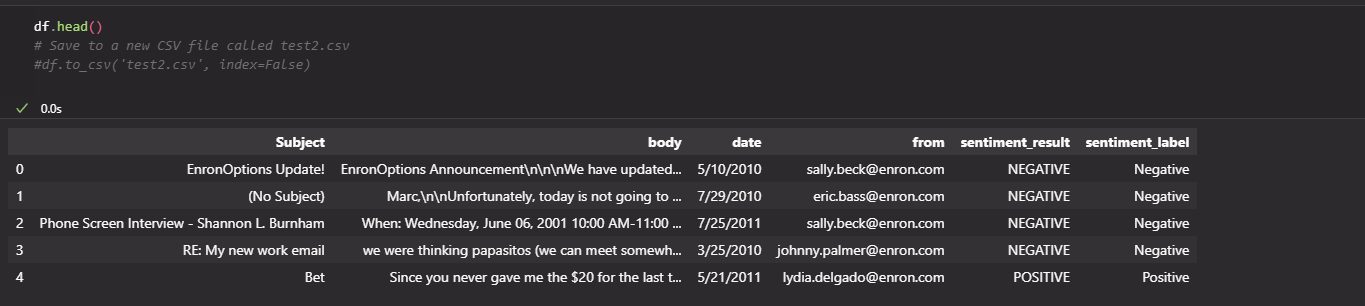
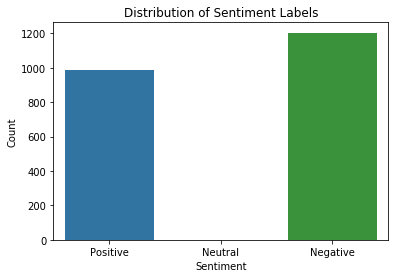
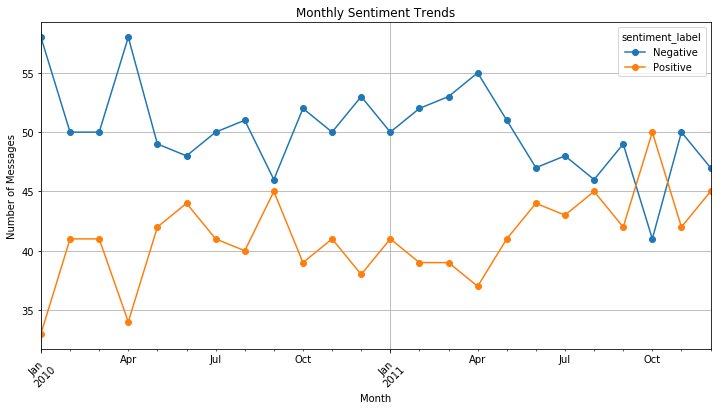
Employee Sentiment & Engagement Analysis - Final Report

**Task 1: Sentiment Labeling**  
**Objective:**  
To assign each employee message a sentiment label: Positive, Negative, or Neutral.  
  
**Approach:**  
I used Hugging Face's transformers library, specifically the DistilBERT model:  
distilbert-base-uncased-finetuned-sst-2-english, which is fine-tuned for sentiment analysis.  
  
The model was accessed via the pipeline("sentiment-analysis") method.  
  
Each message in the 'body' column was passed through this model, which returned a sentiment label.  
  
Results were stored in a new column called sentiment\_result.  
  
These were then mapped into project-specific labels:Positive,Negetive and Neutral.

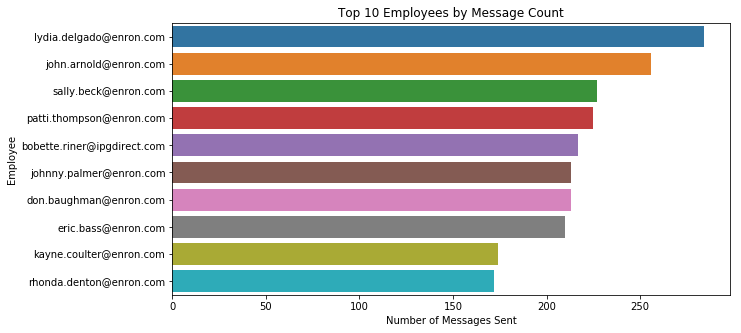


**Justification:**  
  
This method is fast, reproducible, and requires no manual labeling.  
  
Using a pre-trained LLM ensures high-quality predictions without building a model from scratch.  
  
Mapping allows standardization to the project's desired label set.

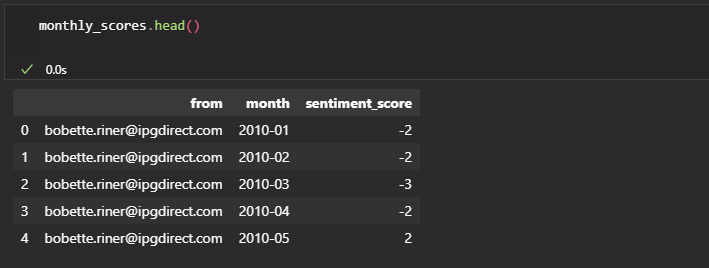
**Task 2: Exploratory Data Analysis (EDA)**  
**Objective:**  
Understand the dataset’s structure, sentiment distribution, and engagement patterns to lay the foundation for sentiment scoring and risk analysis.  
  
The chart shows that almost a 1000 emails have positive sentiment while around 1200 emails have a negative sentiment with no neutral sentiment.



With the passing of each month it is quite observant that the positive sentiment trend increases while the negative sentiment decreases over time.

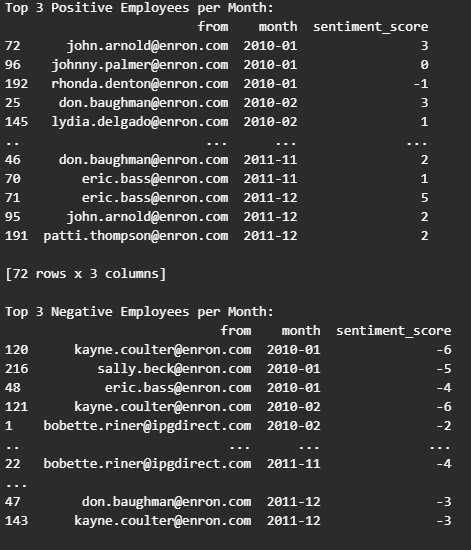
******

The graph above shows the emails from which most of the messages were received .

**Task 3: Employee Score Calculation  
Objective:**Quantify each employee’s overall sentiment trend by computing a monthly sentiment score derived from their messages.  
 **Methodology:**  
A sentiment score is assigned to each message using the following scheme:  
  
Positive → +1  
  
Negative → –1  
  
Neutral → 0  
  
Each employee’s messages are grouped by month using the date column (converted to a monthly period).  
  
Sentiment scores are aggregated per employee per month.  
  
This aggregated score represents how positively or negatively engaged an employee was during that month.

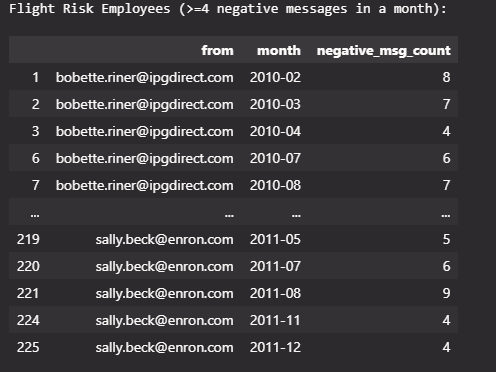
**Outcome:**A new DataFrame is created showing each employee’s sentiment score for every month they sent a message. This forms the foundation for identifying top performers, negative trends, and potential flight risks in future tasks.

**Task 4: Employee Ranking  
Objective:**Identify the most engaged (and potentially disengaged) employees based on their monthly sentiment scores.  
  
**Methodology:**  
For each month:  
  
Find the Top 3 Positive Employees (highest sentiment scores).  
  
Find the Top 3 Negative Employees (lowest sentiment scores).  
  
If there’s a tie, sort alphabetically by employee name.

  
  
**Observation:**The output shows the most top 3 positive/negetive employees from each month. Each month has 3 so a total of 3x24x2 months total of 72 observations.

**Task 5: Flight Risk Identification  
Objective:**Identify employees who might be at risk of leaving the organization, based on their negative sentiment.  
  
**Definition:**Any employee who has sent 4 or more negative messages in a single month is considered a Flight Risk.  
  
**Steps to Implement:**Filter the original dataframe for only messages labeled "Negative".  
  
Group by form (employee email) and month.  
  
Count the number of negative messages per employee per month.  
  
Filter the groups where count ≥ 4 → mark those as Flight Risks.

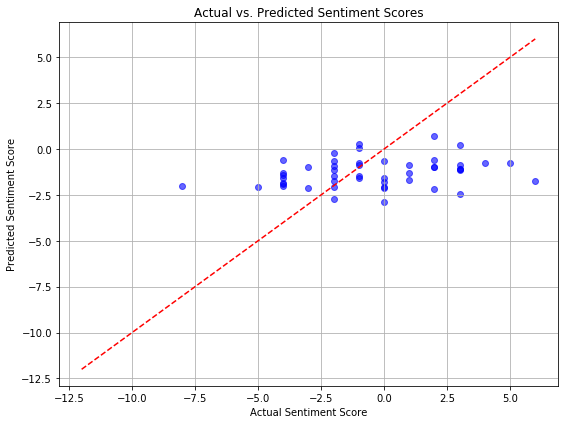
The employees with emails in the result above have more than 4 negative emails and are at risk of leaving the organization.



The employees with emails in the result above have more than 4 negative emails and are at risk of leaving the organization.

**Task 6: Predictive Modeling  
Objective:**Build a Linear Regression model to analyze and predict trends in employee sentiment over time.  
  
**Steps to Implement:**  
Feature Selection:  
To predict sentiment trends, we engineered the following independent variables:  
  
msg\_count: Number of messages sent by each employee per month.  
  
msg\_length: Average length of messages.  
  
word\_count: Average number of words per message.  
  
employee\_id: Encoded employee email addresses.  
  
month\_num: Timestamp of the message month (numeric format for modeling).  
  
Target Variable:  
  
sentiment\_score: Computed as the net monthly sentiment score per employee (Positive = +1, Negative = -1, Neutral = 0).  
  
**Data Preparation:**  
Parsed the date column and grouped the dataset by employee and month.  
  
Aggregated message features and assigned sentiment scores based on the labels.  
  
**Model Training:**  
Split the dataset into training and testing sets (80/20).  
  
Used Linear Regression from sklearn to model the relationship between features and sentiment scores.  
  
**Evaluation Metrics:**  
R² Score: Measures the proportion of variance in the target explained by the features.  
  
RMSE (Root Mean Squared Error): Evaluates the prediction accuracy of the model.





**Observations:**  
A sentiment labeling pipeline was successfully applied using a transformer-based model (distilbert-base-uncased-finetuned-sst-2-english), producing categorized messages as Positive, Negative, or Neutral.  
  
EDA revealed that the majority of employee messages were either Neutral or Positive, with relatively fewer Negative messages.  
  
Monthly sentiment scores were computed per employee, allowing us to rank top positive and negative contributors and flag potential flight risks.  
  
A linear regression model was developed using engineered features such as message count, average message length, and word count to predict sentiment trends.  
  
**Model Performance:**  
The regression model yielded a very low R² score (-0.016) and a relatively high RMSE (~2.91), indicating that the current features and model are insufficient for accurately predicting sentiment trends.  
  
The plot of predicted vs. actual values showed a tight horizontal clustering, reinforcing that the model was unable to learn meaningful relationships.

**Future Work:  
Feature Enhancement:**  
  
Incorporate more semantic features from the text using NLP techniques like:  
  
Sentiment polarity (TextBlob or VADER)  
  
TF-IDF vectorization  
  
Keyword frequency or emotional tone detection  
  
**Model Improvement:**  
Experiment with more powerful models such as Random Forest, XGBoost, or LSTM-based deep learning models for sequential patterns.  
  
**Data Expansion:**  
Train on a larger dataset across more employees and time periods to improve model generalization.  
  
**Real-Time Monitoring:**  
Extend the system to flag high-risk employees dynamically and alert HR using updated sentiment trends monthly or weekly.