Dialog Systems and their Inputs

Alexiei Dingli[1], Darren Scerri[1]

[1] Department of Intelligent Computer Systems, Computer Building, University of Malta, Msida, Malta

{alexiei.dingli, dsce0006}@um.edu.mt

Abstract. One of the main limitations in existent domain-independent conversational agents is that the general and linguistic knowledge of these agents is limited to what the agents' developers explicitly defined. Therefore, a system which analyses user input at a deeper level of abstraction which backs its knowledge with common sense information will essentially result in a system that is capable of providing more adequate responses which in turn result in a better overall user experience.

From this premise, a framework was proposed, and a working prototype was implemented upon this framework. These make use of various natural language processing tools, online and offline knowledge bases, and other information sources, to enable it to comprehend and construct relevant responses.

Keywords: Dialog Systems, External Knowledge Acquisition, RDF, Knowledge Bases, Chatterbots

1 Introduction and Background

Conversational agents are deployed in various forms and designed to cater for different domains and goals, ranging from automated hotel-booking agents, to personal assistants, companions, and agents designed simply for entertainment purposes. Moreover, one can categorize conversational agents into two main types based on how these process the input and generate their output. These two types can be realized as being *chatterbots* and *dialog systems* [1].

The main difference between these two types of conversational agents lies in what these systems are designed to model. Chatterbots model, or rather simulate a conversation in its basic sense, and intend to fool the user that he is communicating with an intelligent entity that does in fact understand what is being said. On the other hand, dialog systems attempt to model the actual dialog process which also incorporates the task of analyzing and understanding the input, which in turn aids in the generation of an adequate dynamic response.

2 Aims and Objectives

The aim of this project is to provide a proof-of-concept generic-conversation framework for conversational agents, and a working prototype that can be categorized as being a hybrid between a chatterbot and a dialog system.

"Building a system that could understand open-ended natural language utterances would require common sense reasoning, the huge open-ended mass of sensory-motor competencies, knowledge and reasoning skills which human beings make use of in their everyday dealings with the world" [2].

With regards to this hypothesis, the system makes use of modern natural language processing technologies and tools to analyze user dialog input while simultaneously using information that is obtainable from external sources to attempt to further understand the input and ultimately generate appropriate responses to the user. Another goal is to merge these various sources to create a single, local, knowledge base which enables the system to keep track of the world of the user, i.e. the relationship and interaction of the user with various entities. These sources include content and information on different entities and concepts, including common sense knowledge and knowledge about specific people and world entities.

The aims and objectives of this project can be summarized as follows:

1. To build an expandable proof-of-concept system that provides a syntactic, semantic and pragmatic understanding of input.
2. To simulate intelligence by providing adequate output and logical conclusions derived from dialogue input and local and external knowledge bases.

Moreover, the proposed system will be as customizable, flexible, and modular as possible, so that it would require minimum effort to upgrade and adapt the system to handle input of varying complexity and topics.

3 Design

The developed prototype consists mainly of three phases: Natural Language Understanding (NLU) phase, intermediary processing phase, and finally the output generation phase.

The NLU module is primarily makes use of ChatScript [3], an award-winning chatterbot engine. ChatScript employs various linguistic technologies to enable support for more flexible linguistic input in terms of syntax and semantics, and more expressive and semantics-oriented pattern matching rules, where matching patterns of meaning (semantics and pragmatics) is considered more important than matching patterns of words (syntax). These technologies and processes include the use of WordNet as a semantic network, Part-of-Speech tagging, pronoun resolution, conceptual relations, and preprocessing abilities.

ChatScript is used by the system mainly as a "normalization phase". This phase is essentially a mapping process that maps natural language input into a more formal

representation using XML. This is essential in order to allow the creation of rules that allow matching of a number of input utterances that are effectively semantically equivalent. This phase is analogous to how Façade maps text to discourse acts.

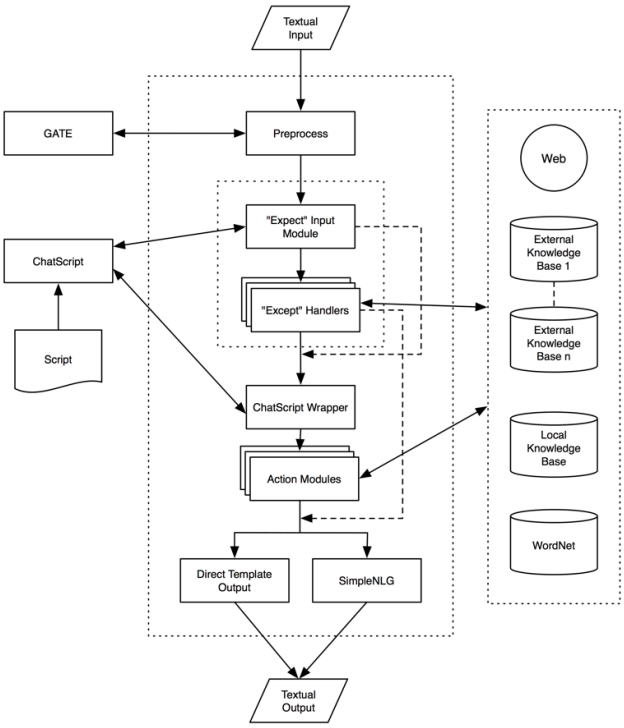

Fig. 1. System Block Diagram

Input is preprocessed using two approaches. Using GATE and ANNIE [4][5], the system attempts to resolve pronouns into their respective named entities. Moreover, ChatScript includes a preprocessing phase in itself, performing actions such as spell checking and term substitution.

The normalization phase allows the system to perform further intermediary processing on the input, such as querying local and external knowledge bases to allow to better understand and respond to the input. This is achieved by a number of processing modules that can be developed to handle specific types of input. The system's dialog manager employs a similar technique used in CONVERSE [6] with regards to these processing modules. The system incorporates a polling technique for which the input is passed through all implemented processing modules, or “Action Modules”, for which each module advertises a score indicating how much it is confident to process the input.

Each action module is able to perform various tasks, include querying and managing local and external knowledge bases through global wrappers, triggering other action modules, querying ChatScript for further normalization, adding and retrieving

topics from the dialog manager, generating output and alerting the dialog manager that the subsequent user utterance is expected to contain certain type of information (such as names, feelings, numbers, etc.).

The knowledge bases used in the system can be grouped into two types: the local RDF knowledge base, and other external knowledge bases (can be either offline or online), which are all accessible throughout the system.

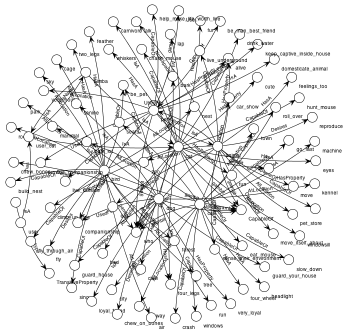

Fig. 2. Example RDF Graph Generated by the System

YAGO and ConceptNet are used by the system as external knowledge bases allowing the system to gather and make use of both entity specific and common sense knowledge. The retrieved data from these knowledge bases can be merged with the local RDF knowledge base to create an amalgamated network of knowledge, combining the "world of the user" with external knowledge.

4 Implementation and Evaluation

A prototype system was developed upon the described framework. This prototype included the development of a number of action modules, in conjunction with their respective ChatScript script which attempt to handle a specific set of user utterances, ranging from possession and sentiment statements, to questions regarding general and specific world entities.

The evaluation was carried out two-fold: hands-on conversation with the system (prototype), and a questionnaire filled by a number of users who engaged in a dialogue with the system, following a pre-defined context and a set of example tasks, tested this prototype and answered questions on a five-point Likert scale which were aimed to be conclusive in whether the system actually achieved its aims and objectives. All results were biased towards the higher end (mean values for answers: 1 - 2.05%, 2 - 6.77%, 3 - 25.84%, 4 - 49.73%, 5 - 15.54%), which imply that the overall result is positive.

The following example shows the system's ability to keep track of user-defined information and depicts a level of understandability:

- User: I have a red car.
- System: OK, noted!

- User: What is the color of my car?
- System: Red.
- User: I also have a cat named Simba.
- System: Glad to know that you have a cat named Simba.
- User: Do you know who is Simba?
- System: Simba is your cat.
- User: Do you know who is Barack Obama?
- System: Barack Hussein Obama II is the 44th and current President of the United States. Also, I have heard a lot of positive things about Barack Obama. What's your view?
- User: I think he is a great president .
- System: Why do you think that Barack Obama is great?
- User: Because he is an exemplary politician.
- System: Ok then, so you think that Barack Obama is great, because he is an exemplary politician.
- User: Are cats capable of chasing mice?
- System: Yes
- User: Can cats fly?
- System: No, I don't think so
- User: Is a cat an animal?
- System: Yes
- User: Is it a machine?
- System: No
- User: Do you know anything about Bill Gates?
- System: Bill Gates is born on date 1955-10-28.
- User: Anything else?
- System: Bill Gates is married to Melinda Gates.
- User: Tell me about cats
- System: Cats are capable of eating meats, killing birds, drinking milks, eating mice, sleeping, climbing up trees and chasing mice.

It can be concluded that the system is regarded as having potential to expand and improve to further achieve higher quality results and ultimately fulfill the aims and objectives in a greater sense. Naturally, there are still possible areas for improvement since the evaluation was done on a prototype with strict boundaries.

The system's ability to make use of external sources to support and enhance its knowledge of the real world can be considered as being a step forward towards the implementation of more natural and human-like conversational systems. The system is able to exploit the vast amounts of data found in structured knowledge databases that are consulted to both understand, and answer in a natural and informed manner.

References

1. Klüwer, T.: From Chatbots to Dialog Systems. International Journal, 1–22, (2011).
2. Mateas, M., Stern, A.: Natural Language Understanding in Façade: Surface-text Processing, (2004).
3. Wilcox, B.: Suzette, the Most Human Computer, (2011)
4. Bontcheva, K., Cunningham, H., Maynard, D., Tablan V., Saggion H.: Developing Reusable and Robust Language Processing Components for Information Systems Using Gate. 3rd International Workshop on Natural Language and Information Systems, 223–227, (2002).
5. Cunningham, H., Humphreys, K., Gaizauska, R.: GATE - a Tipster-Based General Architecture for Text Engineering. Proceedings of the TIPSTER Text Program (Phase III) 6 Month Workshop, (1997).
6. Levy, D., Batacharia, B., Catizone, R., Krotov, A., Wilks, Y.: CONVERSE: a Conversational Companion. Proceedings of the 1st International Workshop on Human-Computer Conversation, (1997).