Spoken interaction on mobiles *

Caroline Voeffray
University of Fribourg
1700 Fribourg
Switzerland
caroline.voeffray@unifr.ch

ABSTRACT
Spoken interaction on mobile devices is a hot topic. Today different ways are used to interact with mobile devices, for example touch or speech-based interaction. These technologies were created to replace traditional typing input; making a more usable and attractive interface. There exist two kinds of speech recognition technology: speaker-dependent or speaker-independent algorithms. There are several ways to do voice interaction, the main methods being dictation and long queries. Some problems like errors correction, are appearing with these new applications. The most famous speech-based application is Siri, which is included with the iphone 4GS. This paper presents a short overview of existing technologies for spoken interaction on mobile devices. The focus being on what exists today and the cost and problems associated with these technologies.

General Terms
Spoken interaction on mobiles

Keywords
Emotion HCI, Affective HCI, Affective technology, Affective systems

1. INTRODUCTION
Today the trend in user interface research is to replace the standard keyboard input with a more interactive interface. New applications allow users to interact in a manner similar to their non-computing lives. They can touch the screen with fingers or speak to the device to complete an action. Buttons and text typing are being replaced by this new technology. A lot of potential applications are possible using speech recognition with mobile devices. Indeed easy-to-use user interfaces are decisive for mobile devices. In the coming few years speech-based interfaces will strongly increase because several elements facilitate the arrival of this new type of interaction. Firstly, mobile phones are everywhere. Secondly speech capture is very easy with these devices and thirdly, the technology for speech recognition is improving rapidly. Finally, small devices with touch screens are not convenient for typing text, so spoken input is a more pleasant and natural way to operate these devices. For a long time the main goal of interface design was to facilitate the interaction with more natural modes, and recent technological advances have enabled this realization. Thus, voice interaction interface has potential to be the technology that enables future miniaturization of devices[4],[1].

The remainder of this paper is organized as follows. First, explanation about techniques and costs of speech recognition, followed by existing ways to use spoken interaction on mobile device. To continue some problems and their solutions are discussed, then error handling with speech-based applications is discussed. Finally an example of an application using with spoken interaction on a mobile device is presented.

2. SPEECH RECOGNITION
The speech recognition on mobile devices is different from general speech recognition. Some basic properties of general ASR applications are the same but some requirements and characteristics are special and specific to mobile devices. Two different techniques for Automatic Speech Recognition (ASR) exist, speaker dependent technology or speaker-independent technology. The complexity and the robustness of these techniques is different and each technology has its benefits and drawbacks. The technology is selected according to the application and the desired use. Each solution generates costs that must be included during the implementation[2]. The speech control on a mobile device must always be as transparent as possible for the users because when a user purchases a mobile device, they are not buying a speech recognition system. External factors like design, usability and the "cool factor" play important roles when a user chooses between different ASR systems. Today ASR is an imperfect technology, so it is essential to use all possible techniques, which maximizes the recognition performance. The speech recognition should not be dependent on special hardware because only the most technically savvy users will use the ASR feature and the goal is to attract as many users as possible[1].

2.1 Techniques

*Seminar Multimodal Interaction on Mobiles Devices : https://diuf.unifr.ch/main/diva/teaching/seminars/seminar-multimodal-interaction-mobiles-devices

There exist two kinds of Speech recognition technology, the speaker-dependent and the speaker-independent solutions. Some various technical issues must be solved to make the transition from speaker-dependent to speaker-independent ASR. The automatic speech recognition service can be implemented on a network server or it can be distributed between the device and the server. Unlike a full implementation on terminals, the use of efficient networks servers allows for more versatile and computationally complex ASR applications[1].

2.2 Speaker-dependent

Nowadays in wireless devices, principally in mobile phones, the speaker-dependent or speaker-trained automatic speech recognition technology is most popular. This simple technology includes several qualities that create its popularity and satisfy the requirements of high recognition accuracy and low implementation costs. This technology has a high degree of robustness combined with a low complexity of implementation. Moreover because the ASR is trained by users this solution allows a multilingual system without additional effort. It's why today and in the near future, this is the primary technology available in mobile phones and wireless devices. Despite its success, there is only a limited range of different applications that can be implemented using a speaker-dependent recognition engine. The most important disadvantage of this technology is that the users must go through a training phase before using the application. This training step, which allows the ASR to recognize all vocabulary items, is boring and time consuming. Speaker-trained systems are a technically proven solution for embedded wireless devices, but because of several application-specific issues and usability reasons, the speaker-independent ASR technology is preferable. That is why future visions cannot be realized with user-dependent technology[1].

2.3 Speaker-independent

The fundamental difference from speaker-dependent technology is that a speaker-independent speech-interaction doesn't require a training phase in order to work. This technology is able to recognize users' instructions when the speech recognition application operates. Speaker-independent ASR is a very interesting and flexible way to make mobile application. However, some technical obstacles must be solved before its wide-scale utilization. Indeed the multilingualism of users and the noise robustness must be managed. Moreover the development of ASR language must be more cost-efficient, but the speaker-independent ASR is feasible. Speaker-independent ASR systems are already available for PCs and network based ASR services. The next logical step is to include this technology in embedded devices to realize more polyvalent ASR applications[1].

2.4 Cost

A speech-based application is very expensive in terms of battery and memory consumption. Because of this, the use of memory is carefully optimized by setting severe limitations on the ASR technology. These restrictions are necessary because power and memory are limited resources in mobile devices. Because batteries are limited computing resources are available only for a short time. Due to this, it isn't possible to run "continuous listening" types of ASR applications or perform background computation like online background noise estimation. Mobile devices are destined for mass markets and the price must be minimized, so all new features or functions that increase the costs of a device need to be well justified. For example, in theory it's technically feasible to create country or language specific devices but the logistic and cost issues make this impossible in practice. To manage different languages a new ASR framework is needed, which costs a lot of money.

The speech recognition needs to be separated in two parts to be cost efficient: the implementation phase and the development phase of ASR systems. It's essential to have a technology for the speech recognition be compact and have a low implementation complexity (in term of memory and computational overhead) to significantly reduce the costs. Low complexity includes two different concepts; a "standard" implementation of ASR algorithms or ASR algorithms that are a set of methods providing the same recognition performance with smaller resource requirements. The evaluation of an ASR system is based on two complementary elements, the decoding speed and the memory consumption. If the memory usage increases the decoding process accelerates and vice-versa. Memory consumption is a greater problem than the decoding speed, which is why we need methods to reduce that consumption with an efficient decoding speed. Most of the memory consumption is used by acoustic models, but if the run-time memory consumption is minimized, memory occupation by acoustic models is decreased.

An other element is cost efficiency via multilingual ASR. The problem with language dependency is associated only with speaker-independent ASR technology. The development of speaker-independent systems is time-consuming and expensive; to reduce the costs it's important to have a flexible multilingual ASR framework where it is easy to add new languages with minimum efforts and produce an acceptable recognition rate. The multilinguality of ASR systems has a particular importance for wireless devices because the same systems are sold in many places with multiple languages. It is better to have a "global" ASR engine integrated into devices that is able to support the required languages[1].

3. SPOKEN INTERACTION ON MOBILES

There are different ways to interact with a mobile by speech. One method is to use keywords query but this solution is not discussed in this paper. Two other alternatives, explained below, are dictation and long queries. User interfaces with a speech-based interaction can provide an experience that is similar to human-to-human communication. With these technologies the transfers of information between human and machine are more natural and efficient. In addition traditional text entry methods do not allow for hands-free and eyes-free usage because the user is involved with the interface; for this speech-based systems are a better alternative. ASR is utilized for a huge variety of applications like voice-based personal computer control or customer service application like automated flight status inquiries. Due to low processing power and memory in mobile phones, until today implementations of speech interaction was limited to simple voice command functionalities like name or number dialling and application shortcuts. With the advancement of technology, ASR is expected to evolve to support new features

like text input via voice with better performance[3]. Response time and accuracy continue to improve which suggest that the usage of voice-activated queries and commands will significantly increase in coming few years[4].

3.1 Dictation

Dictation system allows users to dictate text to a mobile phone and then the speech recognition engine embedded in the phone converts the voice input into text. This solution has a lot of advantages but also a set of usability problems. The main purpose of this method is to facilitate text entry in mobile devices. Compared to the typing speed, the normal speech of humans is much faster. At 125-150 words per minute, dictation is a powerful and highly efficient utility if used for text input. But a large vocabulary is necessary to achieve satisfactory recognition accuracy. Thus, the most important challenge for an acceptable dictation system is to capture an extensive vocabulary spoken by different speakers in diverse acoustic conditions. Three kinds of interactions are required to complete tasks with speech-based text input, dictating, navigating to errors and error correction. When a user dictates a text, they spend 25 to 33 percent of the time on dictation and the rest of the time is spent detecting, navigating to and correcting errors because these systems are so predisposed to errors. Bad acoustic conditions increase the error rate, so it's important to have a robust and efficient error correction method.

Some Guidelines are needed to improve the design of applications using dictation on mobile devices. Firstly, it appears that users are more tolerant with errors when the dictated information is personal rather than information for an official communication use. High recognition accuracy is required for dictation features to be used as a text input option for communication. With the dictation system, it's important to have audible feedback (Text-to-Speech) of the result because it is useful in hands-free and eyes-free situations. It appears that the speech input method is most effective when combined with other modalities. A strong need for dictation features exists because this feature seems to be very useful if it is easily accessible[3].

3.2 Long queries

When a user uses a speech interaction system it is natural to speak in full sentences. Indeed researches demonstrates that users prefer natural expression of queries over keywords. Each year the web search engine query length increases and the full sentences recognition becomes faster and more accurate. Nowadays more and more systems accept quasi-natural language interfaces. But the difficulty for natural speech recognition is the users' colloquial language. The system must be tolerant to variations in syntax of request and "sloppy commands". A Web search engines supports a variety of formulations for certain kinds of questions. All answers come from a database or a pre-defined program. To solve the problem of "sloppy commands", some tools have been developed so users can express commands with greater flexibility. Another problem with normally speech systems is that the searcher doesn't have the vocabulary to search what is needed. The combination of online colloquial databases and good search engines reduce the vocabulary problem. The future development of voice-based input is a dialogue: a give and take[4].

4. PROBLEMS

The speech-based interaction involves various problems. The main problem is the noise robustness. Speech-based applications must be able to understand what users say even if the environment is noisy. Therefore, when the algorithm is designed noise robustness must be seriously addressed because even in controlled noise conditions the recognition accuracy tends to degrade. In mobile devices, the signal-to-noise ratio varies between + 30 and -10 decibels and the background noise can range from stationary to highly non-stationary noise signals. And it isn't a guarantee that two consecutive utterances would be spoken in the same noise conditions because the environment can suddenly change. As mentioned previously, the battery in mobile devices is limited, so it is not possible to continuously track noise characteristics with an online estimation technique[1]. The complexity of capturing large vocabulary spoken by diverse speakers in various acoustic conditions prevents the successful deployment of ASR[3].

Besides the background noise there are other elements that make the speech recognition difficult. Indeed, the ASR interface must be adapted to bad conditions, speaker, and specific characteristics of language. Some elements interfere with the speech recognition like background noise, speaker specific voice quality, speaker specific pronunciation, and language specific characteristics like tonal features or specific accent. To have a high recognition rates, the ASR application must adapt to these conditions as well as possible.

Secondly the complexity of mobile environments hinders the performance in terms of speech recognition accuracy and error handling but the development of ASR technologies and their applications are progressing[3]. One drawback of ASR interaction is that speaking itself creates noise. When the user speaks aloud, their voice input can disturb people who are around them. To solve this problem an exciting solution could be a microphone that hears the user without people nearby hearing the conversation[4].

Another problem that has already been mentioned above is the vocabulary. Because most of the time people speak with a colloquial language. It's difficult for algorithms to match the users' words with terms used in the informative documents. But algorithms do continue to improve results for difficult queries[4]. A problem with data security may also appear, because during the speech recognition most of algorithms send the collected data via the Web for processing. This solution requires a secure data transmission to protect personal information.

5. ERROR HANDLING

Error handling is the correction of recognition errors by the user, which happen simply because of the imperfect recognition by the ASR. This repair is still a considerable problem. Despite significant improvement in recognition algorithms, errors are not completely eliminated. That's why in the design and the implementation of speech-based ASR systems speech recognitions errors remain a serious problem[5]. Different solutions must be considered for error handling. Methods have to be efficient and easy to use.

The most intuitive way to correct errors seems to be the

"re-speak the entire phrase" technique. If the ASR doesn't understand the user's query the user must repeat the entire sentence. Another solution could be to provide a multimodal interaction where users are allowed to select a mistake manually and "re-speak" only this target word to replace the error. This method would prevent the introduction of new errors into the query when the user repeats the full sentence.

A more traditional method is having the user select mistaken and type the letters with the keyboard[3].

Some previous researches says that a multimodal system could accelerate the correction of errors. Indeed a multimodal repair is more accurate and faster than a unimodal error correction by respeaking. Multimodal means the use of more than one modality like keyboard, mouse, speech, gesture and handwriting input. Some components must be included in this system, such as recognition components, components to capture user input, components to present the output to the user and modules to support integration. The use of multiple modalities helps to prevent repeated errors. For example a study demonstrated that an interface which supports simultaneous speech and pen input reduces the total time to complete the task and correct errors.

To repair errors two phases are needed, first the error must be detected and after that it can be corrected. To locate errors, the user can use pointing or voice commands. The most natural and efficient way is pointing but voice-selection of errors already exists in commercial dictation systems. When a user interacts with a speech-based multimodal interface the primary input is usually continuous speech. This first input is interpreted with an adequate ASR and after that a feedback is provided to the user; it can be visual or the action is executed. After the feedback, the user or the system decides if a recognition error has happened. When the recognition is accepted, no repair is necessary, but if an error is detected one or more repair interactions is needed until all errors are corrected. When errors are detected and located the user chooses an appropriate method to provide the correction input. Before recognizing the correction input, the repair context is updated. After that, the correlation step selects the recognition output from appropriate recognizers, and it optionally applies algorithms to increase the likelihood of successful correction. When the final hypothesis is selected, the system provides the new feedback to the user.

Two different techniques to make correction are presented in [5]. The first is the respeak method that is the preferred for user because it's like human-human dialogue. But the repetition of a query does not necessarily increase the probability, this being unlike a human-human interaction. This is because when a user repeats a query it doesn't necessarily eliminate the cause of recognition errors. Moreover when the user hyperarticulates the accuracy of recognition deteriorates. In fact, because the recognizer is trained to normally pronounced speech, the hyperarticulation repeats errors. It's better to use another modality to correct errors, for example spelling verbally or handwriting incorrect words.

The second presented method allows users to modify items by using pen or mouse gestures. Some previous researches show that pen gestures are intuitive and efficient for such command control tasks. With gestures users can deleting, positioning the cursor or selecting input items. All actions can be done at different input levels, phrases, words or characters within a word. This kind of solution is attractive for applications that include a graphic user interface and where a pen or a mouse is available.

Unfortunately for the moment voice-based systems to create text are less effective than traditional methods. It turns out that the potential gains in productivity are lost during errors correction. Users can create text more efficiently with dictation systems only after prolonged use and learning time[5].

6. CONCLUSION

This paper addressed the different aspect of spoken interaction on mobile devices. It has highlighted the fact that the speech-based interaction is promising way to improve mobile devices, especially mobile phones. This technology has made significant progress in the last few years. There are different methods to create speech-based systems. Firstly there exist two different technologies used for speech recognition, the user-dependent or the user-independent methods. Then it discussed how speech interaction can be divided into three different techniques, keywords queries, dictation and natural expression of long queries. Existing systems must be improved to provide a more robust and more natural way to interact with the human voice. The ideal system would be able to keep only key items and throw away all the background noise to create a real conversation with the user. To conclude, spoken interaction is a promising technique that has its strengths and weaknesses. Strengths are: the interaction in a manner similar to the non-computing lives, the ease of use and the cool factor of interaction. Unfortunately there are also weaknesses like: the cost in terms of battery and memory consumption, the implementation complexity and the robustness of noise. Techniques of speech recognition must be improved before a large commercialization of application based on speech interaction.

7. REFERENCES

[1] Olli Viikki. "ASR in portable wireless devices." *ASRU '01. IEEE Workshop*, p.96-102, 2001.

[2] I. Varga, S. Aalburg, B. Andrassy, S. Astrov, J. G. Bauer, C. Beaugeant, C. Geissler, and H. Hoge "ASR in Mobile Phones - An Industrial Approach." *IEEE Transactions on speech and audio processing*, vol.10, p.562-569, 2002.

[3] S. Basapur1, S. Xu1, M. Ahlenius1, and Y. Seok Lee "User Expectations from Dictation on Mobile Device.s" *J. Jacko (Ed.): HCI 2007, Part II*, p.217-225.

[4] Marti A. Hearst "Towards "Natural" Interactions in Search User Interfaces." http://people.ischool.berkeley.edu/ hearst/papers/cacm11.pdf

[5] B. Suhm, B. Myers, A. Waibel "Multimodal Error Correction for Speech User Interfaces." *ACM TOCHI*, Vol.8 Issue 1, 2001.