



# Spoken Dialogue System for Restaurant Recommendation and Reservation

Rafael E. Banchs, Seokhwan Kim

Institute for Infocomm Research, Singapore

{rembanchs, kims}@i2r.a-star.edu.sg

## Abstract

This Show & Tell demo paper describes a multi-strategy spoken dialogue system for restaurant recommendation and reservation in Singapore. The system uses a three phased dialogue strategy for recommendation, selection and booking. Given its simple architecture, it can be easily adapted to deploy restaurant recommendation services for any city, as far as the required regional data is available.

**Index Terms:** spoken dialogue, booking system.

## 1. Introduction

Nowadays, spoken dialogue applications (such as Siri, Google voice, etc.) are starting to gain popularity given the availability of cloud-based services accessible from mobile platforms. Although general domain spoken dialogue is still far from being usable in commercial applications, restricted domain dialogue engines can achieve high accuracy for specific task-oriented scenarios, in which the number of variables and the vocabulary size are limited. An example of this type of dialogue scenarios is the case of booking systems [1], [2], [3].

In this paper we describe a multi-strategy spoken dialogue engine for restaurant recommendation and reservation that has been implemented for the city of Singapore. The system is based on a three phased dialogue strategy in which the system first recommends few options based on the user’s preferences, then it provides additional information for the user to select one venue, and finally it completes the booking process. In this way, the system actually integrates three specific dialogue engines into a single recommendation and reservation system.

The remainder of the paper is structured as follows. First, in section 2, the data structure and data collection process is described. Then, in section 3, the overall system architecture is presented and the operation of the system is described in detail. Then, in section 4, an actual example of user-system interaction is presented. Finally, in section 5, we present our main conclusions and plans for future work.

## 2. Data Structure and Data Collection

The data structure of the restaurant recommendation and reservation system is closely related to the algorithmic implementation of the system itself (to be described in section 3). According to this, three different variable types are defined: select-variables, info-variables and booking-variables.

Select-variables are those used by the system to filter options based on user explicit preferences. These variables are indeed lists of values for feature descriptors such as: food-type, price-range, area-of-city, and opening-hours. For each specific venue, each variable can contain more than one value. For instance, one restaurant can offer different types of food (Indian, vegetarian, western, etc.), as well as serve food at different opening hours (lunch and dinner). Another special select-variable is the restaurant-name, which can be explicitly stated by the user to select one specific restaurant of interest.

Info-variables, on the other hand, contain information that the system can eventually provide to the user in case she requests for it. These variables consist of strings which are specific for each of the restaurants in the dataset. This type of variable includes information such as: address, phone-number, url, email-address, general-description, positive-reviews, etc.

The last type of system variables, booking-variables, is similar to select-variables in the pragmatic sense that the system needs to ask the user to specify her preference for them. However, different from select-variables, these variables are not used in the recommendation phase as they are more related to the booking process. These variables are: restaurant-name, number-of-guests, booking-date and sitting-time.

The data collection was conducted by using an automated crawling system for collecting both select- and info-variables from different web sources. These web sources included websites such as: Streetdirectory (www.streetdirectory.com), STDirectory (directory.stclassifieds.sg) and HungryGoWhere (www.hungrygowhere.com). For the automatic collection of the data, the MICRA crawling system was used [4]. Some manual post-edition was also conducted for cleaning and correcting some data problems resulting from website structure inconsistencies and crawling errors.

## 3. System Architecture and Operation

Figure 1 presents a schematic diagram of the overall system architecture. As seen from the figure, the recommendation and reservation task is divided into three subtasks: recommendation, selection and booking, where each individual sub-system mainly interacts with one of the variable types.

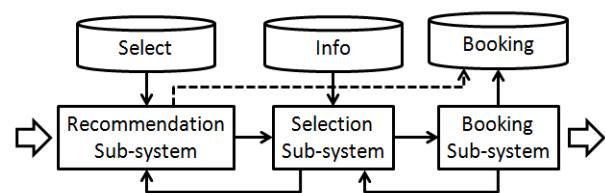


Figure 1: Schematic diagram of the spoken dialogue system

Also, as seen from the figure, the system allows for the user to transit back from the booking to the selection sub-system, and from this last one to the recommendation one. This enables the system for dialogue recovery, in case an error occurs, and for handling user changes of intention.

The basic operation of each spoken dialogue sub-system can be described as follows:

**Recommendation sub-system.** It is responsible for collecting the user preferences. It will keep asking questions about select-variable values (food-type, price-range, area-of-city, etc.) until the user provided answers allow for reducing the list of candidate restaurants to four or less. If during this phase the user also provides information regarding booking-variables, such variables are also updated (see dotted line in figure 1).

**Selection sub-system.** It is responsible for helping the user to decide on one of the recommended options. In this phase the user can ask the system about specific information on the provided options. This sub-system uses the info-variables to address the user requests. It also prompts the user for either selecting a particular venue or starting a new recommendation session, transiting either forward to the booking sub-system or backward to the recommendation sub-system, accordingly.

**Booking sub-system.** It is responsible for completing the booking, in case the user actually wants to make a reservation. During this phase, booking-variable values are collected (or only those remaining empty slots, in case the user already specified some of these values during the recommendation phase). Once all slots are filled, the reservation is completed and the user receives a “booking confirmation”.

For the implementation of the three sub-systems, a multi-strategy architecture, which is depicted in Figure 2, has been adopted. This architecture is intended to improve the robustness of the overall system against the errors that might occur in each sub-component, which are typically accumulated along the integrated sequence of system components. In this architecture, every level has a set of multiple sub-components, each of which is based on a different approach. The results obtained in parallel from the different sub-components at the same level might differ or even be in conflict with each other. However, in our architecture, none of these results are filtered out. Instead, all of them are passed to the next level sub-components which can take advantage of a multitude of inputs. In this way, the damages caused by the erroneous input from another sub-component can be diminished.

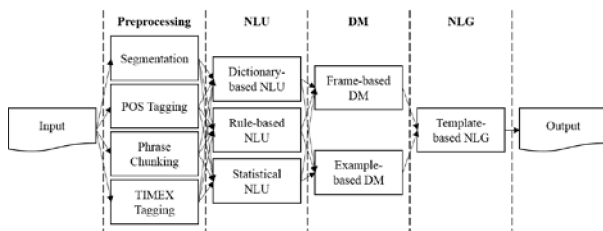


Figure 2. Multi-strategy dialogue system architecture

Every sub-system consists of a sequence of four process levels: preprocessing, natural language understanding (NLU), dialogue management (DM), and natural language generation (NLG). In the preprocessing level, some linguistic analyses including word segmentation, part-of-speech tagging, and base phrase chunking are performed by using the Stanford CoreNLP toolkit<sup>1</sup>, while temporal expressions are recognized by a rule-based TimEx tagger. These preprocessed results are used by three different NLU’s level sub-components: a dictionary-based, a rule-based and a statistical method. Then, the system actions can be also obtained from multiple DM strategies incorporating frame-based and example-based methods. Finally, all these intermediate results are used for generating a system response with a template-based NLG strategy.

The integration of the multi-strategy dialogue system with the automatic speech recognition and text-to-speech engines was carried out by using the APOLLO integration platform [5]. The output interface includes a web-browser frame, in which different visual information is displayed.

<sup>1</sup> <http://nlp.stanford.edu/software/corenlp.shtml>

## 4. User-system Interaction Example

In this section, an actual user-system interaction example is presented. For each turn, the user input (U) is provided along with the system response (S) and the output display, if any.

U: can you recommend an italian restaurant

S: Of course, there is a lot of good italian food in Singapore! What range of prices you have in mind: cheap, intermediate or expensive?

U: very expensive please

S: I have found 2 restaurants for you. Do you want more information about any of these or do you want to make a reservation? (Displays restaurant list: (1) La Cantina in Venezia, (2) Alkaff Mansion)

U: give me more information about la cantina

S: La Cantina in Venezia is located in Netheravon Road and serves Italian cuisine (Displays La Cantina’s website)

U: what about the second one

S: In this case it would be Alkaff Mansion gets a new lease of life as the building receives a face lift from its new owners LHN Group. The premises is converted into a semi-fine dining Italian restaurant with an integrated cafe and cocktail bar. (Displays Alkaff’s website)

U: okay good

S: Do you want to make a reservation at Alkaff Mansion Ristorante?

U: yes please

S: Ok, I will do a reservation for you! And, you will be interested in dinner or lunch?

U: I am considering dinner

S: Ok, good, you want it for dinner. What day do you want this reservation for?

U: tomorrow

S: Fine, tomorrow seems to be a nice day for good food! And, how many persons is this reservation for?

U: we will be six persons

S: Your reservation has been made! (Displays reservation details)

## 5. Conclusions and Future Work

We have presented a multi-strategy spoken dialogue system for restaurant recommendation and reservation services. The system uses a three phased dialogue strategy for recommendation, selection and booking. As future work, we plan to develop a mobile app to make the system accessible from any location, and to exploit GPS-tracking capabilities of mobile platforms to make the system location-aware. We also plan to conduct a pilot test by connecting to an actual booking system.

## 6. Acknowledgements

The authors want to thank I<sup>2</sup>R for its permission to publish this work, and Dr. R. Jiang for his assistance with APOLLO.

## 7. References

- [1] E. Levin et al (2000) The AT&T-DARPA communicator mixed-initiative spoken dialogue system, in Proc. Int’l Conf. Spoken Language Processing.
- [2] L. Lamel et al (1999) The LIMSI ARISE system for train travel information, in Proc. ICASSP, pp. 501–504
- [3] K. Komatani et al (2005) User modeling in spoken dialogue systems to generate flexible guidance, User Modeling and User-Adapted Interaction, 15(1), pp. 169–183
- [4] R. E. Banchs (2013) MICRA: a Mini Crawler for the automatic collection of web data, Internal Report, I<sup>2</sup>R, Singapore.
- [5] R. Jiang et al (2012) Component pluggable dialogue framework and its application to social robots, in Proc. IWSDS