Conversational Agent and Management Tools for Conference and Tourism Domain

Luis Fernando D'Haro, Seokhwan Kim, Rafael E. Banchs

Institute for Infocomm Research - A*STAR
1 Fusionopolis Way, Connexis South
Singapore, 138632
{luisdhe, kims, rembanchs}@i2r.a-star.edu.sg

Abstract

In this paper we describe a platform and a set of useful tools that allow the fast creation of conversation agents for two different domain applications: a conference information system and local tourist guide. The paper provides detailed descriptions of the implemented tools for developer management and automatic extraction of conference-dependent information. Finally, we provide some usage statistics of the agent that was included in the mobile APP used last year at Interspeech conference.

Index Terms: conversational agent, management tools, conference mobile app.

1. Introduction

Nowadays, there is an increasing interest in using conversational agents especially on mobile devices since they allow users to quickly find information on the Web, to access to different command and control functions on the device, but also to have some fun with the agent using its chat capabilities. In the literature, we can find several examples of conversational agents used for different domains such as weather forecast [1], tourism [2], etc. For mobiles, the most popular applications are Apple´s Siri [3], Google Now, or Cortana. These multi-domain and multi-tasks agents are able, among many others, to save appointments, send text messages, providing weather or transportation information, or make phone calls. For Interspeech 2014, we deployed what is, up to the best of our knowledge, the first conversational Q&A agent specifically designed for handling simultaneously conference and local tourist information for the place where the conference is held [4]. However, the creation of this agent also involved the development of different tools for management of the queries, logs, as well as handling of fuzzy search, creation of abbreviation dictionaries, or automatic extraction of synonyms for allowing approximated search of titles and conference sessions. The paper is organized to describe the architecture in section 2; then, in section 3, the management tools, and in section 4, some usage statistics collected when the full system was used during the conference.

2. Architecture

The system architecture has three main components: 1) the client system implemented in the mobile application, 2) a web socket server which runs the service and internally communicates with the orchestration and searching modules, and 3) the different resources (e.g. databases, dictionaries, and models) and management tools used to provide the information to the users, and control the system.

The process to search for an answer is that the user poses a query using the graphical interface available on the mobile. Then, this is send as a JSON message consisting of: the query, the domain (conference or tourism), and GPS coordinates, if the user allowed for sharing them. Then, the orchestration is done in several steps: 1) A search in the index for generic questions or greetings like: '*what can you do?*', '*What is your name?*', or '*how can I start using you?*', 2) In case there is not an answer, we start a search on the conference or the tourist search engine depending on the user's selection through the mobile app. In case of the conference domain, the system allows to search by authors, affiliations, countries, titles, conference sessions and events, as well as general queries about conference facilities. Here, the system recognizes conference domain entities from the input by means of a fuzzy search algorithm; the algorithm is robust to a certain degree of input misspellings (e.g. authors' names). Then, the extracted entities are expanded with knowledge bases, including the information of keywords, abbreviations and synonyms (extracted automatically as explained in section 3.1). For example, the system could capture the concept '*Microsoft*' as an affiliation and '*speaker recognition*' and *'speaker identification'* as technical terms from the user input '*show me papers from MS on speaker identification'*. Then, a SQL query is generated from this semantic representation for searching papers, authors, or sessions in the conference database.

For tourist information, the system is able to search for maps, restaurants, sightseeing locations, as well as for local and general information (history, currency, exchange rates, laws, etc). Here, the module uses a search topic classifier. Depending on this classification, the system analyzes the query in order to extract the information. Besides, the system uses GPS coordinates to provide geo-contextual information for shopping malls, transportation or restaurants.

Then, if there is still not an answer, the system attempts a new search in the index for generic queries, but using a more relaxed threshold. In the last instance, the system tries to search on internet. Finally, the system sends back the answer to the mobile app by means of a JSON message which indicates: a) the best agent to show the information the question, b) the information the avatar will provide, c) type of information to display, e.g. a map, an external website, HTML content, a list of papers/authors/sessions, or simply a chat answer, d) the URL to be displayed: this way it is possible to show maps, websites, restaurants information, or pictures from sightseeing points.

3. Management Tools

On the other hand, the development of the agent not only required the creation of a Q&A index but also several tools for detecting scientific abbreviations, paraphrasing, management of logs, edition of questions and answers, and fuzzy search.

3.1. Automatic detection of abbreviations

In order to allow users to search for specific abbreviations or their expansions, the system, at runtime, should allow the use of either the short or the long form, while it expands the corresponding query to include all the variations. To make easy the creation of a dictionary of abbreviations and expansions, we deployed an offline module based on the algorithms described in [5][6]. This module, when used for Interspeech 2014 detected 420 abbreviations extracted only from the abstracts.

3.2. Paraphrasing

In order to allow users to use keyword searches and to include possible synonyms for those, we created an automatic paraphrase engine which given a list of canonical sentences generates new ones based on searching for semantically words. In the literature, we can find several algorithms for creating paraphrases combining different levels of abstractions and techniques. Our algorithm first extract noun phrases from chunks, then for each word in the chunk it looks for hyponyms and hypernyms in WordNet 3.0, semantically and contextually related terms and phrases using Word2Vec[1] trained on Interspeech papers from the last 5 conferences, and additional syntactic words from a Thesaurus dictionary. Then the candidate synonyms are filtered out based on the ratio between the perplexities of the original word and the candidate calculated using a language model trained on papers from previous years. This module extracted from the titles, sessions names and abstracts of the Interspeech 2014 papers a filtered list of 1607 keywords and 2128 expansions. E.g.: for the term speaker recognition the expanded options are: speaker classification, speaker detection, cluster recognition, dialect recognition, gender recognition, speaker recognizer, speaker identification.

3.3. Log Management

Another of the important tools we deployed for this system was a log manager which could allow us to perform, among others, the following tasks: a) Get a list of the most common or recent queries made to the system, b) Get a list of all or recent queries the system was not been able to answer, c) Allow the edition of the question and answers index (i.e. addition, edition or deletion) with the possibility of including time restrictions (i.e. allowing the system to give a specific answer given a time slot while given another answer at a different time), d) Send of broadcasting messages to all or specific users using the mobile app in order to share news with them or to give them answer that previously were not included in the system, and e) Get usage statistics for the different people that downloaded the app. Thanks to this tool, during the conference it was possible to promote a game among the attenders in order to reward the users with asked the highest number of different questions or the one who could find out the agent's age. On the other hand, these tools allowed us to update the Q&A index a total number of 13 times with more than 400 new queries, which allowed the system to retrieve now a correct answer for 159 similar queries.

[1] http://code.google.com/p/word2vec/

3.4. Fuzzy Search

One particular problem when searching for the authors or affiliations in a paper is the high chance for misspellings on the user's query. To reduce this problem, we implemented an algorithm based on estimating a combined distance between a vector space model, the edition distance, and phonetic similarity between the query and the system vocabulary.

4. Usage Statistics

The system was used at Interspeech 2014 receiving a total number of 2360 queries (1628 unique) from 215 different users (with an average number of queries of 10.98 and standard deviation of 22.13). The final Q&A index included 2365 different queries, information about 633 papers, 93 sessions, 1695 different authors, and 668 affiliations. For tourism, we included info about 35 touristic attractions, 81 different kinds of food, 120 sectors to search for restaurants along Singapore, and 39 typical dishes.

5. Conclusions and Future Works

We have described a multi-domain and multi-task conversational agent used in the context of a big scientific conference. Thanks to the management tools and modular architecture the system was able to handle 222 queries about tourism, 480 about the conference and 1598 chat interactions.

As current work, we are working on several new tools and graphical user interfaces that help developers to quickly deploy a similar system for different conferences. Besides, we are working on new improvements to the paraphrasing module to avoid non-grammatical sentences and to apply better filters.

6. Acknowledgements

To Jochen Walter Ehnes, Nicholas de Laczkovich and Tilo Westermann for integrating the agent on the mobile app.

7. References

[1] Maragoudakis M, "MeteoBayes: Effective plan recognition in a weather dialogue system", IEEE Intelligent Systems, 22(1), p. 66-77, 2007.

[2] Niculescu A I, Jiang R, Kim S, Yeo K-H, D'Haro L F, Niswar A, Banchs R. "SARA: Singapore's Automated Responsive Assistant, a multimodal dialogue system for touristic information". *Proc. MobiWIS, 2014*. pp. 153-164.

[3] Bellegarda J. "Spoken Language Understanding for Natural Interaction: The Siri Experience", Natural Interaction with Robots, Knowbots and Smartphones: Putting Spoken Dialog Systems into Practice, p 3-14. 2014.

[4] D'Haro, L. F., Kim, S., Yeo, K-H., Jiang, R., Niculescu, A. I., Banchs, R. E., Li, H. "CLARA: a multifunctional virtual agent for conference support and touristic information", IWSDS 2015.

[5] Schwartz, A. S., Hearst, M.A, "A simple algorithm for identifying abbreviation definitions in biomedical text." (2003): 451-62. In Pacific Symposium on Biocomputing. World Scientific Press, Singapore.

[6] Y. Park, and R.J. Byrd, "Hybrid Text Mining for Finding Abbreviations and Their Definitions" Conference on Empirical Methods in Natural Language Processing, Pittsburgh, PA, June 2001: 126-133