Computing Sentiment Analysis on Emails using AI

Karthik Anne
Sai Gangu
Tejas Kabra
Hannah Li
Anshul Mago
Shafin Ula
Keshav Vidjeabaskaran

Table of contents

01 The Team

02 The Problem

03 Our Solution

04 Challenges/Learnings

The Team

Meet our team

Keshav

Scrum Master

Senior at Dulles HS

Shafin Ula

Java/Python Developer

Senior at Dulles HS

Sai Gangu

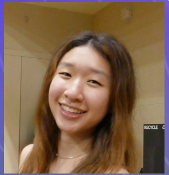
Architect

Junior at Dulles HS

Hannah

Java/Python Developer

Junior at Clements HS



Anshul

Java/Python Developer

Senior at Clements HS

Tejas

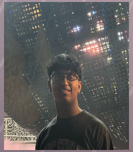
Java/Python Developer

Senior at Seven Lakes HS

Karthik Anne

Business Analyst

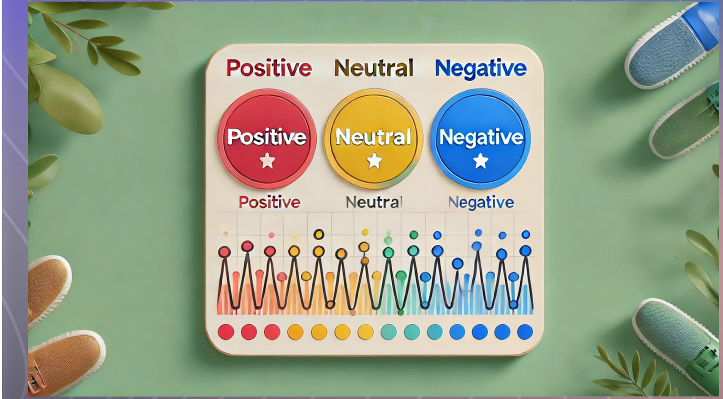
Senior at Cypress Ranch HS



Problem

Problem Statement

In today's digital workplace, the high volume of emails can hide important sentiments, leading to misunderstandings and missed opportunities. Traditional email systems can't effectively analyze emotional tone, making it hard to capture customer sentiment and satisfaction. Our project aims to analyze email sentiment using AI to uncover insights, improve communication, and foster a positive work culture.



Solution

Solution

Our Model

Using OpenAI, we created a model that detects sentiments given an email. We tested the NLP model using emails generated from a variety of AI models.

Our Design

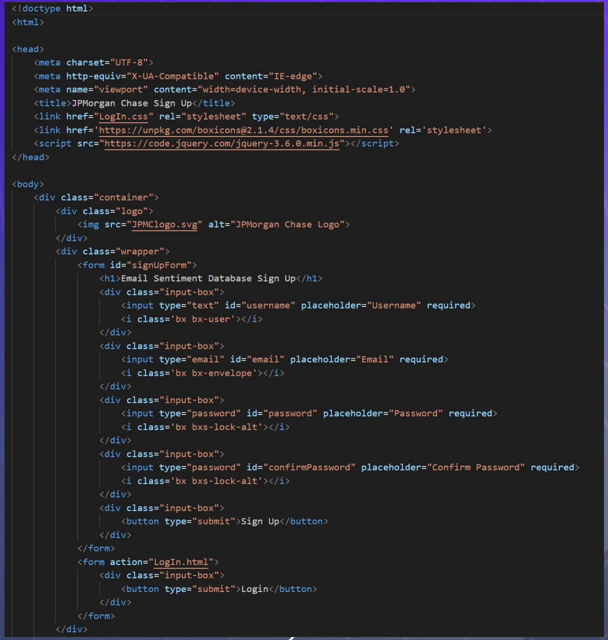
Implementing HTML, CSS, and Javascript for the frontend and python for the backend, we created a website designed to streamline this process.

Our IDE

VSCode is a customizable, extensible editor with built-in debugging, Git integration, and intelligent code completion.

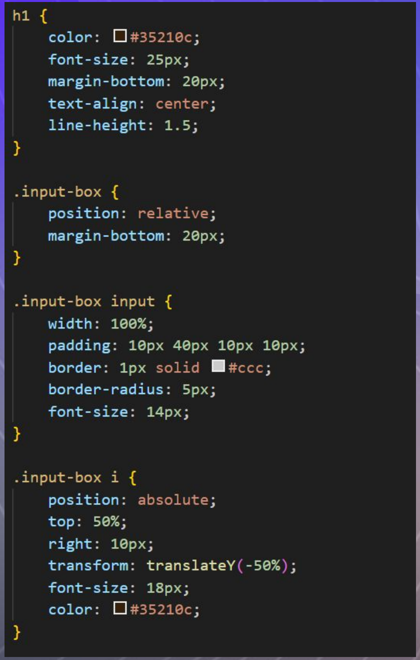
Frontend

HTML Sample Code



```html
<!doctype html>
<html>

<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE-edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>JPMorgan Chase Sign Up</title>
    <link href="LogIn.css" rel="stylesheet" type="text/css">
    <link href='https://unpkg.com/boxicons@2.1.4/css/boxicons.min.css' rel='stylesheet'>
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>

<body>
    <div class="container">
        <div class="logo">
            <img src="JPMClogo.svg" alt="JPMorgan Chase Logo">
        </div>
        <div class="wrapper">
            <form id="signUpForm">
                <h1>Email Sentiment Database Sign Up</h1>
                <div class="input-box">
                    <input type="text" id="username" placeholder="Username" required>
                    <i class='bx bx-user'></i>
                </div>
                <div class="input-box">
                    <input type="email" id="email" placeholder="Email" required>
                    <i class='bx bx-envelope'></i>
                </div>
                <div class="input-box">
                    <input type="password" id="password" placeholder="Password" required>
                    <i class='bx bxs-lock-alt'></i>
                </div>
                <div class="input-box">
                    <input type="password" id="confirmPassword" placeholder="Confirm Password" required>
                    <i class='bx bxs-lock-alt'></i>
                </div>
                <div class="input-box">
                    <button type="submit">Sign Up</button>
                </div>
            </form>
            <form action="LogIn.html">
                <div class="input-box">
                    <button type="submit">Login</button>
                </div>
            </form>
        </div>
```

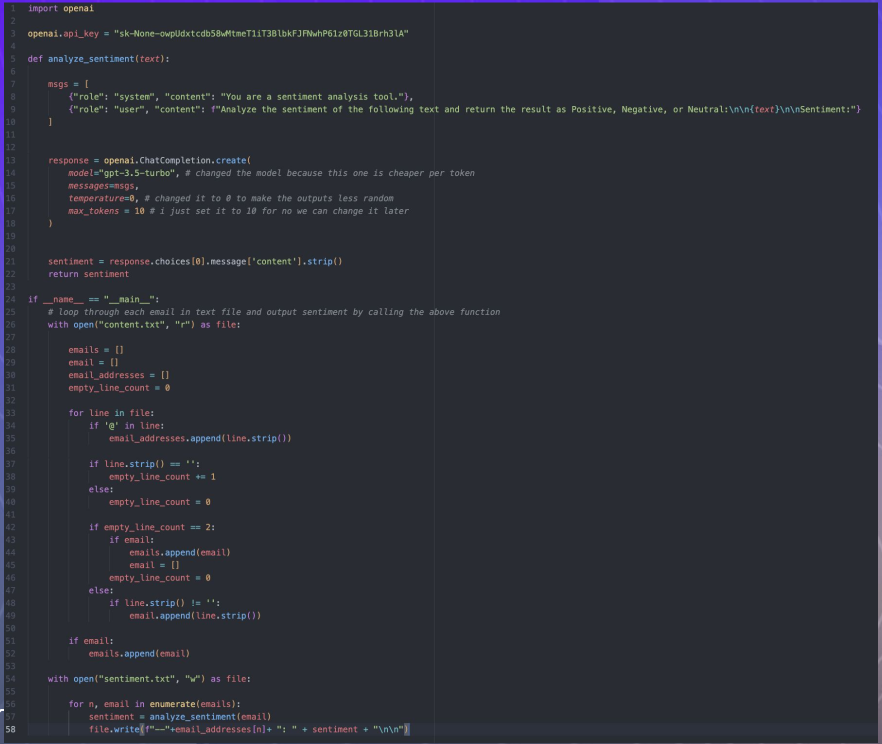
CSS Sample Code



```css
h1 {
    color: #35210c;
    font-size: 25px;
    margin-bottom: 20px;
    text-align: center;
    line-height: 1.5;
}

.input-box {
    position: relative;
    margin-bottom: 20px;
}

.input-box input {
    width: 100%;
    padding: 10px 40px 10px 10px;
    border: 1px solid #ccc;
    border-radius: 5px;
    font-size: 14px;
}

.input-box i {
    position: absolute;
    top: 50%;
    right: 10px;
    transform: translateY(-50%);
    font-size: 18px;
    color: #35210c;
}
```

Backend

Sentiment Analysis Python code



```python
import openai

openai.api_key = "sk-None-owpUdxtcdb58wMtmeT1iT3BlbkFJFNwhP61z0TGL31Brh3lA"

def analyze_sentiment(text):

    msgs = [
        {"role": "system", "content": "You are a sentiment analysis tool."},
        {"role": "user", "content": f"Analyze the sentiment of the following text and return the result as Positive, Negative, or Neutral:\n\n{text}\n\nSentiment:"}
    ]


    response = openai.ChatCompletion.create(
        model="gpt-3.5-turbo", # changed the model because this one is cheaper per token
        messages=msgs,
        temperature=0, # changed it to 0 to make the outputs less random
        max_tokens = 10 # i just set it to 10 for no we can change it later
    )


    sentiment = response.choices[0].message['content'].strip()
    return sentiment

if __name__ == "__main__":
    # loop through each email in text file and output sentiment by calling the above function
    with open("content.txt", "r") as file:

        emails = []
        email = []
        email_addresses = []
        empty_line_count = 0

        for line in file:
            if '@' in line:
                email_addresses.append(line.strip())

            if line.strip() == '':
                empty_line_count += 1
            else:
                empty_line_count = 0

            if empty_line_count == 2:
                if email:
                    emails.append(email)
                    email = []
                empty_line_count = 0
            else:
                if line.strip() != '':
                    email.append(line.strip())

        if email:
            emails.append(email)

    with open("sentiment.txt", "w") as file:

        for n, email in enumerate(emails):
            sentiment = analyze_sentiment(email)
            file.write(f"--"+email_addresses[n]+ ": " + sentiment + "\n\n")
```

Challenges Faced

Challenges Faced

Integration Issues

- Gmail and Yahoo don't seem to allow connection to Python.
- Issues with Google not allowing a user to sign in through the provided API.

Technical Errors

- Getting an error when trying to clone the shared repository.
- Getting a ModuleNotFoundError for OpenAI when trying to run the script.

Decision-Making

- Deciding whether to use OpenAI or train our own model.
- Struggling on weighing out the pros and cons of using different AI services.

Overcoming Challenges

Integration Issues

- Must enable “Less Secure Apps” in Gmail
- Turn on two-factor authentication
- Generate an app password in the security section

Technical Errors

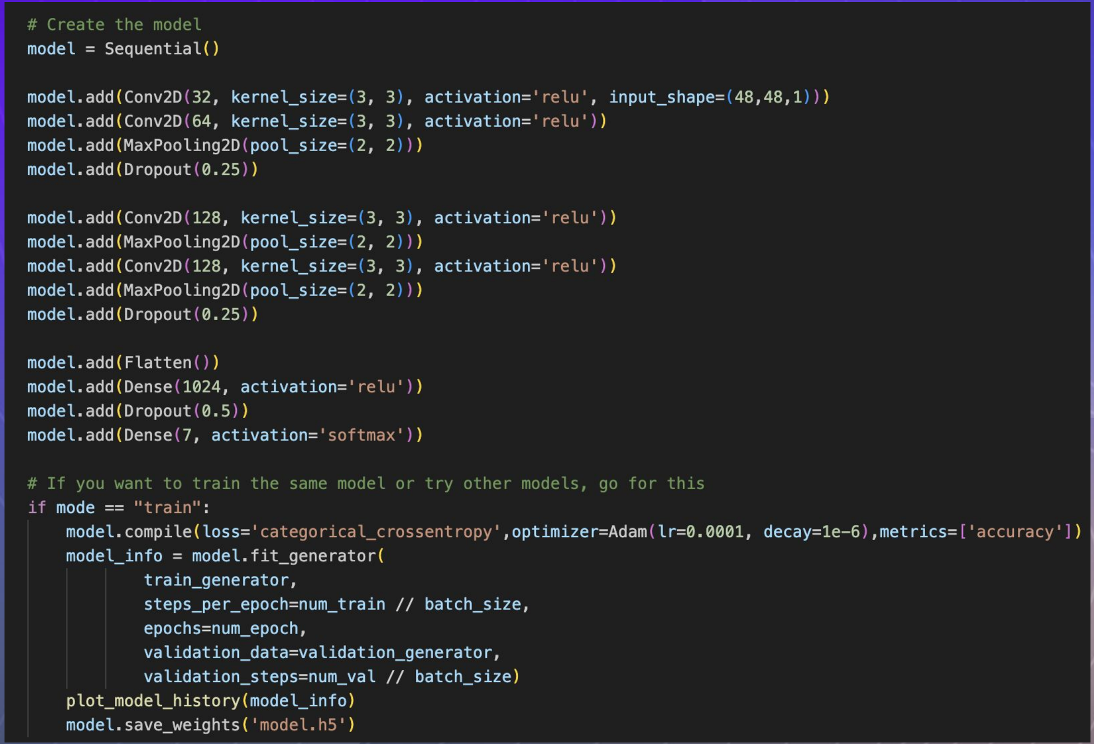
- Problem for Mac Users; solved the issue by using GitHub Desktop
- Solved by making sure openAI is downloaded and used on the same version of python on your IDE

Decision-Making

- After evaluating pros and cons for each option, we collectively decided on using OpenAI over training our own model as it would be more efficient

Project Demo

Face Detection Model



```
# Create the model
model = Sequential()

model.add(Conv2D(32, kernel_size=(3, 3), activation='relu', input_shape=(48,48,1)))
model.add(Conv2D(64, kernel_size=(3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Conv2D(128, kernel_size=(3, 3), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.25))

model.add(Flatten())
model.add(Dense(1024, activation='relu'))
model.add(Dropout(0.5))
model.add(Dense(7, activation='softmax'))

# If you want to train the same model or try other models, go for this
if mode == "train":
    model.compile(loss='categorical_crossentropy',optimizer=Adam(lr=0.0001, decay=1e-6),metrics=['accuracy'])
    model_info = model.fit_generator(
            train_generator,
            steps_per_epoch=num_train // batch_size,
            epochs=num_epoch,
            validation_data=validation_generator,
            validation_steps=num_val // batch_size)
    plot_model_history(model_info)
    model.save_weights('model.h5')
```

Thank you