Towards A Context-Based Dialog Management Layer for Expert Systems

Victor Hung, Avelino Gonzalez, Ronald Demara
University of Central Florida
Intelligent Systems Laboratory
victor@isl.ucf.edu, gonzalez@ucf.edu, demara@mail.ucf.edu

Abstract

Knowledge engineering relies on a unified effort of two components for success: 1) a strong expert knowledge back-end, and 2) an effective user interface. We address the latter issue, identifying the need for intuitive human-computer interface to channel expert knowledge. Speech-based conversation agents describe those computer-based entities that interact with humans to help accomplish a certain task via spoken word input. This paper proposes a method of managing spoken dialog interactions in response to recognizing the human user's goals when accessing an expert system. It is important to note that a set of goals can co-exist during a single conversation, and that each goal may be presented in an asynchronous manner. Such a stipulation exists to enhance the naturalness of the interaction. Inspired by Context-Based Reasoning, the dialog system features a goal management system that ultimately controls the behavior of the expert system.

1. Introduction

Building an effective expert system involves both a robust knowledge base as well as a functional human-computer interaction (HCI) interface. Often times, the latter of these components is overlooked, where expert system designers opt for a simplistic text-based querying methodology. In this paper, we propose an expert system interface layer that relies on speech-based input. This work is inspired by popular media's notional machines that converse with humans through spoken natural language. Examples of these are Kubrick's HAL, Knight Rider's KITT, Star Trek's Commander Data and Lucas' C3PO, among others [1]. These have the characteristics of speaking human languages in a very *natural* manner. In reality, this is a very difficult problem to solve, not only from the standpoint of 'catching' the spoken words, but of making sense out of them and composing an intelligent response based on knowledge. With this in mind, the general problem addressed by this work is how to feasibly make such dialog in HCI more natural, while at the same time, computationally efficient.

Existing assistive dialog systems, such as automated telephone operators are meant to allow human users to complete a single task in a serial manner. This style of conversation comes off as unnatural, as natural conversations do not always follow a linear train of thought. Under this model of dialog, topics of discussion may be interrupted and then re-visited on a continuous basis [2]. This research attempts to capture the human ability to recognize unexpected topic changes and provide seamless transitions between these shifts. Thus, a sense of naturalness may result from a system's capability to support a multitude of user goals in asynchronous time.

In short, the research here describes the mechanisms necessary for building a dialog system layer that addresses the user's goals for accessing an expert system. The remainder of this paper gives a briefing of the current efforts in assistive dialog systems, followed by a description of the approach and evaluation process for the proposed context-driven dialog manager.

2. Background

The following presents a background survey of three major technologies related to the proposed context-based dialog manager: Natural Language Processing (NLP), dialog system design, and Context-Based Reasoning (CxBR).

2.1 Natural Language Processing

NLP refers to the branch of artificial intelligence in which a human agent interfaces a machine in his or her own native tongue, whether through text-based entry or through spoken word speech input. Wilks [3] identifies four major issues associated with NLP: linguistic systems, knowledge representation structures,

information corpora, and statistical and quantitative methods. Linguistic systems refer to those NLP approaches that interpret user input at the grammatical level. Knowledge representation describes the process of transferring information into symbolic forms. Information corpora refer to those massive data sources that provide extensive knowledge of individual words or phrases for the sake of semantic resolution, such as those found in early thesaurus-like databases [4] to full-fledged ontologies [5]. In statistical and quantitative NLP, mathematical correlations, such as those found in Machine Learning (ML) are derived between massive amounts of data points made up of language elements.

2.2 Dialog System Design

Dialog systems represent a specialized field of NLP whose purpose is to provide an HCI method that resembles a conversational exchange between two humans. Additionally, dialog systems also exist to serve a particular purpose, such as providing information to its users or aiding them with completing a certain task. The channel of communication between the user and the dialog system may be based on text, speech and even gestures, or a combination of the three.

The work here focuses on speech-based dialogs. Spoken dialog systems exist with a wide range of complexity, especially when dealing with the level of "openness" in which the conversation domain can support. Specifically, dialog systems range in different expected response complexities, from single-word utterances to full-on natural language sentences. Often times, these differences can mean the trade-off between real-time processing ability and performance robustness [6]. Clearly, dialog system architects are pitted with many other design choices to select from. Flycht-Eriksson and Jönsson [7] list the four major components of typical dialog systems as the Interpreter, the Generator, the Domain Knowledge Manager (DKM), and the Dialog Manager (DS). The Interpreter is essentially the speech recognition unit, and the Generator is the speech synthesis module. The DKM acts as the back-end database for a dialog system, supplying it with various facts and figures. The DM serves as the conversation control mechanism, whose primary operation is to come up with a coherent response given a user utterance.

Conversation agents, or chatbots, refer to those software programs designed to interact with a human user in a natural language manner (using typed or verbalized inputs). Weizenbaum's psychotherapist-based ELIZA [11] represents the pioneering chatting agent software. After many decades, much improvement has been made over ELIZA to augment the realism and complexity of chatting agents. Sansonnet et al [12] propose a conversation agent architecture that emphasizes the requirement for domain-independent operation to maintain genericity. Galvão, et al [13] present Persona-AIML, which gives personality to agents based on the Artificial Intelligence Markup Language [14].

A specialized chatbot whose physical presence is incorporated into the HCI is known as an embodied conversation agent. The notion these interactive agents has existed with the inception of the computing age. Idealistic visions of these agents are rife with extraordinary capabilities, yet state-of-the-art technology yields only a sliver of these expectations. The 2000's presented an evolution of these embodied agents, beginning with the work of Cassell et al [15], whose conversational playmate, Sam, exhibited the idea that even a child could feel comfortable when interacting with a machine-based being. Bickmore and Picard [16] presented their studies with Laura, a personal trainer chatbot that showed that caring embodied agents proved more effective than ones of affective indifference.

The later half of the decade sought more ambitious goals in creating interactive agents. Lee et al [17] experimented with using robots as conversational agents, specifically an animatronic product spokesperson penguin named Mel. In this work, Lee et al supported the notion that humans could indeed interact with a physically engaging and conversationally interactive machine. Traum's [18] work extends this idea with Sergeant Blackwell, an agent whose conversational capabilities provide the user with a more natural human-computer interaction.

Similar issues affect each of the dialog systems agents presented above. For one, these projects do not employ a purely quantitative evaluation method. Many times, chatbots are judged using qualitative human arbitration. This style of software validation is often deemed unacceptable, especially in commercial engineering practices. Another problem that plagues chatbots is the narrow domain applicability of each individual system.

2.3 Context-Based Reasoning

The concepts involved in dialog system design can benefit from the tenets of CxBR. CxBR refers to a paradigm of agent behavior that reflects the idea that humans themselves operate in a contextually relevant manner, where only a fraction of their intelligence is needed for different situations [19]. The actual knowledge needed to function by both CxBR agents and humans is a function of the state of its internal and

external environments. The CxBR architecture consists of Contexts, Context-Transition Logic, Missions, and the Agent Interface. The state of an agent's environment, in both internal and external terms, makes up its behavioral Context. Context-Transition Logic manages the activation of contexts. Missions incorporate contexts and context-transition logic, resulting in the highest level of agent behavior. The agent interface exists as the link to the physical environment that encapsulates the CxBR agent.

Context-based methods have been used successfully in NLP. Specifically, the resolution of semantic ambiguity has been aided by contextualization techniques. The following presents related research featuring context-based disambiguation.

Sidner [23] insist that the design of embodied conversation agents requires a contextually-biased approach when dealing with natural language inputs. Such a feature, however, tends to eliminate the chance for a truly open dialog between humans and computers, since the agent's repertoire of understandable vocabulary severely pruned. Fügen et al [24] reduce speech recognition errors through context-based methods, as assisted by a knowledge-based dialog manager. Lieberman et al [25] use a common sense knowledge base, ConceptNet, to improve speech recognition using a context-based ML method. Sarma and Palmer [26] implemented a context-based prediction model to improve recognition and repair of speech inputs. Serridge [27] improved segment-based speech recognition using contexts from domain information. Young's [28] MINDS system incorporates context to make predictions and expectations on user's speech input.

The aforementioned context-based techniques all demonstrate how contextual cues can be exploited to enhance speech recognition. In the proposed dialog system, a conversation agent architecture is proposed that extends the use of CxBR beyond speech recognition, applying its principles in dialog and knowledge management. The resulting system can be used to permit spoken communication with an avatar. This avatar can serve as an interactive agent in a simulation or as the primary interface to an intelligent tutoring system.

3. Approach

This paper describes the components necessary to build a context-based dialog manager. The following section covers the main technical issues involved in the proposed context-driven dialog manager. These include: the goal management ideology, a framework for goal management, the context topology and a knowledge management model.

3.1 Goal Management

The major component of the proposed dialog system is the concept of *goal management*. Goal management refers to the processes that recognize and satisfy the interlocutor's needs as conveyed by his or her utterances. This section explains why goal management is important in a conversation and how it can enhance the naturalness of an HCI session.

Within any conversation, regardless of the presence of machine agents, there exists some sense of goal-oriented activity on the part of all its participants. Often, these activities are characterized as some form of knowledge transfer, such as requesting or delivering information. Every participant contributes utterances, or speech acts, to drive the conversation toward purposefulness. In a two-party conversation, both sides go into the conversation with the intention of getting something out of the interaction. The participants begin talking to one another in an initial state, only to end in a different state, a goal state. This model of conversation assumes that its conclusion occurs when both participants are satisfied with how much they have achieved from the session. Hence, under normal conditions, the goals of both speakers are accomplished when their conversation ends.

This same model may be applied to the interaction between an assistive conversation agent and a human user. The chatbot's primary goal as an assistive entity is to satisfy the user's needs. The human's goals, on the other hand, are simply the tasks that he or she wants to accomplish from talking to the conversation agent. Unbeknownst to the chatbot, the human's goals could be any number of things. The only way the machine can determine the user's intentions is to infer them from oral interactions. This is the essence of the goal management concept - the idea that an assistive dialog system understands a user's needs through the use of conversation management.

The aim of providing a goal management system is to offer a general approach to creating the effect of a natural, open dialog HCI experience. Open dialog refers to a loose set of conversational input constraints, allowing the agent to handle a wide range of user utterances. Additionally, one or more user goals can exist at any time during an open dialog interaction. This contrasts with the closed, highly-constrained and unnatural multiple choice-style of input expectation found in automated airline booking and credit card payment systems. Moreover, these types of interactions can only accommodate one user task at a time. An open dialog style allows for a more natural flow of

language. To realistically accomplish the illusion of open dialog through goal management, the following assumptions must exist: 1) the dialog system is limited to an expert domain, and the user is cognizant of the dialog system's functionality as an expert entity (this constrains the user to a topical context that the chatbot is deeply familiar, without jeopardizing the open dialog style), and 2) the user's goals are limited to those related to the chatbot's expertise (this assumption dictates that the user understands the agent's limitations as a domain-specific entity).

3.2 Framework for Goal Management

Goal management in a dialog system involves three parts: 1) goal recognition, 2) goal bookkeeping and 3) context topology. Goal recognition refers to the process of analyzing user input utterances to determine the proper conversational goal that is to be addressed. This is analogous to the context activation process in CxBR methods [19]. Goal bookkeeping deals with keeping track of the identified goals in an ordered manner. Bookkeeping simply services the recognized goals in the order they are received, using a stack. Context topology refers to the entire set of speech acts of the conversation agent. This structure also includes the transitional actions when moving between contexts when a goal shift is detected. The context topology carries out the responses needed to clear out the goal bookkeeping stack. The next sections further describe each of these goal management components.

Goal recognition is accomplished using linguistic analysis of each user utterance. This is similar to the inference engine found in CxBR systems [19], where conditioned predicate logic rules determine the active context according to the environmental state. The difference with the goal recognizer, however, is that the context is resolved using keywords and phrases that are extracted from a parts-of-speech parsing of input responses. With the aid of the knowledge base described previously, the user utterance is interpreted, and the context associated with this understanding is activated.

Goal bookkeeping describes the process of servicing every identified goal in the order that it is presented. Immediately after recognizing a goal, it is laced into the goal stack. The goal stack is modeled after Branting et al's [2] Discourse Goal Stack Model (DGSM), which only supported minor detours from the conversation path. In the proposed work, more complex interruptions may occur, such as switching to entirely different contexts. Thus, Branting et al's DGSM must be modified to be able to handle conversation paths that experience entire paradigm shifts between context changes.

3.3 Context Topology

In this work, goal management is achieved using a context-based approach. For an oral conversation, these circumstances refer to the state of both the internal state of the conversation agent and the state of the human user. For every context, there is an associated goal and a group of relevant actions that are executed to achieve this goal. A goal is defined as an end state that an agent desires to reach.

It is imperative that a dialog system be able to properly manage conversation goals, as the user can have multiple goals and he or she may introduce new goals at any time. Henceforth, the system must be able to service many goals at one time, as well as be prepared to take on more goals, unannounced. This necessity to be able to jump between different goals in real-time lends itself to the CxBR architecture. CxBR agents provide responses that are directly related to its active context. The fact that contexts correspond to accomplishing particular goals combined with the idea that conversational goals take on a very fluid nature yields the assertion that goal management can be facilitated using CxBR methods.

The context topology carries the executed actions of the dialog system. This execution is directed by the goal bookkeeping component. Upon receiving the activated goal to be addressed from the goal stack, the context topology operates on this signal to provide the proper agent response. Each context within the context topology corresponds to a certain conversational task, whether user motivated (external) or agent motivated (internal). Most of these conversational tasks adhere to a specific user task goal. These are known as *User Goal*-Driven Contexts. The remaining conversational tasks constitute the *Agent Goal*-Driven Contexts. The inclusion of all User Goal-Centered Contexts and Agent Goal-Driven Contexts constitutes the entire embodied conversation agent context topology.

3.4 Knowledge Management

One aspect of goal management is goal recognition, the process of identifying which conversational goals need to be addressed. In this paper, a context-driven inference engine performs this service, as dictated by the CxBR architecture. Thus, a strong knowledge base must be in place for proper goal recognition. The information found in this knowledge base is analogous to the rote knowledge that a human learns and manipulates to make decisions. In this paper, three knowledge models are considered: domain-specific knowledge, conversational knowledge, and user-profile

knowledge. The scope and depth of the domain-specific knowledge is modeled after that of a traditional expert system [22] where a domain specialist meticulously adds information to a machine by hand. For this research, the domain-specific knowledge is organized as a semantic network. The conversational knowledge will be formulated as a backbone of common sense knowledge required to accommodate for understanding of basic conversational vocabulary, as seen in WordNet [5] and ConceptNet [16]. Upon identifying the user, the user-profile knowledge allows, the knowledge manager can immediately retrieve his or her individual profile. The information contained in this source may drive the agent's conversational actions using slot-filling [9]. User profiling is particularly important in HCI because it escalates the level of realism and conveys an effect of personalization.

Contextualized knowledge refers to a cross-section of all three knowledge sources that is relevant for the active context of the conversation. Each piece of information within the knowledge manager is annotated with a context tag. Upon determining the context of a conversation, knowledge that is labeled with the current context is elicited as valid information for the conversation and funneled into the contextualized knowledge base. After establishing this subset of information, the dialog manager can then work with a manageable portion of the entire knowledge base. This is especially true when performing goal management, which may require memory-intensive processes. The concept of contextualized knowledge is a novel feature of this dialog system. The idea that only a portion of an agent's entire knowledge is needed at any given time reflects how a human does not require every single fact he or she knows to make decisions. A CxBR-based architecture lends itself to this concept, since the determination of an active context, and therefore an active set of contextualized knowledge based, is a built-in function of CxBR.

4. Agent Evaluation

It is unclear on how to quantitatively describe how well a conversation agent performs, or how much better one is over another. One of the hurdles in this research remains the definition of naturalness, as in how well a chatbot can maintain a *natural* conversation flow. This section discusses some current chatbot evaluation practices, as well as a definition for naturalness for the sake of the proposed dialog system featured in this paper.

Typically, conversation agents have been evaluated subjectively, usually involving a user questionnaire. Semeraro et al [8] employ this technique, using seven appraisal characteristics: impression, command, effectiveness, navigability, ability to learn, ability to aid, and comprehension. Users would declare their satisfaction for each of these metrics, ranging from 'Very Unsatisfied' to 'Very Satisfied.' Semeraro et al recognized that subjective evaluation could not provide statistically verifiable conclusiveness, but instead serves as a general indicator of performance. Shawar and Atwell [20] propose a universal chatbot evaluation system. They suggest three metrics: dialog efficiency, dialog quality metric, and users' satisfaction. They concluded that the proper assessment of chatbots was the end result in how successfully it accomplishes its intended goals. Rzepka et al [29] used a 1-to-10, or Likert, scale for two metrics: a 'naturalness degree,' and a 'will of continuing a conversation degree.' While their assessment system did not identify a concrete baseline for universal naturalness, they were able to make relative measurements of naturalness between different dialog management approaches.

The evaluation system featured in this research will be derived from the PARAdigm for DIalogue System Evaluation (PARADISE) [30]. In PARADISE, the master objective is user satisfaction, which is comprised of task success and dialog costs. Dialog costs consist of efficiency measures and qualitative measures.

5. Conclusion

This paper proposes the general approach for a CxBR-based dialog management system. A literature review was presented to represent the related background efforts in NLP, dialog system design and context-based methods. A high-level description of the dialog system was introduced to identify the elements needed to accomplish the goal management aspect of the conversation agent. Finally, experimentation plans were discussed to provide agent performance evaluation.

6. References

[1] Breazeal, C. (2005). Socially intelligent robots. *Interactions*, 12(2), pp. 19-22.

[2] Branting, K., Lester, J., and Mott, B. (2004). Dialogue Management for Conversational Case-Based Reasoning. *Proc. of the 7th European Conference on Case-Based Reasoning*, pp. 77-90.

[3] Wilks, Y. (2005) The History of Natural Language Processing and Machine Translation, In Encyclopedia of Language and Linguistics, Kluwer, Amsterdam.

[4] Spärck Jones, K. (1964). Synonymy and Semantic Classification. Ph.D. thesis, University of Cambridge, Cambridge, England.

[5] Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K. (1990). WordNet: an on-line lexical database. *International Journal of Lexicography*, 3(4).

[6] Allen, J. F., Byron, D. K., Dzikovska, M., Ferguson, G., Galescu, L., and Stent, A. (2001). Towards Conversational Human-Computer Interaction. *AI magazine*.

[7] Flycht-Eriksson, A., and Jönsson, A. (2000). Dialogue and domain knowledge management in dialogue systems. *Proc. of the 1st SIGdial workshop on Discourse and dialogue*, 10, pp. 121-130.

[8] Semeraro, G., Andersen, H. H. K., Andersen, V., Lops, P., and Abbattista, F. (2003). Evaluation and Validation of a Conversational Agent Embodied in a Bookstore. Universal Access: Theoretical Perspectives, Practice and Experience. *Lecture Notes in Computer Science*, 2615, pp. 360-371.

[9] McTear, M. F. (2002) Spoken dialogue technology: enabling the conversational interface. *ACM Computing Surveys*, 34(1), pp. 90-169.

[10] Sonntag, D. (2006). Towards Combining Finite State, Ontologies, and Data Driven Approaches to Dialogue Management for Multimodal Question Answering. *Information Society Language Technologies Conference*.

[11] Weizenbaum, J. (1966). ELIZA-a computer program for the study of natural language communication between man and machine. *Communications of the ACM*, 9(1), pp. 36-45.

[12] Sansonnet, J., Leray, D., and Martin, J. (2006). Architecture of a Framework for Generic Assisting Conversational Agents. *Lecture Notes in Computer Science*, 4133, pp. 145-146.

[13] Galvão, A. M., Barros, F. A., Neves, A. M. M., and Ramalho, G. (2004). Persona-AIML: An Architecture Developing Chatterbots with Personality. *AAMAS*, pp. 1266-1267.

[14] Wallace, R. (2002). AIML Pattern Matching Simplified. *Technical report*, A.L.I.C.E. AI foundation.

[15] Cassell, J., Ananny, M., Basu, A., Bickmore, T., Chong, P., Mellis, D., Ryokai, K., Smith, J., Vilhjálmsson , H., and Yan, H. (2000). Shared reality: Physical collaboration with a virtual peer. *Proceedings of CHI 2000.*

[16] Liu, H., and Singh, P. (2004). ConceptNet: A Practical Commonsense Reasoning Toolkit. *BT Technology Journal*, 22.

[17] Lee, C., Sidner, C., and Kidd, C. (2005). Engagement During Dialogues with Robots. *AAAI Spring Symposia.*

[18] Traum, D. (2006). Talking to Virtual Humans: Dialogue Models and Methodologies for Embodied Conversational Agents. *ZiF Workshop*, pp. 296-309.

[19] Stensrud, B. S., Barrett, G. C., Trinh, V. C., and Gonzalez, A. J. (2004). Context-Based Reasoning: A Revised Specification. *FLAIRS Conference 2004.*

[20] Shawar, B. A., and Atwell, E. (2007). Different measurements metrics to evaluate a chatbot system. *Proc. of the 2nd Workshop on TextGraphs: Graph-Based Algorithms for Natural Language Processing.*

[21] Bickmore, T. W., and Picard, R. W. (2004). Towards Caring Machines. *Computer Human Interaction.*

[22] Gonzalez, A. J., and Dankel, D. D. (1993). The Engineering of Knowledge-Based Systems Theory and Practice. Prentice-Hall Inc., Englewood Cliffs, New Jersey.

[23] Sidner, C. (2002). Building Spoken Language Collaborative Interface Agents. *Practical Dialogue Systems.*

[24] Fügen, C., Holzapfel, H., and Waibel, A. (2004). Tight Coupling of speech Recognition and Dialog Management - Dialog-Context Dependent Grammar Weighting for Speech Recognition. *INTERSPEECH 2004 8th International Conference on Spoken Language Processing.*

[25] Lieberman, H., Faaborg, A., Daher, W., and Espinosa, J. (2005). How to Wreck a Nice Beach You Sing Calm Incense. *International Conference on Intelligent User Interfaces.*

[26] Sarma, A., and Palmer, D. D. (2004). Context-based Speech Recognition Error Detection and Correction. *HLT-NAACL 2004: Short Papers*, pp. 85-88.

[27] Serridge, B. (1997). Context-Dependent Modeling in a Segment Based Speech Recognition System, M.S. Thesis, MIT.

[28] Young, S. R. (1989). The MINDS system: using context and dialog to enhance speech. *Proceedings of the workshop on Speech and Natural Language*, pp. 131-136.

[29] Rzepka, R., Ge, Y., and Araki, K. (2005). Naturalness of an Utterance Based on the Automatically Retrieved Commonsense. In *Proc. of Nineteenth IJCAI.*

[30] Walker, M. A., Litman, D. J., Kamm, C. A., and Abella, A. (1997). PARADISE: a framework for evaluating spoken dialogue agents. *Proc. of the 8th conference on European chapter of the Association for Computational Linguistics*, pp. 271-280.