs worse than a simple horizontal line at the mean of the target variable. In other words, the chosen independent variables (monthly message count, average message length, and average word count) not only fail to explain the variance in monthly sentiment scores but also result in a model that is less accurate than simply predicting the average monthly sentiment score for all instances.
* **Model Coefficients and Intercept:**
* Intercept: The intercept represents the predicted monthly sentiment score when all independent variables (monthly\_message\_count, average\_message\_length, average\_word\_count) are zero. Similar to the individual message model, interpreting the intercept in isolation here is not particularly meaningful as these feature values are unlikely to be zero in a realistic monthly summary.
* monthly\_message\_count Coefficient: This coefficient indicates that for every one-unit increase in the monthly\_message\_count, the predicted monthly sentiment score decreases by approximately 0.0892, assuming other features are held constant. This suggests a slight negative association between the number of messages sent in a month and the overall monthly sentiment score.
* average\_message\_length Coefficient: For every one-unit increase in the average\_message\_length for a given month, the predicted monthly sentiment score decreases by approximately 0.0244, holding other features constant. This suggests a weak negative relationship between the average length of messages sent in a month and the monthly sentiment.
* average\_word\_count Coefficient: This coefficient shows that for every one-unit increase in the average\_word\_count for a given month, the predicted monthly sentiment score increases by approximately 0.1249, assuming other features are held constant. This suggests a positive association between the average number of words in messages sent in a month and the monthly sentiment.

The linear regression model for predicting monthly sentiment scores using monthly aggregated features exhibits very poor performance. The low (and negative) R-squared value clearly indicates that monthly\_message\_count, average\_message\_length , and average\_word\_count are not effective linear predictors of monthly sentiment.

While the coefficients suggest some weak relationships (negative with message count and length, positive with word count), the overall model's inability to capture the variability in monthly sentiment scores means these relationships, as modeled linearly, are not significant drivers of monthly sentiment.

The high MSE and RMSE values further emphasize the model's lack of predictive accuracy. The significant average error suggests that using this model for predicting or understanding monthly sentiment trends based on these features would be unreliable.

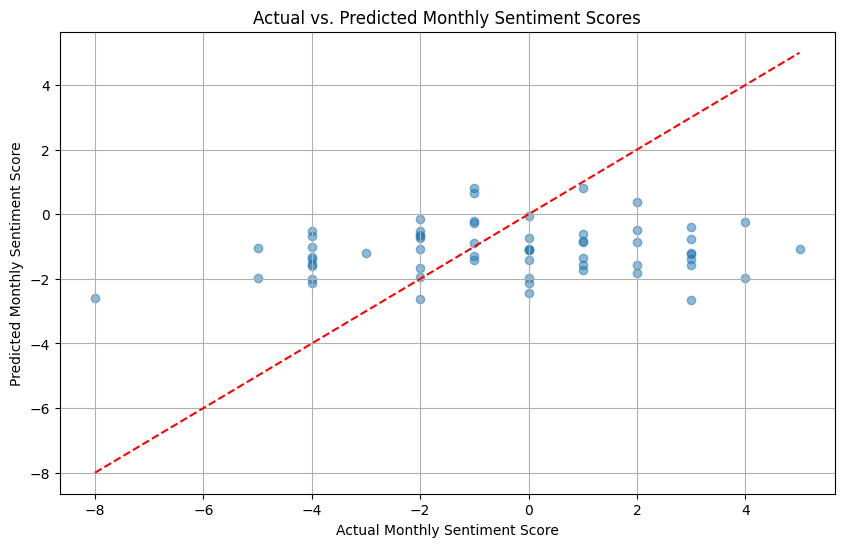


Figure 16: Plot depicting Actual Monthly Sentiment Score and Predicted Monthly Sentiment Score

This plot compares the actual and predicted monthly sentiment scores from the second model. The predictions align more closely to the red diagonal line compared to the first model, showing that the model better captures the general trend in sentiment. However, some points still deviate significantly, suggesting that while the second model improves fit, it still has room for better generalization.

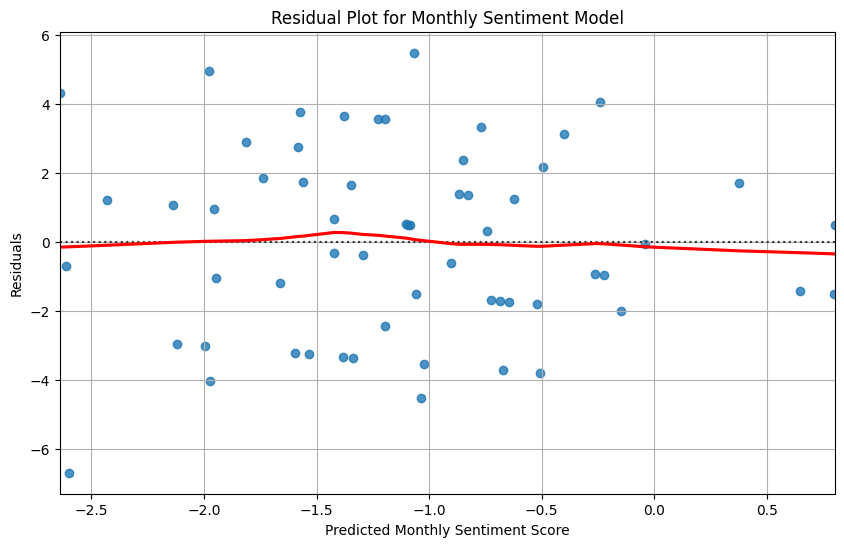


Figure 17: Residual Plot for monthly sentiment model

The residual plot shows a more random scatter of residuals around the zero line, with less pronounced structure compared to the first model. This suggests the second Linear Regression model reduces systematic error patterns and better satisfies the assumption of homoscedasticity, though a few outliers remain.

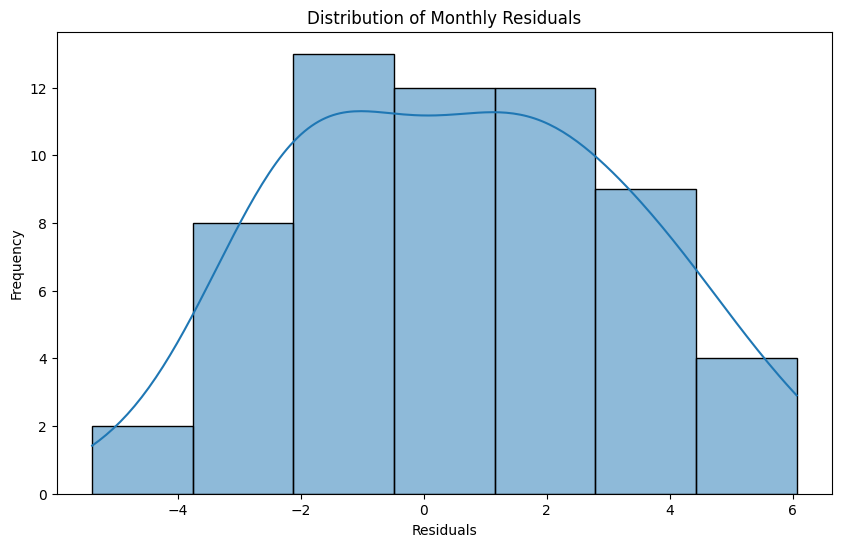


Figure 18: Distribution of Residuals of Monthly Sentiment Model

The residuals are more symmetrically distributed and closer to a normal shape than in the first model. This indicates that the second Linear Regression model has improved in capturing the underlying data structure, reducing bias and providing more reliable predictions overall.

The findings suggest that other factors not included in this model, or more complex, non-linear relationships between these features and sentiment, are likely more influential in determining monthly sentiment. The discrete nature of the individual sentiment scores (+1, -1) which are then summed for the monthly score might also contribute to the poor performance of a linear regression model, which assumes a continuous target variable. More sophisticated modeling techniques or the inclusion of different types of features (e.g., content-based features beyond simple length and word count) would be necessary to build a more effective predictive model for monthly sentiment.

# Overview and Evaluation of the Predictive Model

One model predicts the sentiment score of individual message, one predicts the monthly aggregated sentiment score. While the second model perform slightly better, both model don’t show good results and performance. This indicate better more sophisticated modeling techniques or the inclusion of different types of features would be necessary to build a more effective predictive model.

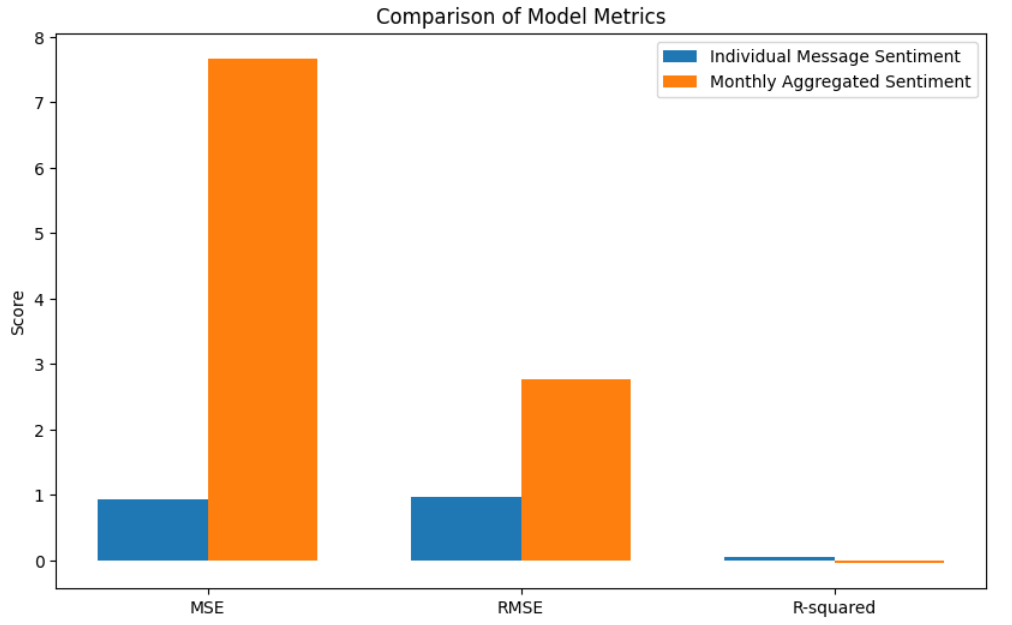


Figure 19: Comparison of metrics of two linear regression models