

# Wizard of Oz Experiment for a Telephony-Based City Transport Dialog System

Danijel Koržinek, Łukasz Brocki, Ryszard Gubrynowicz, and Krzysztof Marasek  
Polish-Japanese Institute of Information Technology, Warsaw, Poland

## Abstract

One of the primary uses for computer dialog systems is automated telephone services and call centers. This paper explores the prospect of employing such a system in the call center of the Warsaw Transport Authority, an organization that governs the public city transport of Warsaw (Poland). The research was carried out through a Wizard-of-Oz study and is arguably the first experiment of such scale in Poland.

**Keywords:** wizard-of-oz, dialog system, telephony, speech recognition, spoken language understanding

## 1 Introduction

Dialog systems (De Mori, 1998) are a very complicated topic in the research of human-machine interaction. The goal is to make a computer system to communicate with the users by voice and to create an illusion of a real person. In telephony, this encompasses natural-sounding speech synthesis, accurate speech recognition and intelligent reasoning. This is extremely hard to achieve, yet more and more systems are being introduced all the time.

Worldwide, these systems vary in quality. The corpus-based speech synthesis sounds fairly natural most of the time, but in order to create better impression of human voice it is often replaced by the recordings of professional speakers, if the domain allows such a solution. Speech recognition in such systems ranges from being able to recognize several to several hundred words at a time. Ultimately, the biggest problem lies in the reasoning component of the system. In some cases, the dialog can be reduced to a finite set of questions and answers, but people seldom obey such rigid schemes and even the most thoroughly engineered dialogs are often broken by their users. This is the reason why many of the systems serve merely as a front-end to traditional, human operated centers.

In Poland, there are no real dialog systems in use to date. That is why, when confronted with an automated system, many users don't know how to follow the step-by-step instructions generated during the human-machine dialog. One of the motivations for this experiment was to evaluate the reaction of people to this new technology.

The experiment presented in this paper is a part of research done within the LUNA project, which tries to take the idea of dialog design one step further. The

goal of the project is to develop a toolkit that would provide a mechanism for utilizing the state-of-the-art methods of speech recognition and spoken language understanding in a manner that would simultaneously empower and simplify the dialog design process.

This project is a group effort of several research organizations from several countries. So far, our research group (Polish-Japanese Institute of Information Technology and Institute of Computer Sciences of the Polish Academy of Sciences) has successfully collected and annotated a set of human-to-human dialogs recorded in the call center of the Warsaw Transport Authority (Mykowiecka et al., 2007). The next part of the project is to collect a similar, but human-to-machine set of dialogs. Since there were no adequate speech recognition systems for the Polish language available at the time, a wizard-of-oz (Kelley, 1985) approach was adopted.

## 2 Experiment Description

The goal of the experiment was to record 500 human-machine dialogs. The domain of the dialog had to be similar to the human-human recordings recorded earlier. The dialogs should mimic the functionality of a plausible automated dialog system as close as possible. At the same time, it was crucial not to disrupt the normal operation of the call center where the experiment was carried out.

The dialogs were designed in such a way that the computer would acquire all the information from the user by asking questions and the final answer would be provided by the human operator. Because there are a lot of different possible answers to the problems posed by users, it would be very hard to design an efficient interface for providing these answers in a wizard-of-oz fashion. Nevertheless, most of the dialog was performed correctly and the final answer was usually predictable and irrelevant to the experiment.

The dialog domain was split in the following categories:

**Time schedule**, where the system asks for the line number of the transport in question, the stop name, the direction of travel (there are usually two) and the approximate time the person wishes to travel. The operator responds by providing several departure times before and after the requested time.

**Search route**, where the system asks for the beginning and the destination of travel and the operator provides the quickest route.

**Reduced fares**, where the system determines the type of the passenger by asking a series of questions and then the system provides an answer containing the type of the fare reduction and the required documents that entitle the passenger to the given reduction. After that the user may choose to hear the information again, ask for further information (which connects him to the operator) or disconnect.

**Lost and found**, where the system asks for the time and place of the lost item and the operator provides the information on how to contact the appropriate office for found items.

**Complaints**, where the system asks for the location and the time of the cause of complaint and allows the user to record a description of the occurrence.

**Other information**, which connects directly to the operator.

The experiment was carried out using a modified telephony portal platform.

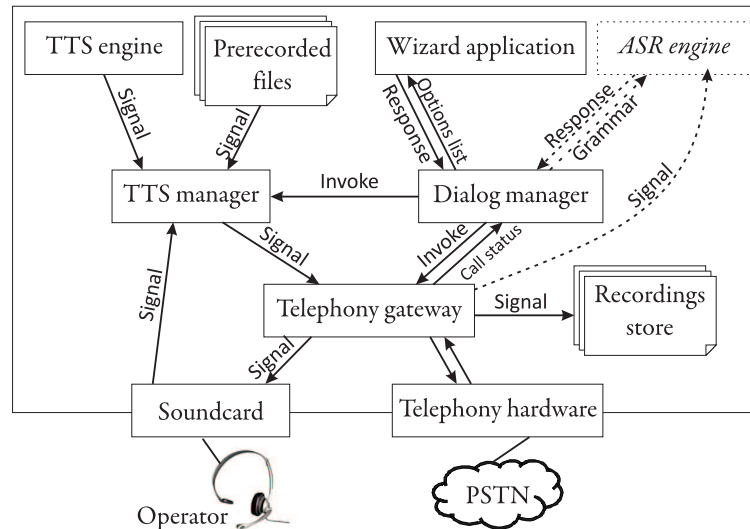


FIGURE 1: System architecture. The ASR component (dashed) is replaced by a human-operated GUI application. Also, all the conversations are both monitored through the soundcard and recorded on the hard disk.

This system normally consists of a computer with specialized telephony hardware and custom software for speech recognition and speech synthesis. It also contains a module for simple, menu-based dialog design. In the modified version of the system (fig. 1), the automatic speech recognition component was replaced by a simple windows application that allows the user to act as the "wizard" in the system. Other modifications included letting the user listen-in on the conversation between the human and the computer and the ability to let the operator talk to the user using a microphone.

The experiment was carried out using one of the phone lines of the call center. When a call would arrive, the caller would be greeted by a short message and a list of options for a given menu. Once the user spoke one of the options, the operator chose an appropriate option from a list visible on his computer screen. This action moved the dialog to the next menu, where the user heard another set of options, which were also immediately displayed on the operators screen. This exchange continued on until the operator was presented with a single option, to speak with the user directly. Besides that, the operator could in any menu choose to: ask the user to repeat what he said, terminate the dialog and speak to the user directly or disconnect the call.

### 3 Experiment Results

The experiment took 13 days to complete with, on average, 5 hrs of recording per day. 844 people called, of which 459 spoke through one of the categories, 155 asked to talk with the operator and the rest either didn't want or didn't know how to

use the system. The breakdown of calls to different categories is presented in table 1.

Category	Number of calls	Percentage of dialogs	Average duration (HM)	Average duration (HH)
Time schedule	137	30%	93.78	62.13
Search route	215	47%	144.86	100.05
Reduced fares	39	8%	141.17	61.14
Lost and found	11	2%	128.02	N/A
Complaints	57	12%	116.06	N/A

TABLE 1: Experiment results. HM – human to machine. HH – human to human. Durations are in seconds.

User reactions to the system were mixed. Most people felt surprised and the first set of options, where the user has to choose one of the categories, had to be repeated at least once. Some people were pleased with how the system worked, but some were irritated. It has to be noted that many of the callers were regular clients of the call center and felt the system was unnecessary.

“Time schedule” was one of the best working categories. Providing that the callers knew all the required information, the operator was able to produce an answer with much less hassle, as the information was organized and complete. However, when the user wasn’t prepared, the system failed.

The “route searching” category was too difficult to design robustly, because the start and destination points are very hard to define. People rarely know the exact address of both locations and often use idiosyncratic names. These names are not understandable by a computer and sometimes even by the human operators.

“Reduced fares” was the only category that did not require a human operator to provide an answer. However, the only information that could have been incorporated this way was available in other interfaces, such as fliers and internet pages. Most people that called regarding this category wanted to ask either the questions that were not answered in other media or that were not clear.

“Lost and found” category was seldom used, but fulfilled its purpose, just like the time schedule category, provided that all required information was available.

Finally, the “complaints” was the favorite category among the human operators. Since this task amounted to the user providing information, without expecting any reply, this was the only task that could have been achieved completely automatically almost all the time. Before the experiment, people would often get very emotional when complaining to the human operators. The automatic system allowed for retaining the same functionality of acquiring the necessary information, without the added stress factor.

The durations in the table are lengths, in seconds, of both the WoZ and the human operator portion of the call. The human-human durations were not all available because the earlier experiment (Marasek and Gubrynowicz, 2007) had a slightly different design. The values are not representative of a fully automated system.

## 4 Conclusion

The Warsaw Transport Authority call center proved to be a very difficult task for automation. The authors of the experiment gained a lot of insight in the flaws and improvements of their initial design. Some of the categories, like the time schedule and lost and found show hope, while search route seems too difficult to solve without some expert human knowledge. Categories like reduced fares also seem promising, especially if they could be occasionally updated with frequently asked questions.

Finally, the user experience, even if harsh sometimes, seemed to be improving over time. By the end of the third week some people were completely at ease with the system. It is our hope that as the level of the technological improvement of dialog systems raises so will the acceptance by the users and their satisfaction with these systems.

## 5 Acknowledgments

The authors would like to thank the Warsaw Transport Authority for their exceptional help. This research is funded by the Luna (IST 033549) project.

## References

- John F. KELLEY (1985), *CAL—A Natural Language program developed with the OZ Paradigm: Implications for Supercomputing Systems*, First International Conference on Supercomputing Systems (St. Petersburg, Florida, 16-20 December 1985), New York: ACM, pp. 238-248
- Renato DE MORI(1998), *Spoken Dialogues with Computers*, Academic Press, London, 1998.
- Agnieszka MYKOWIECKA, Krzysztof MARASEK, Malgorzata MARCINIAK, Joanna RABIEGA-WISNIEWSKA and Ryszard GUBRYNOWICZ (2007), *Annotation of Polish spoken dialogs in LUNA project*, 3rd Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, Poznan, Poland, 2007
- Krzysztof MARASEK and Ryszard GUBRYNOWICZ (2007), *Polish human-human spoken dialog transcriptions—experience from LUNA project*, 38th Poznan Linguistic Meeting, Gniezno, Poland

