University of Osnabrück

Winter Semester: 2017/18

Final Project: Speech dialogue systems and embodied conversational agents
Lectures: Prof. Dr. Kirsten Bergmann

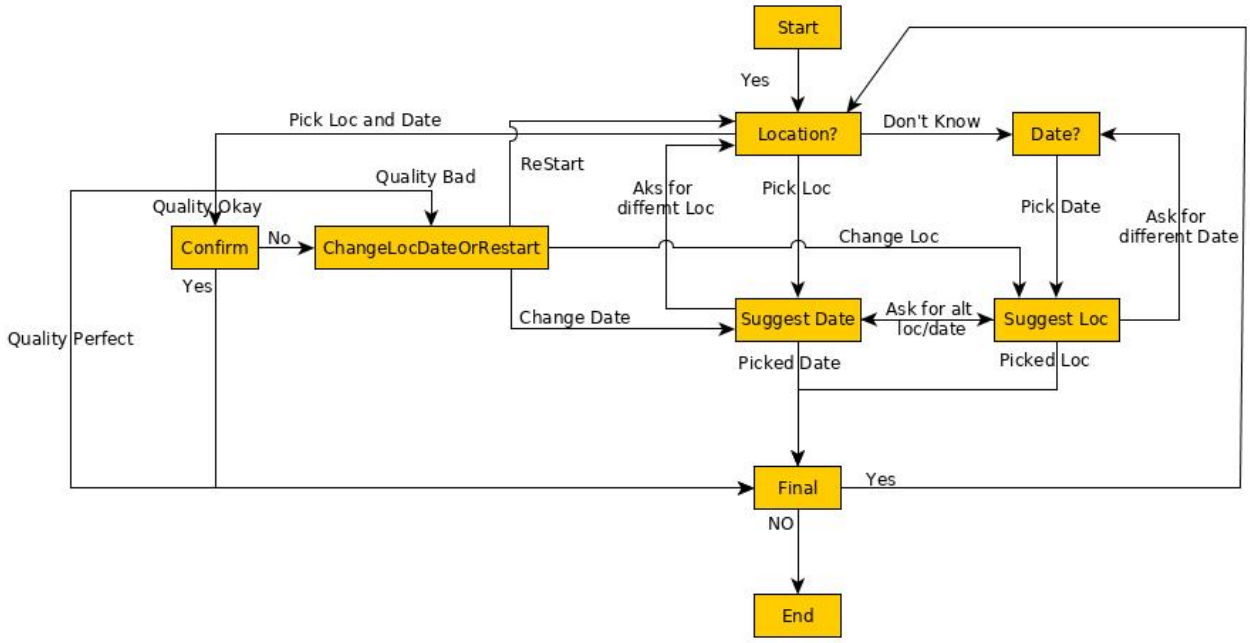
Skiing Info

Group Members:
1. Shahd Safarani
2. Roman Treutlein
3. Manish Kumar

1. Domain/Functionality

We have developed a dialogue assistant to help ski enthusiast to suggest near by ski resorts and also provide its condition with respect to snow quality and available dates.

The system can handle 3-4 dialogue turns and can continue as per the inputs. Please see the flowchart which explains the working of out dialogue assistant.



2. Motivation

As part of the **Speech dialogue systems and embodied conversational agents** course and seminar, we learned about many dialogue agents and frameworks. So as part of our final project and interest we thought about using this acquired knowledge to build an interactive chatbot related to Skiing using one of these frameworks. Due to the simplicity and compatibility, we decided to use IrisTK as our dialogue framework.

One of our teammates is a regular skier and had this idea about building an interactive system for skiing enthusiast. The system will be able to give information about snow condition for few specified ski resorts for current week. It also has the functionality to suggest dates and location which is good for skiing.

3. Front-end and back-end

Our system runs as a single stand-alone thick client on a window based system either a laptop or system. We have created ski resort database in form of a xml file which is used by the system to process the input data and provide the required details as per data flow.

Front end: IrisTK has its own application interface for dialogue processing and output generation and this is used as a front end for our system.

Back-end: Back end for IrisTK is java based system which helps to design the flow and data processing.

Data: As of now, we are using the hard coded data for resort and its condition. But it can be extended to use any database supported by IrisTK.

4. Framework

We have used IrisTK as the dialogue system. The choice is mainly motivated by its compatibility and ease of use. Also, availability of tutorials have been added advantage for the new users.

5. How to use the system?

1. Install IrisTK & eclipse.
2. Clone the git repository (https://github.com/rTreutlein/guess) into the apps folder of IrisTK
3. Run the java application.

6. Evaluation

We have manually evaluated the dialogue system by checking all the defined dialogue turns. Also, by randomly providing the inputs which is not expected to check system behavior for exceptions.

7. Conclusion

The system implementation is simple and it works as per the requirement. The dialogue turns can be handled with great ease with the help of IrisTK framework.

The basic structure and implementation well as per out requirement but we can enhance it from database perspective. As of now, the data is hard coded, instead, use of database would be preferable for real time dialogue assistance.