Grover - Tech Produkte mieten

Conversational Application for Grover

Authors:
Ioannis Gerochristos

Supervisor:
Igor Nikolaev

July 31, 2017

1 Introduction

The purpose of this short document is to describe the conversational application that I developed as a homework for the interview process of Grover. We will begin by giving a description of how I encountered the input data and the whole problem, specifying any assumptions that I made during the implementation process. Afterwards, I will present the logic of the application and try to guide the reader through my thought process. Finally, I shall present some example runs of the application, stating the outcomes for the potential customer.

2 Input Data

I received the data in a content of an email; as a result, I had in my hands fully unstructured content. I thought that it was appropriate (in order to exploit the advantages of Python as well) to create an "input.csv" file, containing 26 rows (25 for each of the entries, and the first one for the headers of the file). After carefully examining my data, I came to the conclusion that an appropriate delimiter for my csv file would be the semicolon (";") character, since it was not appearing at all in the entirety of the file. So, I proceeded in the creation of the corresponding .csv file.

3 Thought Process

The goal of this short project was to propose an appropriate item to a potential customer, taking into consideration his overall preferences, and producing as output the item (or set of appropriate items) along with their subscription plan prices. I will not try to break down my thought process while I was tackling the problem in hand.

While roaming through the data, I found out that one could break down the input data according to their "Category. This would result in 6 categories. In the end though, I decided to unify the categories of "Gaming & VR" and "Drones", since I estimated that they should be attracting the same target group of people in general, so with a more general question we could end

up getting more customers to pay for the services. Each of these categories appears to be more or less separated from each other. In other words, a customer looking for a mobile phone, is much probably interested in mobile phones and not in Smart Home equipment (even in case he was, he could simply re-run the application from scratch). So I thought that it was a nice initial step so as to initially separate my items. So now we wind up with 5 categories:

3.1 Assumptions

In my initial configuration, I would always return one unique item to the client. In the end, I figures out that if two items have the same price (thus bringing to the company the same amount of money) and belong in the same answer set, they should be both returned to the client, and then he would have to decide which one suits him better (This is the case for the VR, for the non-iOS laptops, for the sport smart-watches and for the robotic vacuum cleaners, since in all of these cases the items involved have the same price and are actually more or less identical, so it was hard to even compare them). One suggestion would be to provide some links for all of the related products in the webpage interface, when such a case occurs.

3.2 Problem break-down

I will now provide an analytical brake-down of the problem and the categorizations that I came up with.

- **Phones & Tablets**: This category contains 6 items. A very natural distinction between them, is of course the operating system. Normally, people are almost certain whether they are going for an Android or for an Apple device. Thus, my question was of the form: "Which of the 2 operating systems do you favor?", resulting in 2 categories:
  - **Apple**: In our input there are 3 Apple items of these categories. After some thought, I figured out that the next question should be of the form: "Do you prefer a small or a large storage space", since storage space is a feature that is extremely important for

some clients, but very unimportant for others. Again we get 2 separated categories:

    * **Small storage Apple devices**: There is only one item in the data corresponding to this description, so we return it to the client.
    * **Big storage Apple devices**: There are 2 phones belonging to this category. Since we are talking about the "iPhone7" vs "iPhone7 Plus", we are simply asking the client: "if he prefers larger or smaller screen sizes", since it is pretty clear difference between the 2 models, getting:
        · **Big storage Apple devices with bigger screen**: Only 1 phone, so return it to the customer
        · **Big storage Apple devices with smaller screen**: Only 1 phone, so return it to the customer

  – **Samsung**: In the case of the "Samsung" phones, since we have only two entries, we are asking the same question about the sizes of the screens, ending up with:

    * **Samsung devices with bigger screen**: Only 1 phone, so return it to the customer
    * **Samsung devices with smaller screen**: Only 1 phone, so return it to the customer

- **Gaming & Gadgets**: If the user selects this choice, then we immediately ask him "If he is more interested in VRs or Drones", resulting in 2 categories:

  – **Drones**: In the case of the "Drones" option, we simply ask the user if he: "cares for a more powerful, more robust and increased autonomy model, or he could settle for less". We end up with the categories:

    * **Sophisticated Drones**: Only 1 entry, so return it to the user **Simpler Drones**: Only 1 entry, so return it to the user

  – **Gaming & VRs**: In the case of the VRs, we return to the user both of the available systems, for the reasons that were described in the "Assumptions" subsection.

- **Computing**: Regarding the computers, again we can see 2 distinct categories, namely the iOS and the non-iOS(Windows) laptops. Correspondingly, this is the follow-up question that we pose to the user, namely if "He is a fan of the Windows or the iOS operating system". Upon his reply, we get:
  - **iOS Laptops**: Within the iOS laptops, we distinguish 2 categories, that users can differentiate from each other, according to their needs:
    - **iOS laptops with a larger display but with less memory (RAM)**: Among these 2 products, ultimately I provide a categorization according to their CPUs:
      - **iOS laptops with a larger display but with less memory (RAM) and an i7 processor**: Only 1 entry, so return it to the user
      - **iOS laptops with a larger display but with less memory (RAM) and an i5 processor**: Only 1 entry, so return it to the user
    - **iOS laptops with a smaller display but with more memory (RAM)**: In this case, we categorize according to the hard disk space of the models:
      - **iOS laptops with a smaller display but with more memory (RAM) and a larger hard drive**: Only 1 entry, so return it to the user
      - **iOS laptops with a smaller display but with more memory (RAM) and a smaller hard drive**: Only 1 entry, so return it to the user
  - **Non-iOS(Windows) Laptops**: In that case, we return both of the products from the input, for the reasons stated in the "Assumptions" subsection
- **Wearables**: In the case of the "Wearables", I actually had to do some research on the products in hand, since I had very small familiarity with such gadgets. As a result of it, I realized that 2 of the smart-watches presented are more suitable for sports activities and are also promoted as such, while the rest have a more casual look-and-feel. So the first question to the user is of the form: "Are you planning to use this smart

watch mostly: for sports activities, or as casual wear?". Depending on the answer we get:

  - **Wearables for sports activities**: In this case, we returned both of the products ("Watch Ambit 3","Watch V800") again for the reasons stated in the "Assumptions" section.
  - **Wearables with a casual look-and-feel**: In this category we have 3 items. Two of them are Apple products and one is not, so again our question is of the nature: "Are you a fan of Apple products?"
    * **Wearables with a casual look-and-feel for Apple fans**: We end up with 2 apple smart-watches of different size and subscription plan, so eventually we ask whether the user: "prefers a bigger watch, or a smaller one".
      · **Wearables with a casual look-and-feel for Apple fans that prefer bigger watches**: 1 entry, so return the result
      · **Wearables with a casual look-and-feel for Apple fans that prefer smaller watches**: 1 entry, so return the result
    * **Wearables with a casual look-and-feel for non-Apple fans**: Only 1 entry in this case, so return the result to the user.

- **Smart Home**: Last but not least, for the "Smart Home" category I was able to identify 3 subcategories, shown in the query posed to the client: "Are you interested in products for: (1) listening to some music, (2) making some tasty milk creations, or (3) cleaning your house?"

  - **Smart Home products for music**: Since the 2 items in our data are slightly different and have different subscription plans, I looked into them and came up with this appropriate query that sums up neatly their differences: "Would you prefer a: (1) stronger but bigger speaker, or (2) a less strong but more compact one?"
    * **Smart Home products for music that are bigger and stronger**: Only 1 entry in this case, so return the product to the client

        - **Smart Home products for music that are smaller and more compact**: Only 1 entry in this case, so return the product to the client
    - **Smart Home products for cooking**: Only 1 entry in this case, so return the product to the client
    - **Smart Home products for cleaning**: Return both of the items, for the reasons stated in the "Assumptions"

4 Implementation Details

In this section I shall give some important implementation details, regarding the flow of the program.

1. The contents of the input file are saved in a dictionary of dictionaries, as it was regarded of being the most appropriate data structure for representing this knowledge base.

2. I am using While loops to consecutively ask for inputs from the user at each step. If the input is matching with the choices available at each of the levels, then we proceed to the next layer, otherwise the user is prompted to provide a correct input.

3. In all of the steps, the program is parsing the input(natural language) of the user/client as strings. I am using regular expressions to compare these inputs with corresponding strings according to the loop that we are, and if the input is matched with part of them (thus covering the cases of "lazy" inputs from the user, e.g. use of "clean" instead of "cleaning") it is considered as a match (re.search()). Additionally, the input is not affected by capital or lower cases (re.IGNORECASE).

5 Execution examples

Here I will give some images with some execution examples, but please feel free to check for all possibilities

```
Welcome to the application! If at any point you would like to finish your session, please type exit and then hit enter!
Please select what type of product you would like to purchase:
*** Phones & Tablets ***
*** Gaming & Gadgets ***
*** Computing ***
*** Wearables ***
*** Smart Home ***
Type in one of the above categories!phones
Do you prefer Apple or Samsung products? (apple/samsung)apple
Do you prefer 32GBs or 128GBs of internal memory? (32/128)128G
Are you a fan of bigger or smaller screen sizes? (bigger/smaller)smaller
Estimation Complete! We are suggesting the following item(s):
iPhone 7 128GB, with a subscription plan of: 44.99€
Would you be interested in searching for another product? (Yes/No)No
Thank you for using our app! Hope to see you again in the future!
```

Figure 1: Apple phone with 128GBs of memory and smaller screen

```
Welcome to the application! If at any point you would like to finish your session, please type exit and then hit enter!
Please select what type of product you would like to purchase:
*** Phones & Tablets ***
*** Gaming & Gadgets ***
*** Computing ***
*** Wearables ***
*** Smart Home ***
Type in one of the above categories!wear
Are you planning to use this smart watch mostly for sports activities, or as casual wear ?casual
Are you a fan of Apple products? (yes/no)yes
Would you prefer a bigger watch, or a smaller one? (bigger/smaller)smaller
Estimation Complete! We are suggesting the following item(s):
Watch 38mm, with a subscription plan of: 39.99€
Would you be interested in searching for another product? (Yes/No)yes
Please select what type of product you would like to purchase:
*** Phones & Tablets ***
*** Gaming & Gadgets ***
```

Figure 2: Casual, smaller Apple smart-watch

```
Welcome to the application! If at any point you would like to finish your session, please type exit and then hit enter!
Please select what type of product you would like to purchase:
*** Phones & Tablets ***
*** Gaming & Gadgets ***
*** Computing ***
*** Wearables ***
*** Smart Home ***
Type in one of the above categories!Gaming & Gadgets
Are you interested in: VRs or Drones?Dron
Do you care for a more powerful, more robust and increased autonomy model, or could you settle for less? (Yes/No)yes
Estimation Complete! We are suggesting the following item(s):
Drone BEBOP 2, with a subscription plan of: 59.99€
Would you be interested in searching for another product? (Yes/No)
```

Figure 3: The stronger of the two drones