Reka take home assignment: Sales prospecting bot

Background

Outbound sales representative

An outbound sales representative is responsible for proactively reaching out to potential customers to generate leads (prospecting) and close sales.

Difficulty with prospecting

Prospecting can be challenging because it requires identifying and reaching out to the right potential customers, often with limited information about their specific needs or readiness to buy. Capturing attention in a crowded marketplace demands persistence, creativity, and personalized communication. Additionally, the high likelihood of rejection and the time-intensive nature of the process can be discouraging and require strong resilience.

Goal of this assignment

For this assignment, your task is to **develop several backend components for a prospecting bot** designed to help outbound sales representatives prospect more efficiently and effectively. **The user journey for this bot is outlined below** (you are not required to implement all these steps for this assignment):

1. During the onboarding process, the user visits a website and provides all the necessary information for the bot to operate. This includes details about the user's company, the product they are selling, and any other relevant information about their ideal customer profile (ICP).
2. Using the information from the onboarding process, the bot periodically searches the internet to identify potential leads and gathers detailed research on each one.
3. Leveraging the research from the previous step, the bot drafts a personalized email using a large language model (LLM) and sends it to the identified leads.

Assignment Part 1 – Design the database schema

Building on the description from the previous section, your first task is to **design the database schema for the prospecting bot**. The database should encapsulate the entire user journey outlined earlier. Specifically:

1. The schema should include all the information provided during the onboarding process. You are encouraged to research and apply your best judgment in defining what details should be collected during onboarding. Approach this with the perspective of the user—real outbound sales representatives.
2. The schema should also include tables required for the bot's functionality as described in the user journey. **You are free to begin with part 2 of this assignment first and refine the database design as you progress**.

Deliverable

- Database schema using any database of your choice. This schema will also be utilized in part 2 of the assignment.
- Include a description of your database design in the README file that will accompany your submission.

Assignment Part 2 – Lead research "cron job"

Implement Step 2 of the user journey **in python**:

"Using the information from the onboarding process, the bot periodically searches the internet to identify potential leads and gathers detailed research on each one."

This step should be implemented like a "cron job" that periodically searches for leads indefinitely.

You can use these additional tools in your deliverable

1. You may use the following mock function as a lead search API. i.e., The following function can be called in your code for lead search. You don't have to implement it, but you can mock it for testing.

```
def search_leads(keywords: list[str]) -> list[dict]:
""" Search for leads given a list of keywords.
Args:
  keywords: list of keywords relevant to the lead.
```

```
Returns:
 A list of dictionaries with the following fields:
   "company_name": the name of the company.
   "company_website": the website of the lead's company.
   "lead_email": the email of this lead.
   "lead_name": name of the lead
"""

        # Assume this function is implemented.
```

2. Groq API for LLM inference: https://console.groq.com/docs/api-reference#chat-create

Deliverable

- Provide a runnable implementation of the lead research "cron job" in Python.
    - **Input:** The cron job should use the `search_leads` function (defined above) to generate a list of potential leads.
    - It should leverage LLMs (via the Groq API, as defined above) to perform research on each lead.
        - **Hint:** Since you have access to the lead's company's website, you can pass HTML pages directly into the LLM's prompt context for analysis.
    - You are encouraged to use your best judgment to determine how the bot should gather and analyze information about the lead and their company. Keep in mind that the ultimate goal is for the bot to send a personalized email to the lead.
    - **Output:** The output of the cron job should be data written to the database, following the schema you defined in the previous part of the assignment.
    - **Technical stack:** Feel free to use any libraries/SDKs/APIs you are familiar with. (e.g., AWS SQS)

Evaluation Criteria:
Your solution will be assessed based on its **robustness for production deployment** and its ability to scale effectively. For this assignment, your code should either 1) runnable using AWS services or 2) runnable locally, but include a short description on how you would adapt it to be deployed on AWS.

Submission

Compress your solution into a zip file and send it to eric@reka.ai once completed.

Once your solution is reviewed, we will schedule a follow up meeting to discuss your solution.