The Evaluation of Spoken Dialog Management Models for Multimodal HCIs

Rytis Maskeliunas
Information Technology Development and Automation and Control Systems Institutes,
Kaunas University of Technology, Lithuania

Abstract: *The implementation of voice dialogs enables the realization of some of the aims of modern Human Computer Interaction (HCI) services more successfully and efficiently. Sadly the multimodal Lithuanian HCIs carried by the most natural form of communication-speech are still in the prototype stage and no services are provided to end user at the time of writing. This paper describes an experimental evaluation of the possibilities of using the spoken language dialogs as the main modality in modern application control. The recognition accuracy of the tree main types of spoken dialogues (dictation, keyword spotting, isolated utterances) was evaluated and user preference survey was done on proposed multimodal HCIs. The goal of this research was to gather the results by possible everyday future users not familiar with such systems.*

Keywords: *Spoken dialog, dialog management, HCI, speech recognition, multimodal interactions.*

Received June 25, 2011; accepted December 30, 2012; published online January 29, 2013

1. Introduction

Typical HCI dialog systems cover a well-defined application and perform several tasks within. Such system might be viewed as an interface between the user and the computer. It gathers user input and translates them into specific tasks. For example, in a multimodal dialog system for a mobility device, the user may command the wheelchair using a spoken dialog, a video based dialog (gestures) or a touch based dialog, to perform a basic tasks like setting the direction of moving or going forwards or backwards [12, 16]. Similarly, in a program with a Graphical User Interface (GUI), the user draws might input the necessary information much faster by added voice modality especially if he uses a smartphone or a tablet than by only a typical touch or keyboard input [18]. Another example of a dialog management system for information retrieval are the typical call-center applications that enable a database research on the basis of user requests.

One of the first HCI dialog modeling was done on the air traffic control application simulators [10, 13]. Almost parallel a more advanced study was done on HMI dialog, considering spontaneous speech effects, including disfluencies, hesitations, repeated words and repairs, etc., was done modeling flight traffic information [19]. Another systems used grammar formalism, for example L'ATIS for air traffic [2], MASK [5] and ARISE for train traffic [7], information retrieval.

Current state of the art dialog modeling approaches are based on the use of belief networks [14] and bayesian networks [11]. Some dialogs are modeled combining n-grams and stochastic context-free grammars [6], other propose a stochastic approach [8].

The dialog model provides a general description of the different application related situations: request for information, repetition, confirmation, etc. It also specifies the relations between these situations. Four classic dialog modeling approaches are defined and are recommended used to model HCIs [3], (the structural models based on linguistic knowledge i.e., [15], the plan-oriented models based on artificial intelligence and employ the notions of plan, planning and plan recognition i.e., [9], the logic models based on a modal logic to represent the mental attitude of the interlocutor and the reasoning induced by these attitudes i.e., [1], the task-oriented models are closely related to the application, where the knowledge about the dialog is combined with the task knowledge i.e., [20]. Dialogue tasks [4, 17] in a HCI dialog can be classified in the following way:

- *Learning Tasks:* Knowledge acquisition, where the user is subsumed under teaching or educational tasks.
- *Information Tasks:* The user asks for information in a specific domain (i.e., air traffic schedules).
- *Command Tasks:* The aim of the user is to handle objects in a reference world (i.e., control of a wheelchair)
- *Assistance Tasks:* In certain applications, the user needs to be assisted in decision processes (i.e., translation).

The goal of the research presented below was to gather the results by possible everyday future users not familiar with such systems. The primary voice I/O modality was chosen for the evaluation of the recognition accuracy of the tree main types of spoken dialogues (dictation, keyword spotting, isolated utterances) and user preference survey was done on proposed multimodal HCIs.

2. Chosen HCI Dialog Architecture

The dialog system used for analysis was designed based on a classic architecture of the typical HCI dialog realization as shown in Figure 1.

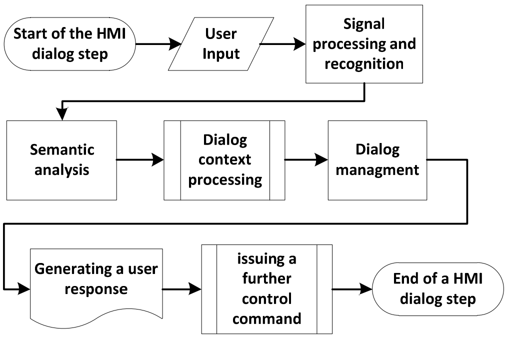
Start of the HMI dialog step
User Input
Signal processing and recognition
Semantic analysis
Dialog context processing
Dialog managment
Generating a user response
issuing a further control command
End of a HMI dialog step

Figure 1. A chosen HCI dialog architecture.

The dialog management models presented in this article use all of the mentioned dialog tasks based on the application being modeled: learning tasks (a user is instructed what to do, i.e., “help command”), information tasks (a user asks and is informed of a specific action, i.e., “in what format should I enter the ID”), command tasks (a user may issue a command, i.e., “clear all fields”, “start-over”, etc.,) and assistance tasks (user is instructed what to do in case of silence, events of incorrect recognition results, etc.). The “algorithm” in principle is very simple can be briefly described in pseudo code. There are four main modalities than can be used in the HCI system: speech processing $y(t)$, video object (gaze) tracking $I(u, v, t)$, touch processing $T(x, y, t)$ and sensor processing $L(\alpha, d)$.

The HCI dialog is executed in 5 stages:

```
Stage 1. The user inputs a signal, either by:
    Speech recognition processing, $s_{asr}(t)\Rightarrow ASR$
    2D video object processing, $V_{rec}(u,z,t)\Rightarrow x_{object}, y_{object}$
    2D touch surface processing, $T_{rec}(x,y,t)\Rightarrow x_{finger}, y_{finger}$
Stage 2. The semantic analysis and action selection depending on the result of the input accuracy function(confidence measure)
$f(accuracy)=I\_value_{measured} - I\_value_{given}$
  If $f(accuracy)>\Theta_{accu}$ Then
      Go to stage 3
  Else
      Go to stage 1
  End_If
Stage 3. The dialog management phase
    Check $f(Rule_{grammar})$
    If $f(Rule_{grammar})\neq 0$ Then
      Go to stage 4
    Else
      Go to stage 1
    End_If
Stage 4. Generate a feedback to the user, either by:
    Speech synthesis, $s_{tts}(t)\Rightarrow TTS$ .
    GUI  $V_{gui}(u,z,t)$
    Haptics on a touch surface, $T_H(x,y,t)\Rightarrow vibration(t)$
  Go to stage 5.
Stage 5. Generate a corresponding command operational instruction to the application backend
  $Output_{cmd}\Rightarrow operation_M$
```

Briefly, in any case a user generates an input signal, which is recognized by the signal recognition component, then it is processed by the semantic analyzer, depending on the syntactic and semantic knowledge contained in the case grammar, the semantic representation of the user input is generated and is stored in the dialog context. Next the task, the dialog model and the other processes in the dialog management module are activated to establish a dialog, to send a command operational instruction to the application backend and to generate a feedback to the user.

3. Dialog Management

The dialog management scenarios below are illustrated only on the case of speech recognition as it mimics the human-human dialogs most closely the same scenarios apply to other modalities and are not repeated in the text (e.g., one can input the same semantic values by finger (either swiping or gestures), by head or hand movements also, by gaze (etc.).

Basically, the dialog management part for a command mode works like this: after a person says something utters a voice command, the input speech signal is processed and the word is checked against the recognition vocabulary if such a command is possible. If the answer is “positive” the confidence value of the recognized phrase is measured and if it is high enough the semantic value is used in further processing. In case of an unclear recognition system sees a few choices as similar an n-best grammar strategy is used and a user is offered not to repeat the phrase, but to choose from some of the possible variations (i.e., “did you say: septintas or devintas?”). After a successful gathering of the input, the semantic value is processed and the application proceeds to the next stage of a dialog. The main advantage of this approach as it is simple to realize, uses simple grammars hopefully resulting in good recognition accuracy, but it is not a natural interface for the user.

To test the user preferences a system was also modified to understand a dictation of detailed instruction a dictation model was added, so a user could form the input data sentences as if he reads from an instruction manual. The biggest disadvantage of such approach is a very complex set of grammar rules and a reduced accuracy of recognition. Another one it is not possible to offer a self-correction list of choices, due to a very same reason complex grammars. Atypical grammar branch of choices one of many is illustrated in Figure 2.

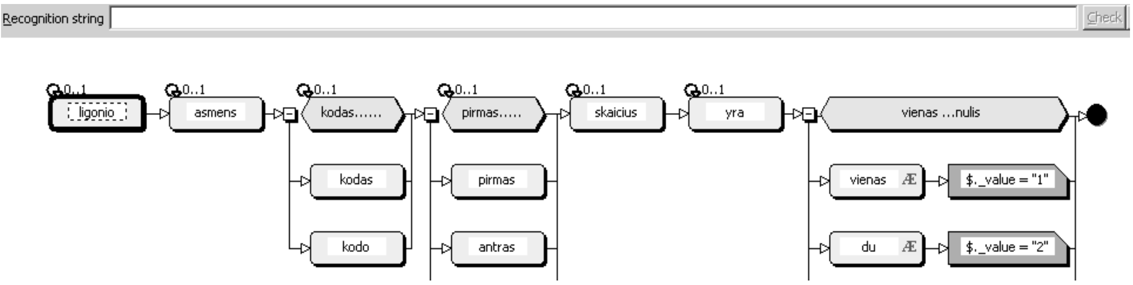

Figure 2. A typical grammar rule branch of answer choices fragment.

In a case of keyword spotting model "keywords" should be understood as the semantically "important" words with a predefined semantic value a system is preprogramed to use a specific set of complex grammar rules. This way a user can speak naturally (for example: "ligonio asmens kodas yra VIENAS, DU, TRYS,...") and a system only catches the important words for this stage of form-filling (in this case "VIENAS, DU, TRYS,..."), assigns the appropriate semantic values and passes for further processing and finally jumps to a next stage in dialog (for example asks to enter the code of an illness). A correction sub-algorithm is also possible in this case, and if available a user is offered a list of selection by voice, or by GUI. The biggest advantage of this approach over the isolated words is the added naturalness, while still maintaining high enough recognition accuracy. The principle is illustrated in Figure 3.

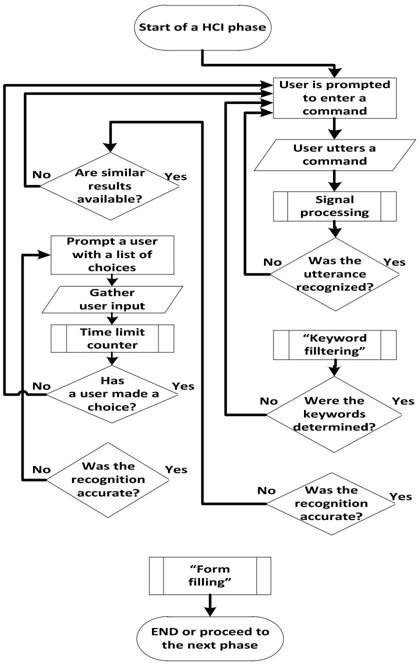
Start of a HCI phase
User is prompted to enter a command
User utters a command
Signal processing
Was the utterance recognized?
No
Yes
"Keyword filltering"
Were the keywords determined?
No
Yes
Was the recognition accurate?
No
Yes
Are similar results available?
No
Yes
Prompt a user with a list of choices
Gather user input
Time limit counter
Has a user made a choice?
No
Yes
Was the recognition accurate?
No
Yes
"Form filling"
END or proceed to the next phase

Figure 3. The schematics of an HCI dialog model capable of fetching the keywords from an input.

4. Experimental Evaluation of the Recognition Accuracy for all Dialog Models

A specific corpus of twelve 5 males, 7 females, age 20-60, native Lithuanians, no speech impairments inexperienced, possible end-users of such systems was built there are no more related Lithuanian language material yet for the experimental evaluation of the 3 chosen models of HCI dialogs. Each speaker was asked to enter a specific set of digit code in various form fields i.e., to pronounce a phrase with a semantic value of a specific digit (0-9). For a more exact evaluation the speaker was asked to repeat the same phrase 100 times. Trying to imitate a possible usage scenario, a test bench was equipped with an average quality Altec Lansig headset and a standard built-in Realtek ALC888 soundcard. All equipment was hoisted in a typical office room (~30-35dB ambience). No additional signal processing such as noise reduction or echo cancellation was enabled. As a Lithuanian speech recognizer a proprietary HMM based system running on a Microsoft Speech Server application platform was used. No other viable alternatives are available for the recognition of the Lithuanian language.

All three previously mentioned dialog models were evaluated. In an isolated words mode a speaker could speak in simple short utterance or just tell the required digits (i.e., One, Two, Three and Eight). In the keywords mode a user could form an input sentence any way he chooses, but the sentence must contain the keyword digit semantic factor, for example (translated from Lithuanian): "The First number of patient's identification code is One"; "The Second number of identification code is Two"; "The Third number of a code is Three"; "The Fourth number is Four"; "The patient's identification code is Six, Two, Seven"; "The identification code is Seven, One, Three, Two"; "The illness code is One, One, Seven" and so on. In a "instruction" mode a speaker was instructed to form an input sentence as if he/she was reading from an instruction manual, thus uttering a long defining the grammatically correct sentence, for example: "The ID code of the patient's illness is One, Two, Three, Four", "The patient's illness was identified as One, Two, Three, Four" and so on. The principle "speak anything you want" was not used due to poor recognition accuracy of dictation by our Lithuanian speech recognizer. The average recognition accuracies of the semantic values of all three modes of spoken dialogs are presented in Figure 4.

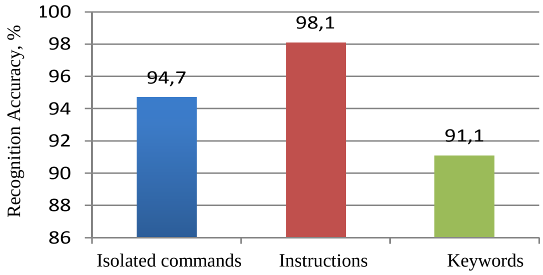

Figure 4. The average recognition accuracies of the semantic values of all the evaluated types of dialog models.

The best recognition accuracy was achieved for the instruction mode. Long and detailed sentences instructions were recognized most accurately by the proprietary recognizer used (98.1%). Isolated utterances were recognized ~3% better than the keyword spotting mode (94.7% and 91.1%). Next the average semantic values were fetched from all recognitions. The overall confidence measure results for all of the three models of spoken dialog are presented in Figure 5.

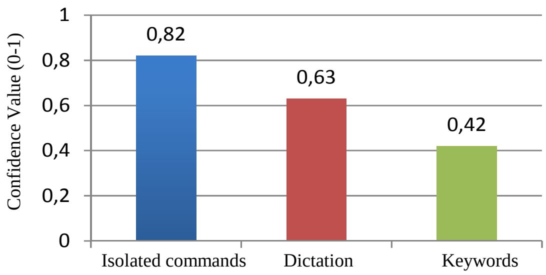

Figure 5. The average confidence measures of the recognitions of all the evaluated types of spoken dialog.

In contrast to the reasonably high recognition accuracy, the overall confidence measures the likelihoods to the "utterance model" in the ASR engine were quite low. The highest reliability highest confidence measure was for the isolated words type of dialogs. The overall rate was ~0.82 compared to 1 being the highest possible number, while the keywords mode came last less than industry acceptable 0.5 ratio was achieved - 0.42, confirming the lowest recognition accuracy of this mode and the possibility of not being very reliable, thus not suitable for real-life use yet. It is important to note that the preliminary results shown here do not mean that dialog trees, such as the naturally sounding keyword mode are not acceptable, but shows the limitations of our recognition system and grammar rules. Further, a more detailed investigation is necessary and will be repeated in the future when more data will be available.

5. The End-User Preference Survey on the Dialog Models

An end-user evaluation of all three human machine dialog systems has been performed by the same participants. We have asked them to subjectively evaluate the HMIs by scoring 1 (very bad) to 10 (superb) according to the following parameters:

- *The Performance:* How fast and easy is the dialog flow i.e., what time it takes to get to the desired goal.
- *The Accuracy:* Not to confuse with the input recognition accuracy evaluate how accurately the application responds to users input, how the utterances and situations are handled, etc.
- *The Naturalness:* How natural the dialog flow is to end-user, comparing to real human persona.
- *The Recall:* How easy it is to remember the control scheme of an application.
- *The Usability:* Overall usability, considering using such types of voice control in day to day application basis aspects.

Overall all inexperience users evaluated all dialog systems as acceptable in Table 1. The ease of use recall of functions, operating instructions, voice commands, etc., and the accuracy of operation were chosen highest (9.1 and 8.8) due to simplicity of operation and straightforward feedback and processing. The performance was rated as quite high 8.1 as the application operated fast and swiftly. Naturalness and usability were rate worst (7.2 and 7.8) because most inexperienced participants expected human like AI and because some of the younger people thought it would be faster just to use a computer or phone keyboard to enter the data, while some of the elderly people might have liked their current "paperwork". All participants agreed that they would like to use such spoken dialog based multimodal systems in ideal conditions, while most indicated that such systems would be usable even in current form of development especially in the telephony call-centers.

Table 1. The average end-user evaluation of the five parameters tested.

Parameter Tested	Average Score (from 1 (Very Bad) to 10 (Superb))
Performance	8,13
Accuracy	8,77
Naturalness	7,24
Recall	9,10
Usability	7,81

6. Conclusions and Discussion

A recognition analysis has shown that the best spoken dialog recognition accuracy was achieved for the instruction mode. Long detailed sentences were recognized most accurately by our recognizer 98.1%. Isolated utterances came close to the keyword spotting (94.7 and 91.1%). In contrast to the reasonably high recognition accuracy the overall confidence measures were quite low. The highest reliability was for the isolated commands. The overall rate was ~0.82 (compared to 1 being the highest possible number). The detailed instructions type of dialogs had similar

0.63 ratio, while the keywords mode were the worst-less than industry acceptable 0.50 ratio was achieved (0.42), confirming the lowest recognition accuracy results for this model and the possibility of not being very reliable, thus not suitable for real-life a current stage of development.

Overall all participating users evaluated all dialog systems as acceptable. The ease of use recall and the accuracy of operation were chosen highest (9.1 and 8.8) due to simplicity of operation and straightforward feedback and processing. The performance was rated as quite high 8.1 as the application operated fast and swiftly. Naturalness and usability were rate worse than the other parameters (7.2 and 7.8) because some of the inexperienced participants expected human like AI.

All participants agreed that they would like to use such spoken dialog based multimodal systems in ideal conditions, while most indicated that such HMI systems would be usable even in a current form of development (especially for the telephony applications. It is important to note that the preliminary results shown in this study do not mean that dialog models, such as the keyword model are not acceptable, but shows the limitations of our system and grammar rules. Further, a more detailed investigation is necessary and will be conducted in the future when more data will be available.

Acknowledgements

This work was done under the "Postdoctoral Fellowship Implementation in Lithuania" project No.: 20101216-90 funded by the Lithuanian Science Council.

References

[1] Allen J., *Natural Language Understanding*, Addison Wesley, 1994.

[2] Bennacef S., Bonneau-Maynard H., Gauvain J., Lamel L., and Minker W., "A Spoken Language System for Information Retrieval," *in Proceedings of the International Conference of Speech and Language Processing*, pp. 1271-1274, 1994.

[3] Deutsch B., "The Structure of Task Oriented Dialogs," *in Proceedings of the IEEE Symposium on Speech Recognition*, Pennsylvania, pp. 1-14, 1974.

[4] Di-Fabbrizio G. and Stent A., "Learning the Structure of Task-Driven Human-Human Dialogs Bangalore," *IEEE Audio, Speech, and Language Processing*, vol. 16, no. 7, pp. 1249-1259, 2008.

[5] Gauvain J., Bennacef S., Devillers L., Lamel L., and Rosset S., "Spoken Language Component of the MASK Kiosk," *in Proceedings of Human Comfort & Security of Information Systems*, Berlin, pp. 93-103, 1997.

[6] Hacioglu K. and Ward W., "Dialog-Context Dependent Language Modeling Combining n-Grams and Stochastic Context-Free Grammars," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, UT, vol. 1, pp. 537-540, 2001.

[7] Lamel L., Rosset S., Gauvain J., and Bennacef S., "The LIMSI ARISE System," *in Proceedings of IEEE 4th Workshop Interactive Voice Technology for Telecommunications Applications*, Torino, pp. 209-214, 1998.

[8] Levin E., Pieraccini R., and Eckert W., "A Stochastic Model of Human-Machine Interaction for Learning Dialog Strategies," *IEEE Speech and Audio Processing*, vol. 8, no. 1, pp. 11-23, 2001.

[9] Litman D. and Allen J., "A Plan Recognition Model for Subdialogues in Conversations," *Technical Report*, Rochester University Ny Department of Computer Science, pp. 163-200, 1987.

[10] Marque F., Bennacef S., Neel F., and Trinh S., "PAROLE: A Vocal Dialogue System for Air Traffic Control Training," *in Proceedings of Applications of Speech Technology*, Germany, pp. 91-94, 1993.

[11] Martinez F., Ferreiros J., Cordoba R., Montero J., San-Segundo R., and Pardo J., "A Bayesian Networks Approach for Dialog Modeling: The Fusion BN," *in Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 4789-4792, 2009.

[12] Maskeliunas R. and Rudzionis V., "Multimodal Interface Model for Socially Dependent People," *in Proceedings of Analysis of Verbal and Nonverbal Communication and Enactment, Lecture Notes in Computer Science*, Berlin, vol. 6800, pp. 113-119, 2011.

[13] Matrouf A., Gauvain J., Neel F., and Mariani J., "Adapting Probability-Transitions in DP Matching Process for an Oral Task-Oriented Dialogue," *in Proceedings of International Conference on Acoustics Speech and Signal Processing*, vol. 1, pp. 569-572, 1990.

[14] Meng H., Wai C., and Pieraccini R., "The use of Belief Networks for Mixed-Initiative Dialog Modeling, Speech and Audio Processing," *IEEE Transactions*, vol. 11, no. 6, pp. 757-773, 2003.

[15] Ostler N., "LOQUI: How Flexible Can a Formal Prototype Be?," *The Structure of Multimodal Dialogue*, pp. 407-416, 1989.

[16] Rudzionis V., Maskeliunas R., and Rudzionis A., "Assistive Tools for the Motor-Handicapped People using Speech Technologies: Lithuanian Case," *in Proceedings of Business Information Systems Workshops, Lecture Notes in Business Information Processing*, vol. 97, pp. 123-131, 2011.

[17] Sarikaya R., Gao Y., Erdogan H., and Picheny M., "Turn-Based Language Modeling for Spoken Dialog Systems," *in Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. I-781-I-784, 2002.

[18] Trabelsi Z., "A Generic Multimodal Architecture for Integrating Voice and Ink XML Formats," *International Arab Journal of Information Technology*, vol. 1, no. 1, pp. 93-101, 2004.

[19] Ward W., "Extracting Information in Spontaneous Speech," *in Proceedings of International Conference of Speech and Language Processing*, pp. 83-86, 1994.

[20] Young S., Hauptmann A., Ward W., Smith E., and Werner P., "High Level Knowledge Sources in Usable Speech Recognition Systems," *Communications of the ACM*, vol. 32, no. 2, pp. 183-194, 1989.

Rytis Maskeliunas received his PhD degree in computer science, in 2009 from Kaunas University of Technology, Lithuania. He is a senior scientific researcher and a project manager in computer science field at Kaunas University of Technology, Information Technology Development and Automation and Control Systems Institutes, with an expertise in development and analysis of multimodal interfaces, automatic speech recognizers. He has won various awards/honours including the National Science Academy Award for Young Scholars of Lithuania in 2010, the Postdoctoral Research Fellowship 2010, the Best Master, in 2004 and Master Work, 2006. He has coordinated/participated in several research projects in computer science domain and was involved in the EU COST actions 278, 2102 and is an MC member (Lithuania) of the currently running COST IC1002. He is a member of an IEEE, author/co-author of over 30 refereed scientific articles and serves as a reviewer for a number of refereed journals. His research interest includes modelling, development and analysis of multimodal interfaces, engineering of virtualization systems, programming web and telephony servers and applications.