

Houston Holman

4/14/23

Chatbot Project Report

Overview

Advertisement Delivery Now (ADeN) is a chatbot that injects subtle advertising into each of its responses. After learning the user's name and the list of companies or products they enjoy, ADeN answers their questions while advertising one of the stated companies or products. The idea behind this is that it might be possible to subsidize the server costs of running chatbots through the placement of advertisements. (Note: I am ethically against this actually becoming a real tool. This project was made for entertainment purposes. I was really curious to see what a chatbot made for a capitalistic dystopia would look like.)

How to Run

1. Install Python 3.10.11 (does not work with Python 3.11 due to bug with spaCy)
2. Navigate to the project directory using cmd
3. Open app.py in an editor and paste in your OpenAI API Key
 - a. If you need one, I can provide mine. Reach out at houston@utdallas.edu
4. Run `pip install --upgrade pip`
5. Run `pip install -U pip setuptools wheel`
6. Run `pip install openai`
7. Run `pip install -U spacy`
8. Run `pip install spacy_download`
9. Run `python app.py`

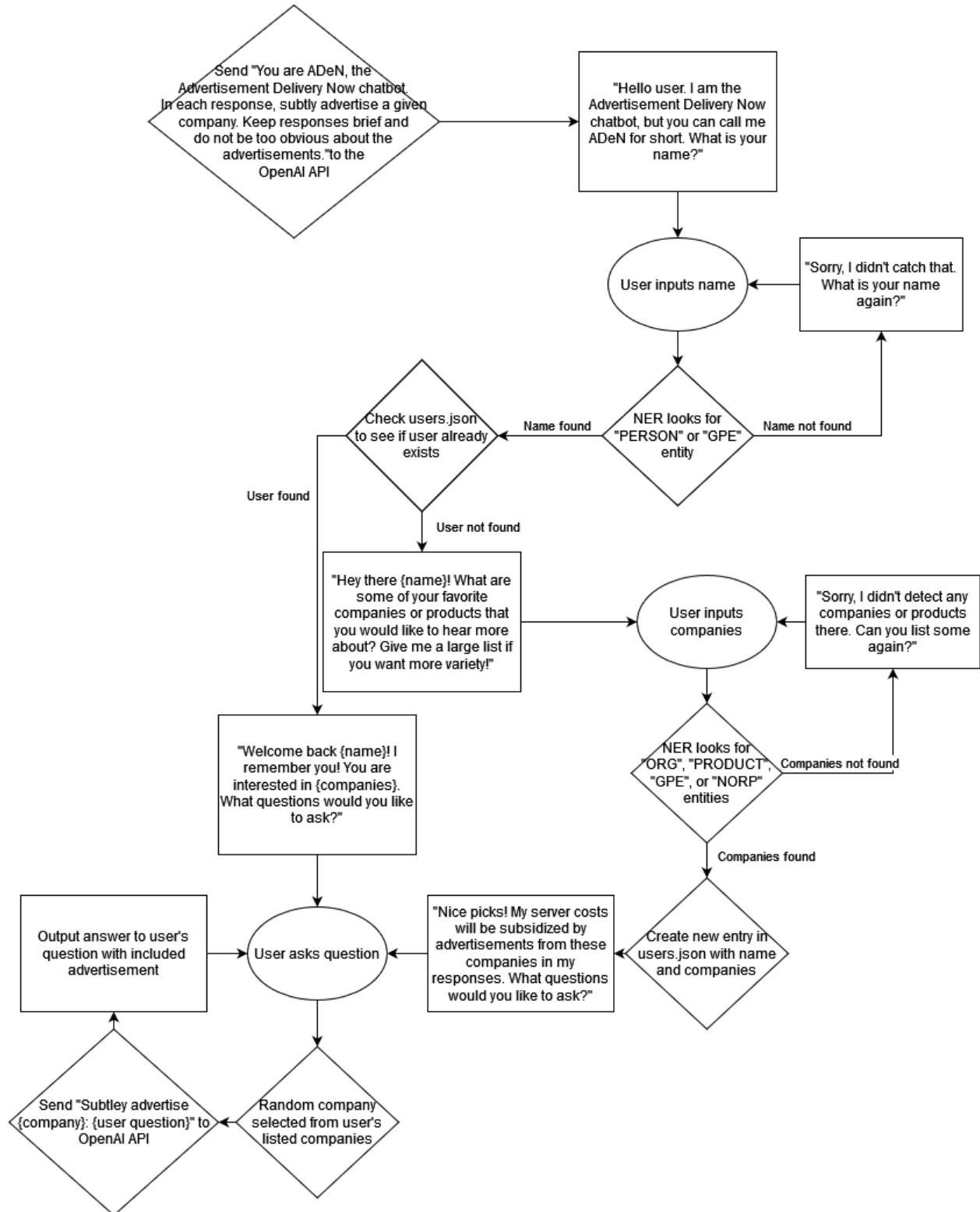
Development Challenges

My original plan with this project was to train my own model based on the r/utdallas subreddit using TensorFlow. After a day of training, the model was still incoherent. My options were to either let it keep training and hope it improved or start from scratch using a new method. Since the model was showing no improvement as time went on, I eventually was forced to start over. I decided to use the OpenAI API because of its ease of use, allowing me to focus on implementing Named Entity Recognition instead of spending all my time training a model.

System Description

ADeN's response generation and knowledge base is handled by the OpenAI API. This API operates as a general-purpose "text in, text out" interface, which enables it to take on any English language task. Before input is sent to the API, Named Entity Recognition (NER) is used to extract relevant details. This is accomplished through the use of the spaCy package. NER is first used to parse out the user's name from their first message. The program scans through a list of entities extracted from the first message and pulls out the first entity labeled "PERSON" or "GPE" (countries, cities, states). "GPE" was added because spaCy was classifying my name as a city. NER is used again to gather the list of companies or products. It checks for entities with the tags "ORG" (organization), "PRODUCT", "GPE" (countries, cities, states), "NORP" (nationality, religious, or political groups). This selection of tags was chosen after testing revealed that many companies were wrongly categorized (the funniest being Reddit classified as "NORP"). To fix this problem, I went with a more inclusive approach, allowing advertisements for many types of entities, including countries and even religions.

Logic Diagram



Sample Dialog Interactions

```
Hello user. I am the Advertisement Delivery Now chatbot, but you can
call me ADeN for short. What is your name?
>My name is Steven
Hey there Steven! What are some of your favorite companies or
products that you would like to hear more about? Give me a large list
if you want more variety!
>Some of my favorite companies include Microsoft, Apple, Reddit,
Tesla, Disney, and Taco Bell.
Nice picks! My server costs will be subsidized by advertisements from
these companies in my responses. What questions would you like to
ask?
>What should I do today?
There are so many things you could do today, from catching up on your
reading to going for a hike. If you're looking for inspiration, I
highly recommend checking out some of the trending posts on Reddit.
You may even discover an activity that you'd never thought of before!
>What should I eat for lunch?
Hmm, that's a tough one. How about something magical and fun for
lunch? You could always visit a Disney park and experience the
delight of eating a Mickey Mouse shaped snack or a character themed
meal.
```

Live Lookup Appendix

ADeN makes use of OpenAI's gpt-3.5-turbo model for looking up data. It is based on GPT-3, a model trained on 429 billion tokens from web crawlers, 67 billion tokens from books, and 3 billion tokens from Wikipedia. OpenAI's version of GPT-3, called InstructGPT, includes safeguards that cause it to generate less made up facts and toxic content.

User Model Appendix

User models are stored in an array located in `users.json`. Here is an example of some created user models:

```
"users": [  
  {  
    "name": "Jim",  
    "companies": [  
      "Tesla",  
      "FritoLay",  
      "CocaCola",  
      "Microsoft",  
      "Netflix",  
      "Canada",  
      "beer"  
    ]  
  },  
  {  
    "name": "Steven",  
    "companies": [  
      "Microsoft",  
      "Apple",  
      "Reddit",  
      "Tesla",  
      "Disney",  
      "Taco Bell"  
    ]  
  }  
]
```

Evaluation

➤ Strengths

- Overall informativeness and coherence in responses
 - Due to GPT 3.5's vast knowledge base
- Ability to pick out names and companies/products
 - Due to spaCy's ability to accurately name entities

➤ Weaknesses

- Occasional miscategorization of entities
 - spaCy occasionally will classify a company as not an entity or mislabel it. To combat this, I had the chatbot accept a large range of labels but it still is not perfect
- Sometimes advertisements are a bit on the nose
 - The goal was to make a bot that subtly hides advertisements in its responses. This works sometimes, but other times the bot has trouble integrating advertisements into its answers. One example of this can be seen here:

```
>Im hungry  
If you're looking for an eco-friendly and high-tech way to satisfy  
your hunger, the future is here with Tesla's Cybertruck.
```