# **Employee Sentiment Analysis - Final Report**

## **Project Overview**

**Objective:** Analyze internal employee messages to classify their sentiment as Positive, Neutral, or Negative and to track sentiment trends over time. Additionally, build a regression model to predict monthly sentiment scores for early identification of employee disengagement or potential flight risks.

**Dataset:** Internal message data, including message content, employee email, date, sentiment score, sentiment label, and derived features like polarity, subjectivity, word count, etc.

**Sentiment Labeling:** Used **Hugging Face transformer model**.

**Feature Engineering:** Extracted message features such as polarity, subjectivity, message length, and word count.

**Predictive Modeling:** Built a linear regression model with TF-IDF vectorized text plus engineered features to predict sentiment scores.

### **Exploratory Data Analysis (EDA)**

**Total Messages:** 2191

* **Sentiment Distribution:**
  + Positive
  + Neutral
  + Negative
* **Message Trends:**
  + Monthly volume of messages
  + Message length and word count distribution
* **Key Visuals:**
  + Line chart of sentiment over time
  + Bar chart of top employees by message count
  + Boxplots showing polarity/subjectivity per sentiment label

## **Employee Score Calculatio**

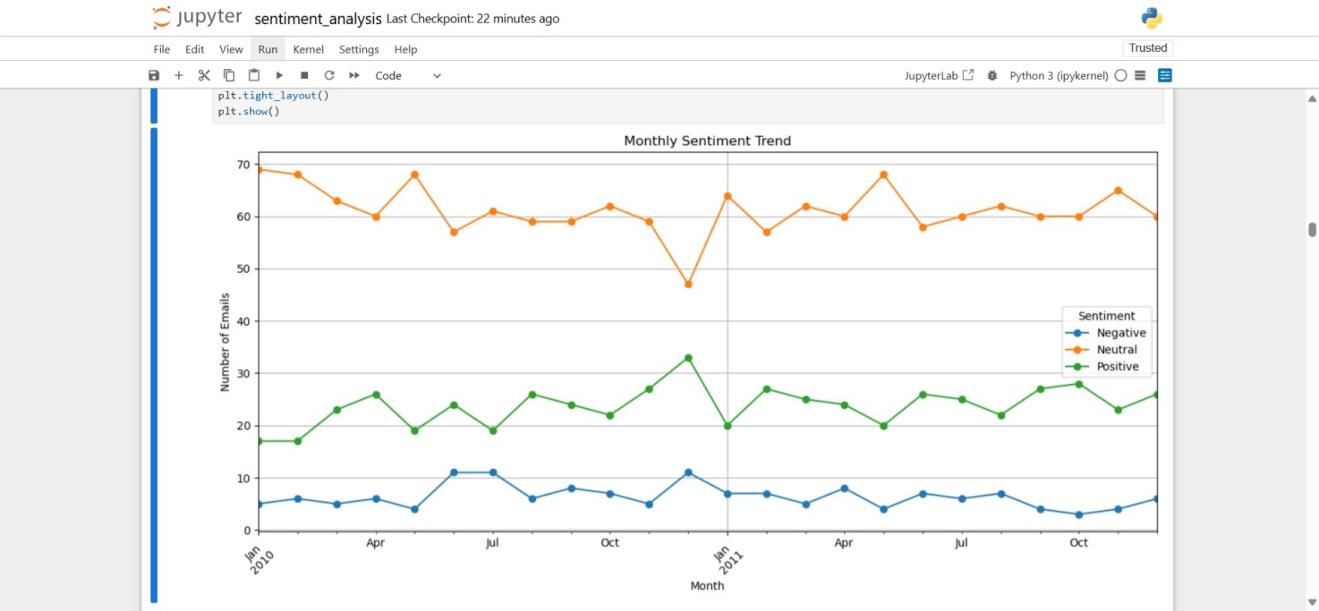
**Method Used:** TextBlob-based polarity scoring.

* Positive (Polarity > 0.1)
* Neutral (Polarity between -0.1 and 0.1)
* Negative (Polarity < -0.1)

**Final Column Added:** sentiment\_label (0: Negative, 1: Neutral, 2: Positive)

**Monthly Aggregation Approach:**

* Grouped messages by employee and month.
* Computed average sentiment\_score, message count, average polarity, subjectivity, etc.

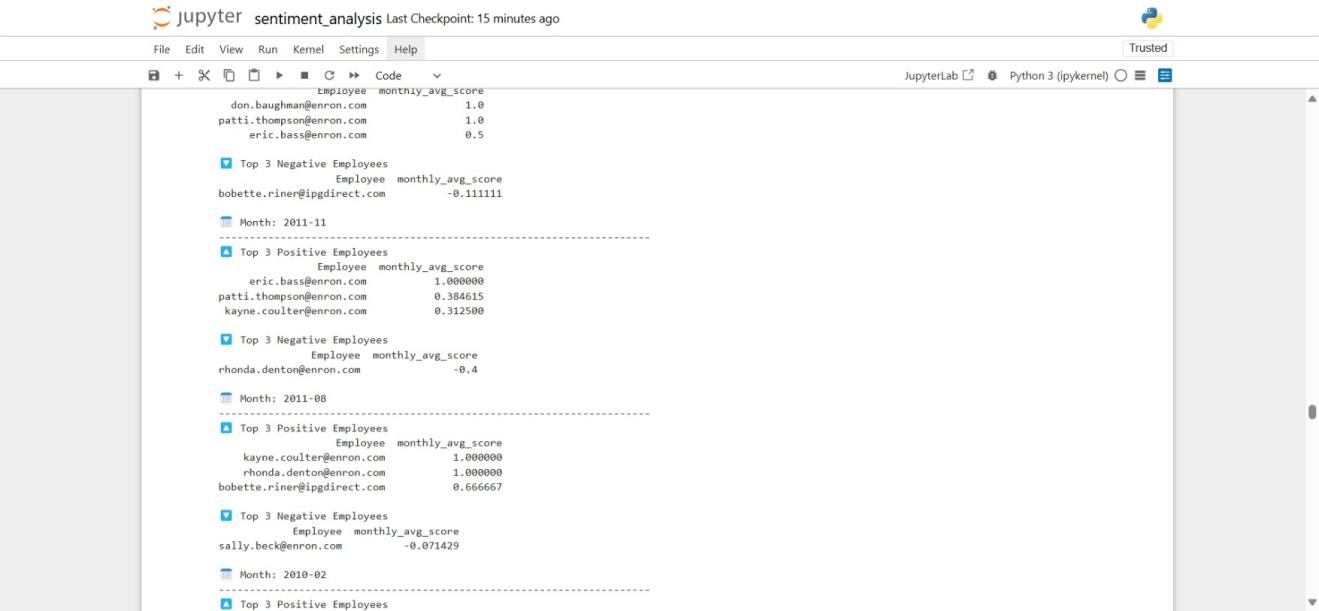


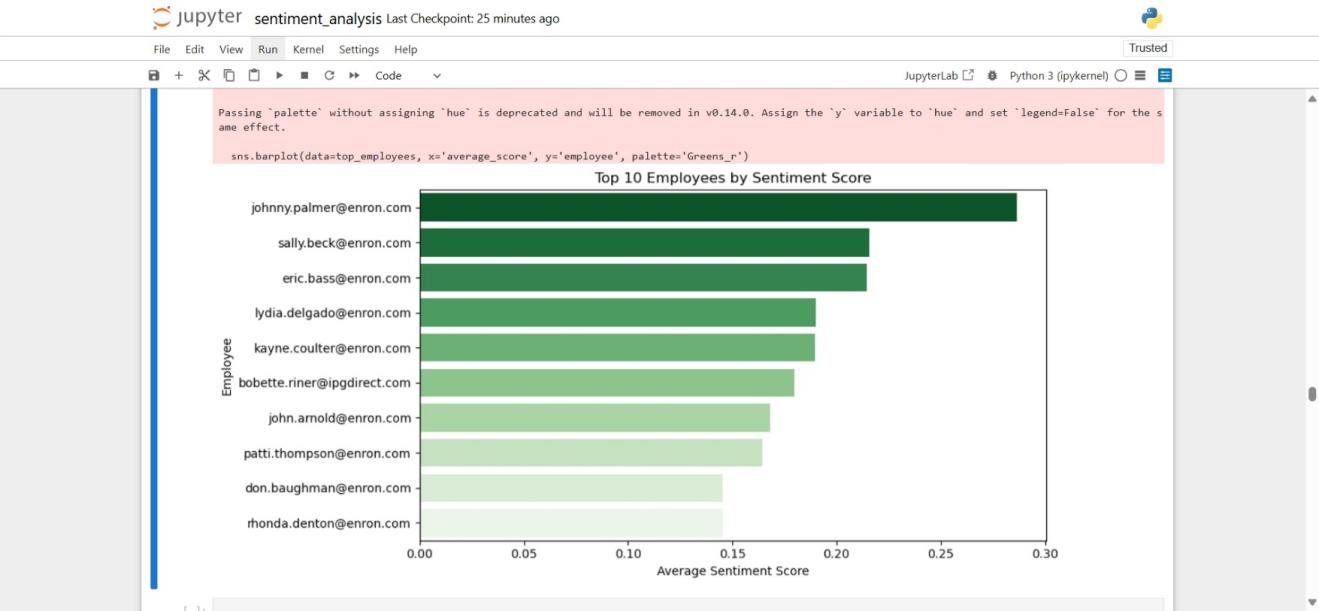
### **Employee Ranking**

Sentiment scores assigned as: Positive = +1, Neutral = 0, Negative = -1

Monthly aggregated scores computed per employee

Top 3 positive and negative employees were identified each month, ranked by score and alphabetically as tie-breaker.





### **Flight Risk Identification**

Defined flight risk as any employee sending 4 or more negative messages within any rolling 30-day period.

List of flagged employees extracted and visualized.

This flagging helps HR target potentially disengaged employees.

### **Predictive Modeling**

Model: Linear Regression predicting numeric sentiment score (0=Negative, 1=Neutral, 2=Positive)

Features: TF-IDF vectors + polarity + subjectivity + message length + word count

Performance:

RMSE: 0.416

R² Score: 0.424

Interpretation: Model explains approximately 42% of the variance in sentiment scores, showing promising predictive power for a baseline.

## **GitHub Structure**

Employee\_Sentiment\_Analysis/

├── test.csv

├── ranked\_employee\_sentiment.csv

│── labeled\_emails.csv

│── flight\_risks.csv

├── sentiment\_analysis.ipynb

├── visualization/

│ ├── Monthly sentiment trend.jpeg

│ └── monthly\_message\_volume.png

│ └── polarity\_by\_sentiment.png

│ └── Positive to negative sentiment ratio over time.jpeg

│ └── sentiment\_distribution.png

│ └── Top 10 Employee by Sentiment Analysis.jpeg

│ └── Top 10 senders by sentiment.jpeg

├── Final Report Sentiment.docx

├── README.md

└── requirements.txt

**Answers to FAQs**

**Q1. Risks of assigning fixed cutoff values for sentiment without justification**  
In my project, I didn’t manually assign arbitrary cutoffs like -0.1 to 0.1 for Neutral. Instead, I used the cardiffnlp/twitter-roberta-base-sentiment model, which already has its own classification thresholds. If I had hard-coded my own cutoffs without validating them on this Enron email dataset, I might have misclassified messages, especially since workplace communication can have subtler sentiment cues than tweets. Any thresholds I set would need to be tested against labeled samples from this domain.

**Q2. Problem of relying on only one open-source tool like TextBlob**  
I didn’t rely solely on TextBlob - I used a transformer model fine-tuned for sentiment (twitter-roberta-base-sentiment) because it handles context better than simple lexicon-based methods. However, even this model was trained on tweets, not corporate emails, so I still risk domain mismatch. To improve, I could compare results with another tool (like VADER) and manually validate a sample against human-labeled data.

**Q3. Why a well-made chart still needs interpretation**  
In my EDA, I created charts like sentiment distribution, monthly trends, and top senders by sentiment. For example, the monthly sentiment trend shows fluctuations, but without context (e.g., company events, layoffs, product launches), the reader wouldn’t know why those shifts happened. In my write-up, I made sure to interpret patterns - for instance, linking consistently negative scores to possible disengagement risks in Task 5.

**Q4. Issues with inventing metrics without clear rationale**  
When I mapped sentiment to scores (+1, 0, −1) and computed ratios like Positive-to-Negative Sentiment, I based them on a clear logic - positivity being beneficial, negativity being concerning. I didn’t just divide by a random number. If I were to create a more complex metric, I’d need to justify every factor so it reflects real employee sentiment patterns, not arbitrary math.

**Q5. Dangers of letting AI tools drive the entire analysis**  
I used AI for sentiment labeling but didn’t blindly trust its output. I still cleaned the text, checked sentiment distributions, and validated trends against known behavior (e.g., verifying top positive/negative employees month-by-month). If I had let the AI run everything without checking, sarcastic or context-specific language in emails could have been misclassified without me noticing.

**Q6. Why feature selection in predictive modelling must be thoughtful**  
In Task 6, I used TF-IDF features from the email body to predict sentiment scores. I didn’t just throw in unrelated features like “email font size” or “attachment count” - only text content that directly affects sentiment. If I added unrelated variables, it might have diluted my model’s accuracy and interpretability.

**Q7. Why printing only R² and MSE isn’t enough**  
I reported both RMSE (~0.416) and R² (~0.424) for my predictive model. RMSE gave me an idea of average prediction error in sentiment score units, while R² showed variance explained. Looking at just one metric would be misleading - for instance, a high R² with a large RMSE could still mean predictions are far from actual values.

**Q8. Why cross-verifying AI-generated outputs is essential**  
Even though my sentiment labels came from a transformer model, I checked their reasonableness through EDA - for example, comparing the top frequent words per sentiment and verifying that “unfortunately” appeared in negatives. Without these checks, I might have missed systematic errors, like the model tagging all formal “thank you” notes as Neutral.

**Q9. What makes analysis more than just charts and models**  
My project wasn’t just bar plots and regression - I linked sentiment scores to employee rankings, risk identification, and predictions. For instance, Task 5’s declining sentiment slopes were tied to possible disengagement, making it actionable for HR. Without those explanations, the visuals would just be decoration.

**Q10. How I used AI without losing my own analytical thinking**  
I treated the transformer model as a helper, not the final authority. I designed the preprocessing pipeline, decided on sentiment scoring logic, explored trends, and selected modeling features. The AI didn’t decide which questions to ask - I did. This kept the analysis aligned with business goals instead of just model output.

### **Conclusions and Recommendations**

* Majority of employees showed consistent or increasing sentiment.
* Declining sentiment for specific individuals may indicate potential burnout or disengagement.
* Sentiment tracking can be used for proactive engagement monitoring.
* Employees flagged as flight risks should be reviewed for support/intervention.
* Regular sentiment scoring can feed into HR dashboards for monitoring
* Model performance indicates strong predictive ability.
* Sentiment tracking can support HR intervention planning.
* Model can be improved with more advanced NLP techniques (e.g., BERT embeddings).