Danijel Koržinek
Krzysztof Marasek
Łukasz Brocki

Polish-Japanese Institute of Information Technology

INFORMATION ARCHITECTURE ASPECTS IN VOICE PORTAL DESIGN

Introduction

Voice User Interfaces (VUIs) have been in common use for many decades [DeMo98], but never gained the wide mainstream attention as other forms of Human-Computer Interaction (HCI). The many reasons for this include the complexity and limited practicality in most scenarios. On the other hand, the systems that absolutely relied on such technology, e.g. telephony call centers, were not accessible by most engineers studying HCI. Given the few companies worldwide developing the VUI technology, much of its HCI concepts were discovered independently from the rest of the industry.

On the other side, we see enormous efforts in developing new concepts and approaches to HCI design. The popularization of the computer as a common household appliance allowed it to flourish not only as a consumer product but also as a cheap and easy platform for engineers. The discovery of the Internet significantly increased the usefulness and the popularity of computers, but also created new challenges and opportunities in the study of HCI. Thanks to millions of engineers working on the development of various components of the computer technology, we have witnessed a very rapid advancement in all its aspects, including HCI.

Ultimately, when it comes to the HCI studies, certain technologies, like VUIs have been lagging behind more popular ones, like the web. That is why in certain situations, it became a common practice to transfer the knowledge gained from one domain of HCI to another. This isn't always possible and there are some obvious limitations, as discussed in this paper.

The authors of this work have been given a task to create a simple automated voice activated telephony service to aid the already existing call center used for providing information about the public city transport system of the city of Warsaw,

Poland. Having little previous experience in designing such systems, the authors used the common principles used in the rest of HCI. This included concepts of User-Centered Design [MaBKSG09] and Information Architecture (IA) [RoMo98].

This paper describes the process and tires to rationalize its usefulness in VUI design. The following chapter describes some basic concepts behind VUI design. After that, the notion of IA and its application to VUI is shown in more detail.

VUI Design Concepts

VUI is a term used for software components that allow the user to interact with the computer using voice. It relies on the technologies of Automatic Speech Recognition (ASR) which is the ability to translate the user's voice into text or commands understandable by the computer and Speech Synthesis, also known as Text-To-Speech (TTS), which allows the text presented by software to be read out loud as synthesized speech.

A VUI is also the core component of a Dialog System. Dialog Systems are one of the most sought after pieces of technology that would allow people to communicate with computers in the same way as with other human beings. While it still ultimately belongs to the realm of science fiction, dialog systems have been implemented with reasonable success in various limited domains, such as call centers. One more component, that is probably the main cause behind the Dialog System's letdown is known as the Dialog Manager, which is a component that should understand the meaning of the users' speech as processed by ASR and provide answers that can later be synthesized using TTS. This is a problem, which has a lot in common with pure AI, although it is not impossible to solve for very limited domains, such as certain call center tasks, e.g. ticket reservation, travel information, simple troubleshooting, etc.

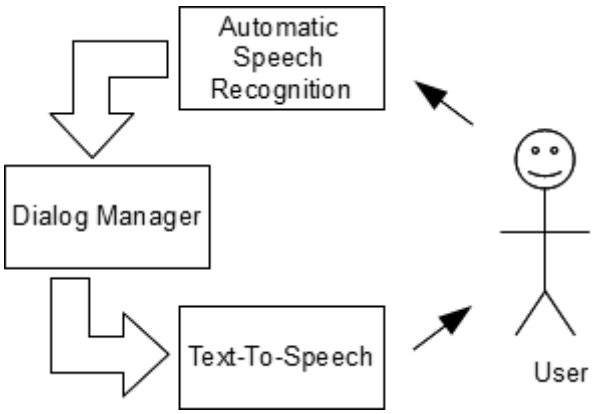

Fig. 1. A diagram of a typical Dialog System and its components

This work is based on the example of an automated call center that provides information on the public transport system in the city of Warsaw. This includes timetable, routing and ticket information for the city's bus, tram, subway and city rail lines. The reason such a system was developed was to reduce the workload of human operators that normally service the call center. Because increasing the number of human operator posts is usually quite expensive (it entails not only labor costs, but also the cost of property), such technology is a popular alternative, even if it doesn't solve the problem completely. A reduction of as little as a few dozens of percent of calls that can be automated is enough for justifying the automated system costs. Apart from lowering costs, such systems are known to help user satisfaction by reducing waiting times, therefore increasing the speed and availability of information. Finally, there are several methods of automating a call center, but none of them is as intuitive as a VUI. This means that a user can speak to a computer just as with another human being and not have to learn the structure and idiosyncrasies of a new system [CoGB04].

All these advantages don't come for free however. There are many limitations of VUI systems compared to other well-known interfaces (Table 1).

Table 1

Comparison of interfaces

	VUI	WWW
Time	Constantly changing – requires memorization. Limited context	Static – always present. Wider presentation of context
Presentation	Sequential	Lots of information in parallel
Tempo	Flow of information controlled independently from the user – computer controls the tempo	Processing of information at user's own pace
Mental model	Cannot predict the next step of the dialog	Clear model
Error correction	Difficult, errors are likely to happen and hard to fix	Easy – errors are easily fixed an system is not error prone
Latency	Slow access to information	Almost immediate

The main characteristics of UIs include:

- modality – text, GUI, voice, touch, etc.,
- style – command driven, menu driven, natural,
- initiative – system, user, mixed,
- domain.

The analyzed public transport call center system was designed as a part of a EU FP6 funded project, LUNA which had a goal of creating a framework for a natural, mixed-initiative dialog systems in any limited domain and while this was also our own research goal, the practical solution was to rely on a well-known menu-driven style and reduce the influence of user initiative in most situations throughout the dialog.

User-Centered Design (UCD) is a common method of designing systems with unknown specification. Unfortunately, in many situations we do not have the luxury of possessing a system to run usability tests on users. This is where a Wizard-of-Oz (WoZ) comes heavily into play and helps resolve major questions, at least until we manage to construct the first prototype for further tests. This was also important in our project [KoBGM08], because it allowed us to collect and test a lot of details about the Information Architecture of the system even before dealing with complex issues such as speech recognition and dialog management. The results of this UCD backed WoZ study are presented in the following sections.

Information Architecture Features

Information Architecture (IA) is a study that tries to emphasize the role of information and its flow within the system [IAIN07]. While other studies deal with similar issues from different points of view, IA becomes very relevant when the system becomes large and too complicated to be understood by the user at a single glance. A common mistake is to give the user too much information at once, making it difficult to find what is relevant. Likewise, providing too little information makes it inconvenient and tiresome to reach the same. Proper categorization and organization of information is one of the main goals of IA.

Ontology

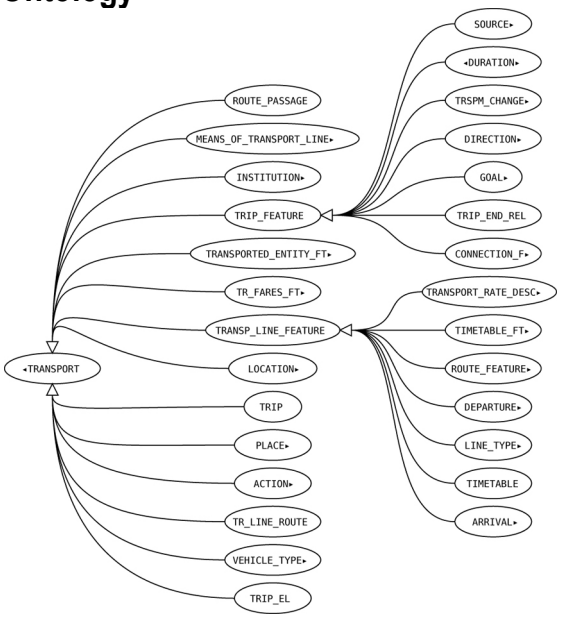

Fig. 2. Breakdown of the "Transport" class in the system ontology

A helpful tool for creating a proper IA analysis of a system is the creation of an ontology. On the example of the call center project, the ontologies were created by a group of linguists that analyzed the recordings of human-human and later also human-computer recordings made during the WoZ phase. They describe the linguistic content of the dialog between the user and the system, interfaced by the call center operator. The preparation of the ontology is even more straightforward and useful in the speech modality as the method of communication is actually the natural language and the concepts described are based on dialog turns, sentences and phrases. Not all the human-human dialogs were described in detail, as people tend to lose focus and chat about various unrelated topics (e.g. weather) when talking to human operators, so only sentences related specifically to the public transport domain were chosen. All the details about this procedure can be found in the book describing the project [Marc11].

Categories

After analyzing the recorded dialogs and studying their ontology, a set of categories was constructed. This was done not only based on user needs but also the functional requirement of the system. It is important however, that both of these are seriously taken into account, otherwise the system might suffer either from being difficult to use or lacking necessary features. The categories chosen for the system were the following:

– timetables – times that buses/trams depart specific stops,
– route lookup – how to reach a destination from a certain location in the city,
– ticket prices and fare reductions – cost of tickets and reduction of these for certain customers (e.g. seniors, students, disabled, etc.),
– lost and found – information about finding lost items in public transport vehicles,
– complaints – general complaints about various issues, e.g.: public transport not on time, driver misconduct, malfunctions, etc.,
– other – anything that doesn't fit in the above categories.

From studying the dialogs, it was apparent that most calls were regarding the first two topics, with route lookup being slightly more favored (timetables were available using other means as well, while route lookup required some prior knowledge of the system). Nevertheless, other categories were also very popular, especially at certain times, e.g.: ticket prices and fare reductions were popular at the start of school year, when students buy their tickets, and complaints were more popular during traffic jams induced by bad weather or traffic accidents.

For practical reasons, not all categories ended up in the final product. When it was infeasible to fulfill a category, it was simply redirected to the human operators, which were still hired to deal with the last category. The "route lookup" for example, was later deemed unrealizable due to the lack of proper language-to-geolocation mapping. It was still useful to leave the proper categorization both to make the user aware of the information they could acquire and to leave the possibility of creating such a solution later.

Attributes

The next step after determining the categories is looking up the attributes for each category. The attributes determine all the information we need to acquire from the user before being able to provide the answer. This is the list of attributes for the individual categories mentioned above:

- timetable: date, line, place (street, stop), direction, time,
- route lookup: start, destination, (time),
- fare reductions: passenger category,
- ticket prices: ticket category,
- lost and found: line/location, time,
- complaints: nothing,
- other: nothing.

An important concept that comes in play here is the ergonomics of the dialog. When we have to ask the user a lot of questions, we have to do this as simply as possible. In case of the first two categories, all the attributes need to be acquired to able to determine the answer, but sometimes certain questions can be skipped or made simpler.

In the timetables category for example, it is essential to know which date the query is about since the schedules are different for work days and holidays. On the other hand, 90% of users are interested in the today's schedule and that is why the first question is "Are you interested in the schedule for today?" and the date question is reserved only for those that answer "No". This is because the procedure for recognizing the dates can become quite time consuming, especially when it comes to misrecognition and error correction.

The "ticket prices" and "fare reductions" categories, on the other hand, have only one attribute with many possible values. To make things more complicated, these values tend to change over time with the city coming up with new rules every couple of months. The ideal ergonomic solution to this problem is a tree structure. In order to reduce the amount of questions we try to group similar val-

ues into subcategories and first ask about those, e.g.: are you a student, a pupil or a senior citizen? Then if the answer is "student", we ask about: are you a student of a domestic or a foreign school? A good guideline is to keep the number of options to no more than 5-6 at each step of the dialog, but to reduce it to even less if possible. Using heuristics based on the analysis of calls is also possible here. In our case, for example, it was determined that most calls about this category were made by seniors above the age of 70 which are eligible to totally free fares. This was in fact a tip that was given to us by experienced call center operators. By asking about the age of the customer as the first question in the dialog (i.e. "Are you older than 70?"), it was possible to rule out about half of the calls without having to make them listen through a long list of categories.

The last two categories are marked to have no attributes. This is because they were designed as such that the system doesn't require any interaction from the user and can immediately initiate an action. In case of complaints, the content of a complaint is simply recorded and stored as an MP3 file that can be later listened to by a member of the staff responsible for filing complaint reports and distributing them to proper units within the organization. The "other" category is simply redirected to the human operator, as mentioned before.

Labels

One final, and often looked over, concept is the idea of proper labeling. When creating a familiar system, especially if it is to replace an older one, it is important to use such labels that users are going to be familiar with. For us, this meant that we had to use names of categories that are as close to the human-human dialogs as possible. These are things that are often based on information far from the system developers reach (e.g. terminology present in information banners and manuals predating the system even by decades) and it would be foolish to try to change them. Such decisions are also sometimes consulted with linguists that posses a greater familiarity of local language use.

Ultimately, the only way to see how this works is to test the system with users, see their reaction and not be afraid of making drastic changes if necessary. Running several stages of WoZ tests can help greatly with reducing costs, but issues may sometimes arise months after the system is fully developed. An interesting example in our project was related to a simple instruction that explains how to connect to the human operator in case the system doesn't behave as expected – a very important feature of the system. In one specific call that lasted for several minutes it was made apparent that the users didn't understand what to

do when instructed “please press any key on your phone to connect to the operator”. Even though from the technical point of view, it made no difference which key the user has to press, changing the prompt to say “press zero to connect to the operator” made the system much easier to use, from the users’ point of view.

Conclusion

In the past, many new directions in the study of HCI were discovered and rediscovered by different people in different domains [Guen12]. This has led to repeated calls to merge knowledge from various similar domains rather than partitioning them even further [Boga12]. It was our goal here to demonstrate that just such an opportunity exists with applying known HCI concepts to the VUI technology. Other work found similar lack of study in VUI design [Skór11]. This is related not only to the IA, as mentioned in the title but also other domains that or out of scope of this work.

In our example project, we managed to apply the concepts of IA to design a system that is still in successful use long after the research project it was designed for. The scope of our work is obviously limited to a specific problem domain and more research should be performed for other tasks. With VUI becoming more viable and popular every day, it is our hope that the wider HCI community will see it as a possible avenue of research to apply their experience.

References

[Boga12] Bogaards P.J.: From cUStomER to CusTOMer: CX as the Naturally for UX in Business. Polish IA Summit, 2012.

[CoGB04] Cohen M.H., Giangola J.P., Balogh J.: Voice User Interface Design. Addison-Wesley Professional, 2004.

[DeMo98] De Mori R. (ed.): Spoken Dialogues with Computers. Academic Press, 1998.

[Guen12] Guenther M.: Intersection. How Enterprise Design Bridges the Gap between Business, Technology and People. Morgan Kaufmann, 2012.

[IAIN07] Information Architecture Institute: What is IA? www.iainstitute.org, 2007.

[KoBGM08] Koržinek D., Brocki Ł, Gubrynowicz R., Marasek K.: Wizard of Oz Experiment for a Telephony-Based City Transport Dialog System. Proc. Intelligent Information Systems Conference, Zakopane 16-18 June 2008.

[MaBKSG09] Marasek K., Brocki Ł., Koržinek D., Szklanny K., Gubrynowicz R.: User-Centered Design for a Voice Portal. Proceedings: Aspects of Natural Language Processing, 2009.

[Marc11] Marciniak M. (ed.): Anotowany korpus dialogów telefonicznych. Exit, 2011.

[RoMo98] Rosenfeld L., Morville P.: Information Architecture for the World Wide Web. O'Reilly & Associates, 1998.

[SkórII] Skórka S.: Systemy nawigacji w przestrzeni słuchowej. Analiza porównawcza. Conference Paper, 24 August 2011.

ASPEKTY ZARZĄDZANIA INFORMACJĄ W PROJEKTOWANIU PORTALU GŁOSOWEGO

Streszczenie

Niniejszy artykuł opisuje zasady projektowania Portalu Głosowego wykorzystywanego w informacji telefonicznej komunikacji miejskiej oraz wpływ Zarządzania Informacją (ang. Information Architecture, IA) podczas jego budowy. IA jest najczęściej omawiane w kontekście projektowania stron internetowych i aplikacji graficznych, ale wiele z jego aspektów można zastosować do innych zadań projektowych. W artykule tym zanalizowano, jak można wykorzystać IA w procesie tworzenia systemu, w którym jedynym sposobem interakcji z użytkownikiem jest mowa. Głównym zadaniem tego systemu jest przekazywanie użytkownikom informacji o komunikacji miejskiej przez telefon. Cel ten osiągnięto poprzez użycie systemu dialogowego wykorzystującego technologię rozpoznawania i syntezy mowy. Procedurę IA zastosowano w kilku fazach projektowania dialogu, zwracając szczególną uwagę na różne ograniczenia tego nietypowego podejścia do interakcji z użytkownikiem. Głównym celem jest zaprojektowanie systemu dostarczającego żądanych informacji w szybki i wygodny sposób. Jest to trudne z kilku powodów: kosztu i ograniczeń czasowych, braku doświadczenia użytkowników w interakcji z podobnymi systemami, braku wglądu w ogólną strukturę systemu, co wymusza na użytkowniku wizualizację struktury we własnym umyśle, treściwej prezentacji znaczącej ilości informacji. Typowych rozwiązań takich problemów, obejmujących nawigację, przeglądanie i wyszukiwanie, nie można bezpośrednio zastosować w takim medium, ale wszystkie mają własne, interesujące odpowiedniki.