

Travel Buddy - A Project Proposal

Paul Freeman, Inga Jatzkowski and Diana Hooper

April 22, 2015

1 Introduction

The system we are developing is an interactive agent capable of engaging users in a dialog about travel. The agent will serve as a travel “buddy”, by expressing an interest in the travel experiences of the user, and by talking about the travel experiences of others. While the primary goal of the agent is to be engaging to each user, it attempts to achieve the primary goal by informing the users of experiences they might enjoy, based on the agent’s beliefs about the users. The purpose is only to provide inspiration for destinations, much like one would often do when conversing about travel. By contrast a travel recommender system would simply just serve up a recommendation and that would be it. Our conversation would only halt when the user no longer desires to converse with the agent, again much like how real conversations would end.

2 Motivation

Many existing interactive agents are essentially elaborate user interfaces for case-based reasoning backends or other data retrieval processes. Additionally, the notion of a basic companion agent has been extensively researched. Our motivation is to provide a hybrid agent which provides enjoyable interaction for users while also providing indirect access to a knowledge base of data on a specific topic. Because we are not building a travel agent or a complete travel recommender, it allows us for more freedom as to what we would like to focus on with our agent. By initially creating an agent which can retain simple facts about the user and then relate them to trips they might find interesting, coupled with prying information out of the user through simple dialogue, we feel we could have something that could be a nice proof of concept. Further work could be to add more nuance to the conversation, such as talking about what the user could bring to certain locations or maybe making suggestions based on their possessions.

There will not be a solution in the form of simply recommending the perfect trip to the user. Our goal is rather to make a simple emulation of how a conversation about travel would unfold. This agent is able to use its a complex user model and extensive knowledge base to engage a user in discovering their own solution. We feel that this approach increases the willingness to interact with the agent, thus giving more opportunities to solve the problem of getting inspiration for new holiday destinations.

3 Related Work

A joint project between researchers in the U.K. and Finland developed an Embodied Conversational Agent (ECA) to assist users in planning their day and reminding them to perform healthy activities. The system works by learning about the users daily activity by engaging them in everyday conversation. Rather than requesting specific knowledge from the user, the conversational approach is more natural for the user, although it relies on the agent making more inferences on its own. The beliefs of the system are then used

to develop a healthy plan for each day and the plan is used to make suggestions to the user.

For example, the agent may learn that the user often has free time before work and suggest to the user that they go to the gym before work. At the end of the day, the agent will then follow up and inquire as to whether the user went to the gym or not. Through this approach, the agent attempts to be less intrusive in the users life, while still providing a relevant service in a friendly way.

In an earlier work conducted by a research group in Italy an Intelligent Travel Recommender (ITR) was developed which used case-based reasoning to provide the user with suitable suggestions concerning destination, accommodation and activities for an upcoming trip. The system uses a very simple interface for gathering the user's preferences and attempts to match their input to its trip databases. In case of failure, i.e. no matched could be found, it suggests a set of relaxations to the user's selected preferences. Additional to this the system exploits case similarity with older sessions to rank the results of the user query by computing the similarities of the items of the current query to those obtained in similar previous sessions.

The ITR is not a travel agent and doesn't try to come up with a conclusive travel plan, it rather suggests suitable locations, activities, etc. and lets the user collect the various trip components in their "travel bag" for later review much like a shopping cart at an online shop.

4 Analysis

Our analysis will be broken into several steps. At the base level, our discovery process will include interviews with "experienced travellers". These people will be used to gather information for our knowledge base, as well as provide us with an idea of topics of discussion during a conversation about travel. Other travel information will be gathered, as needed, from travel blogs and forums.

Beyond interviews, the analysis process will include a variety of diagrams showing the interaction of ideas within our knowledge base. Process diagrams will be used to layout the order of events the agent moves through before presenting a response to the user. Influence diagrams will document information in the knowledge base that could potentially change beliefs in other areas. For instance, if the agent adds a belief that a user dislikes hot weather, this will influence beliefs about their enjoyment of locations with warm climates.

5 Design

The system architecture will be made up out of two main components. The first part is the knowledge base which contains all the information about the previous users of the program and all the trips our "friend" knows about. It also contains the user model that the program keeps of the current user, which contains all the beliefs about the user's travel preferences. And also things that are still unknown and need to be known for the

“friend” to make a suitable recommendation. The other part is the interpretation and mapping of the user input and the generation of useful agent output, both in the form of text.

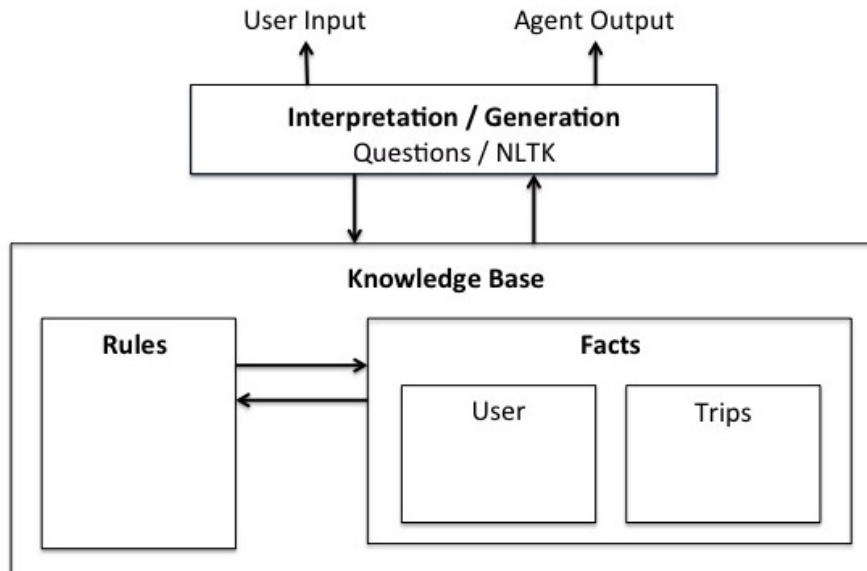


Figure 1: Proposed design for the overall architecture.

The agent itself will be written and designed using the knowledge engine, PyKE, which is a Python module for logic programming based on the popular PROLOG language. PyKE provides us with three kinds of knowledge bases, the Knowledge Fact Base, the Knowledge Rule Base and the Knowledge Question Base.

5.1 Components

The Knowledge Fact Base will contain all the facts we know about the user, such as; name, previous trips, travel possessions, the trips themselves (with associated locations, weather conditions, interesting sites, attractions, and activities).

The Rule Base will be used to encode how the program will make connections between the user’s preferences stored as beliefs and the trips. It will also try to match the user to other users in its knowledge base to find trips that similar users have taken and enjoyed and recommend them to the current user.

The Question Base can be used to store questions to ask the user to enquire about their preferences. The structure of this seems a little inflexible though upon first impression since we aim for a more natural feeling dialog than pre-coded questions so the usefulness of this tool for our system is still up for debate.

A different option for realising a natural language user dialog is employing the Natural Language Toolkit (NLTK) for python. NLTK provides different modules for natural

language processing such as lexicons, parsers or text classification that can be used to map the user input to a number of discourse acts in order to extract the relevant information and update the belief system.

5.2 Operations

In order to build a working system, we will need different operations in order to deal with both interpreting the user input, retaining the informations gained from the interpretation and the mapping and then providing some kind of response. For our system we imagine there would be two different kinds of responses, questions and suggesting. Questions have the purpose of directly asking the user in order to clarify things for the user model. Suggestions can be served up as direct suggestions or be formulated as fabricated previous experiences for the agent to which the user can respond positively or negatively, thereby also providing insight to the user's preferences. We want to present these responses mostly at random in order to avoid making the conversation follow a set pattern.

Our simple interpretation system would work by receiving some user input. Since we are not implementing an extensive natural language processing system, the next step would be to extract known keywords from the input, such as destination names and simple adjectives. Next, we will attempt to extract coherent facts from the keywords and save them to our knowledge base. So if the user inputs "I went to Tahiti. I love Tahiti!" we save the fact that the user went to Tahiti in our knowledge base and another fact about how they see that destination in a very positive light. If there is some difficulty extracting facts from the keywords, perhaps because of conflicts, we might try to clarify by asking the user directly by presenting what the agent deems the most appropriate interpretation and getting their response. If the user does not accept the interpretation, the agent could try presenting another interpretation or just give a confused response. The user might also add general information about destinations or their personalities using these adjectives. By saying "I'm very adventurous.", the agent would update its facts by now adding the fact that they are adventurous. Or they could describe a destination as being relaxing or educating, which would in turn add more detail to the descriptions we have of certain trips.

All of these gathered facts are then sometimes used for making suggestions or asking questions. We could for example use the fact that they went to Tahiti and liked it there to look up destinations which share the same characteristics. For this we have a collection of facts about the different destinations. So Tahiti might have a fact categorizing it as being a tropical destination. Doing simple matching, we might just end up pulling another destination which has this fact and suggesting it. During the matching process we will also try to filter out destinations which have characteristics which are seen negatively by the user. We can either completely rule them out or perhaps base our suggestions around a scoring system based on the features, such that some destinations are allowed despite their shortcomings.

Generally the knowledge base is updated whenever the agent has an interpretation it is sure of which adds more information. Knowledge is retrieved when making suggestions or asking the user about their experiences or preferences.

6 References

- (1) Marc Cavazza, Cameron Smith, Daniel Charlton, Li Zhang, Markku Turunen, and Jaakko Hakulinen. 2008. A 'companion' ECA with planning and activity modelling. In Proceedings of the 7th international joint conference on Autonomous agents and multiagent systems - Volume 3(AAMAS '08), Vol. 3. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1281-1284.
- (2) Ricci, Francesco, et al. "ITR: a case-based travel advisory system." Advances in Case-Based Reasoning. Springer Berlin Heidelberg, 2002. 613-627.
- (3) Loper, Edward, and Steven Bird. "NLTK: The natural language toolkit." Proceedings of the ACL-02 Workshop on Effective tools and methodologies for teaching natural language processing and computational linguistics-Volume 1. Association for Computational Linguistics, 2002.