

Cold Email Generator

Automating the Creation of Personalized Cold

By:

Group No:12

Name	Roll No
Pranav Ganesh	20211CAI0062
Dinesh Kumar K	20211CAI0074
Gowtham Reddy V	20211CAI0151
Sane Venkata Charan	20211CAI0174
Cherukuru Pavan	20211CAI0096

Abstract

- The Cold Email Generator project aims to automate the creation of personalized cold emails by leveraging predefined templates and user-provided data. This tool integrates AI for fetching and summarizing relevant information, ensuring that the generated emails are not only personalized but also contextually relevant. The project utilizes the LangChain and OpenAI libraries to achieve these objectives.

Objectives

- Automate the process of generating personalized cold emails.
- Utilize predefined templates for various types of cold emails.
- Collect and incorporate user input to dynamically fill in the templates.
- Leverage AI to fetch and summarize relevant information to enhance the emails.

Workflow of Cold Email Generator

- **Start**
- **Get Email Subject** (from user input)
- **Select Appropriate Template** (based on subject keywords)
- **Get Personalization Data** (from user input)
- **Generate Email Using Template** (fill template with user data)
- **Display Generated Email**
- **End**

Methodologies Used

1. Library Imports:

- **openai**: This library is used for AI-based text generation. The `openai.Completion.create()` function is utilized to generate summaries and other text completions.
- **requests**: This library is used for making HTTP requests. The `requests.get()` function is used to fetch Wikipedia articles.
- **langchain.chains.LLMChain**: This library handles chains. It's used to define and manage the sequence of operations in the project.
- **langchain.prompts.PromptTemplate**: This library handles templates. It's used to define and manage email templates.

Methodologies Used

► 2. Email Templates:

- **Collaboration:** Template for collaboration emails.
- **Job Opportunity:** Template for job opportunity emails.
- **Product Inquiry:** Template for product inquiry emails.
- The templates are predefined with placeholders for dynamic data.

Methodologies Used

▶ 3. Fetching and Summarizing Information

- ▶ Uses a function that takes a URL, makes a GET request to fetch the article, and returns the text content.

▶ Summarize Text Using OpenAI:

- ▶ Uses a function that uses the OpenAI API to summarize a given text within a specified word limit.

▶ 4. User Input and Email Generation:

▶ Get User Input:

- ▶ Uses a function that prompts the user for input and returns the entered value.

▶ Generate Email Using Template:

- ▶ Uses a function that fills in the template with user-provided data and returns the generated email.

▶ 5. Collecting Personalization Data:

- ▶ Uses a function that collects common data such as the email subject, recipient's name, sender's name, company name, and sender's position. It then collects specific data based on the selected template, including industry, product/service, and value proposition.

Methodologies Used

► 6. Email Generation Chain:

► Chain Class:

- Uses class that defines the email generation chain, which initializes with a template and runs by generating an email using the provided data.

► 7. Main Function:

• Running the Email Generator:

- This block of code handles the main logic:
 - It prompts the user to enter the email subject.
 - It selects the appropriate template based on the subject.
 - It collects personalization data based on the selected template.
 - It creates and runs the email generation chain.
 - It prints the generated email.

Results

```
→ Enter the email subject: job
Enter the email subject: job
Enter the recipient's name: ram
Enter your name: ravi
Enter your company name: presidency
Enter your job title: managetr
Enter the recipient's industry: software
Enter the product/service you offer: software application
Enter the value proposition: software
```

Generated Email:

Subject: job

Hi ram,

I hope this email finds you well. My name is ravi from presidency. We are currently expanding our team and your experience in software caught our attention.

At presidency, we are committed to software. We believe your expertise in software application would be a valuable addition to our team.

I would love to discuss this opportunity with you further. Could we schedule a call at your earliest convenience?

Best regards,
ravi
managetr
presidency

Future Work:

- Incorporate more advanced natural language processing techniques.
- Expand the range of email templates.
- Implement user-friendly interface for broader usability.

Conclusion

- ▶ The Cold Email Generator project automates the process of generating personalized cold emails. By leveraging predefined templates, user input, and AI for summarizing information, the project ensures that the generated emails are personalized and contextually relevant. Future improvements could include expanding the range of templates and incorporating more advanced natural language processing techniques.

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic design. The shapes are layered, with some appearing more prominent than others, and they extend from the right and bottom edges towards the center.

ThankYou