Speech-based Dictionary Application

Thanyaphorn Lerlerdthaiyanupap

University of Tampere
Department of Computer Sciences
Computer Science
M.Sc. thesis
Supervisor: Markku Turunen
June 2008

University of Tampere
Department of Computer Sciences
Computer Science / Software Development
Thanyaphorn Lerlerdthaiyanupap: Speech-based dictionary application
M.Sc. thesis, 54 pages, 11 index and appendix pages
June 2008

For over two decades, speech has been used to interact with computers. During this same period, a number of speech applications have been implemented in many research and commercial projects for various purposes. One educational purpose has been to give a more natural way to communicate between learners and computer-based learning materials. In the field of language learning, previous research on using speech interfaces has mostly been concentrated on the development of speech-based translation systems providing bilingual or multilingual translations in the form of spoken dialogue phrases or sentences within limited domains and the development of computer-aided language learning systems. However, speech-based dictionary applications in the sense of giving a word definition have not much been studied. This is despite the fact that dictionaries are apparently beneficial for supporting language learning through providing various resources such as a list of alphabetical words with their meanings which can also be described in other languages, grammars, pronunciations, and example phrases or sentences. Since speech technologies have increasingly been developed and successfully deployed in many applications, this motivates the study of applying the same technologies to dictionaries. This thesis studies an approach to constructing a speech-based dictionary application connected to the speech technologies currently available. The survey of previous and current research works is presented. The gathered speech technologies are described and a design of the application using some available development environments and languages is proposed. Also, the findings on the implementation of a speech-based dictionary prototype system are explained.

Key words and terms: Dictionary, speech application, speech technology.

Contents

1. Introduction

The possibility of using spoken language to interact with computers has generated wide interest in spoken language technology. Many universities and research laboratories have put their efforts to a development of the spoken language technology. For example, Verbmobil project [Wahlster, 2000], GALAXY [SLS Group, 2005], and Universal Speech Interface [Rosenfeld et al., 2007]. In addition, spoken language technology has offered a number of commercial benefits such as reduced expenses, improved services and new markets access. Because of these commercial benefits, many large companies, such as IBM, Philips, and Microsoft have been motivated to work on numerous research and development specializing in spoken language technology. Embedded ViaVoice, SpeechMagic, and Windows Speech Recognition exemplify their commercial spoken language technology products. The spoken language technology has constantly been growing and it has also become an attractive topic for researchers in universities and companies. Consequently, a large number of speech-based applications have been constructed and readily used.

The use of speech as an interface element has been applied to various applications for more than twenty years since it provides some benefits which are suitable to those applications. For example, speech is obviously the most natural way to communicate with computers. Speech can also be a very efficient method of communication to solve interactive problem tasks. In addition, when a multimodal interface is operated, speech is capable of bringing more bandwidth to the interaction. Furthermore, speech can be used to perform some tasks better than other modalities, a direct manipulation for instance. The use of speech in applications is dependent on the purpose for which the application is constructed. Some applications use speech because it may provide the only possible modality, the most efficient modality or the most preferred modality. Moreover, speech can be used as a supportive, alternative or substitutive modality [Turunen and Hakulinen, 2005, pp. 1].

In practice, speech applications can be constructed from different perspectives for various kinds of applications [Turunen and Hakulinen, 2005, pp. 2]. Examples have included telephony applications to retrieve information services such as bus timetable services. Desktop applications are another area of speech applications aimed at dictating or controlling existing graphical applications, for example, word processors. Multilingual speech applications are mostly used for translation systems. Some other applications involve multimodal applications, information kiosks using speech and haptics modalities for instance. A further type of speech applications is pervasive speech applications. Turunen and Hakulinen give one example of these pervasive applications, namely, the MIT Virtual Room project created by Coen (1998 cited Turunen and Hakulinen, 2005, pp. 2). Speech applications can also be deployed in the educational sector [McTear,

2004, pp. 22]. Educational applications use speech interfaces to facilitate natural communication between the user and computer-based learning applications. One interesting example involves the use of speech technology in language education. Ehsani and Knodt [1998] proposed in their research paper that speech technology had the potential to be deployed in computer-aided language learning. A large example was the E-Language Learning System (ELLS), a joint governmental research and development project between the U.S. Department of Education and the Chinese Ministry of Education. ELLS was a web-based language learning application which used voice recognition and other integrated approaches to provide second language instruction to students and educators in the United States and China [ELLS Technical Work Group, 2002]. In this project, speech technologies were variously deployed in assisting language learning [Zhao, 2002]. For example, speech recognition technologies used to provide dialogues to interact with students and Text-to-Speech (TTS) technologies used to help learning pronunciation through vocabulary flashcards. In ELLS, a multimedia dictionary was another supporting tool proposed to assist learning vocabulary. However, the speech technologies were not applied to the dictionary.

A dictionary serves as a potential supporting tool in language learning. Currently, several formats of dictionaries exist with different usage. For example, paper dictionaries, electronic dictionaries, internet dictionaries and dictionary software. However, all of those dictionaries require other modalities to look up a word definition, e.g. hands, keyboard, mouse and stylus rather than speech modality. Although computer-assisted language learning applications have widely integrated speech technology for improving language instruction and speech-enabled phrasebook lookup applications have already been created for electronic dictionaries, Personal Digital Assistants (PDAs) and mobile devices, dictionary lookup systems using speech have not much been studied or created for Personal Computers (PCs) or laptops.

Such a system can be defined as an application that uses speech as a major modality to look up a word definition and to represent a result of the word definition. In other words, the system receives speech input from a user and then gives a definition result in the form of speech output. A user of the system is defined to be the average people who have normal speaking and listening skills. The system is capable of searching and presenting requested information for the average user. It might not be used by everyone especially the extreme populations such as the elderly and non-natives with strong accents.

This thesis proposes the idea of developing such a speech-based dictionary application for a desktop and laptop computer. The language meant to use for the application is only English. In other words, there will not be a definition or translation in any other languages produced by the system. Through using speech, it shows an

alternative approach to creating dictionary software for providers and to using a dictionary for users. The research questions in this thesis work are addressed as follows:

1. What previous researches have been done?
2. What speech technologies exist nowadays?
3. Which concrete designs for the application should be connected to existing technologies and previous researches?
4. What readily available development tools exists to make the application?

The thesis is divided into six main chapters. The first chapter describes the introduction. Then, chapter 2 shows a survey and review of research on speech-based dictionary systems. The third chapter presents a spoken dialogue technology in theory in which its understanding can be used to implement a speech-based dictionary application. This chapter also includes some examples of current speech technologies. In chapter 4, the principles including design guidelines of spoken dialogue system development are presented. Next in chapter 5, I put forward the idea of developing the speech-based dictionary application. Chapter 6 finally summarizes the findings and proposes the future development.

2. A survey and review of research on speech-based dictionary systems

The deployment of speech technology in language learning field has been extensively researched and developed in recent years. However, the research and the development have tended to focus the integration with speech technology on computer-aided language learning applications and translation systems rather than on dictionary applications. It would thus be of interest to explore how speech technology can be deployed to improve dictionary applications. In this paper, a speech-based dictionary system refers to an application that is created for a desktop and laptop computer and provides users with a word or phrase definition by using speech interface. To put it more simply, the application allows users to speak a word or phrase and then hear a definition. In this case, what are the methods and tools to successfully develop such the system? This chapter surveys and reviews research on speech-based dictionary systems. It is divided into three main parts. The first part reviews the literature relevant to the concepts of speech-based dictionary system. Then, the second part surveys and reviews commercial dictionary products integrated with speech technology. In the final part the institutional research is reviewed in chronological order. This review is aimed to find their methods, tools and outcomes and to point out the future research.

2.1. Literature relevant to the concepts of speech-based dictionary system

Back to 1986, Crystal [1986, pp. 79] presented his idea of using voice-activated terminal to access to electronic dictionaries. His proposal indicates that the ideal dictionary should contain an electronic database which is accessible via voice. It also shows that through using voice, the ideal dictionary should be able to provide the requested information (e.g. word meanings, thesaurus, and usage) either in visual or audible form as preferred. The concept of using voice to access a dictionary database in order to retrieve the required information, such as word meanings, was introduced and was hoped to encourage researchers and developers to make it materialized. However, Crystal's proposal had been surprisingly impractical, i.e. remaining a dream, assumed by de Schryver [2003, pp. 173]. He, in fact, considered Crystal's dream to be unrealized without any proof. Also, he mentioned and graded the degree of realization of some other lexicographer's dreams towards electronic dictionary revolving around audio [de Schryver, 2003, pp. 167]. The results show that representing information (e.g. definitions and example sentences with sandhi, sentence stress, or intonation) in audible form was unfulfilled except the pronunciation of lemma signs and sound connected with certain words. Furthermore, through speech synthesis, presenting example sentences was not realized but instead a representation of the whole definitions for blind and visually-impaired people. In addition, automatic speech recognition lingware failed to achieve evaluating and grading a foreign speech input but recording and comparing tools succeeded in doing so. Unlike Crystal's dream, a dream about a communication between

users and databases with natural language and true dialogues was more or less feasible [de Schryver, 2003, pp. 174].

According to the research article by de Schryver, no investigation was exactly carried out to show why those dreams were unrealized. In other words, the outcomes were based on previous research and the circumstances at that time. The author somehow considered that speech technology at the period he was writing was not capable of enhancing audio features of dictionaries. Still, I remain my research question how current speech technology can be used to improve dictionary applications. As well as, Simpson [2003] pointed out a need for future development in a way that uses sound for indicating the denotation of the word. In any case, this interests the continuation of the exploration of developing a dictionary on the basis of Crystal's proposal, since speech technology is greatly advanced nowadays.

2.2. Commercial dictionary products integrated with speech technology

The following sections present some selected commercial dictionary products utilizing speech technology.

2.2.1. Franklin handheld electronic dictionaries

Handheld English dictionaries are one of the dictionary categories which have been developed and produced by Franklin Electronic Publishers, Inc. [Franklin, 2007]. Some examples of these devices are (Speaking) Merriam-Webster's Collegiate Dictionary 11th Edition, Merriam-Webster Speaking Dictionary & Thesaurus, and Speaking Language Master. The devices offer English word definitions with the ability to speak aloud a searched word and its definitions by Franklins ClariSpeech technology. However, these devices require typing text as an input method to look up word definitions.

2.2.2. Ectaco handheld electronic dictionaries

Electronic handheld dictionaries have been developed and produced since 1990 by the Ectaco Company [Ectaco, 2007]. Several generations of the Ectaco handheld electronic dictionaries have been integrated with advanced speech technologies. Ectaco even built its own Text-to-Speech (TTS) technology and speech recognition engine used specifically in the speech-to-speech translation part. Some examples of Ectaco electronic dictionaries are the 800 series, the TL-2 series, and the X5/X8 series. These devices let users say a phrase limited to given topics and then they provide a spoken translation in a target language. In addition, they offer Ectaco speech recognition system for foreign language studies helping users to improve their pronunciation. The devices include a look-up function that can be performed by inputting text with keyboard or on-screen keyboard using a stylus. Only the X8 series include a voice input of English words function that allows users to spell a letter to obtain the required word definitions or

translations. However, the devices provide speech output in order to represent accurate pronunciation of definitions or translations of a word.

2.2.3. LingvoSoft software products

The LingvoSoft Company, the software division of the Ectaco Company, has developed and produced a large number of language learning, dictionary, translation and localization software products for all major platforms since 1990 [LingvoSoft, 2007]. Concerning the speech-based dictionary system mentioned above, the LingvoSoft dictionaries software for Windows is mainly reviewed. The software provides bidirectional word translation for multiple languages with utilizing text-to-speech technologies. LingvoSoft has uniquely developed its own Text-to-Speech technologies and speech recognition system, i.e. Lingvobit which by now only recognizes the phrases from the predefined list. Therefore, all LingvoSoft dictionaries software is equipped with the Text-to-Speech function but the Lingvobit speech recognition system is particularly utilized in translator software. With the LingvoSoft dictionaries software, users can obtain word translations or explanations by entering, or copying and pasting text and hear them by clicking Text-to-Speech function. This implies that voice input function looking up word definitions or translations is not integrated into LingvoSoft dictionaries software.

2.2.4. Talkman playstation portable

Talkman is the voice-activated multilingual translation software developed by Sony Computer Entertainment for the Sony PlayStation Portable video game console. Its first release offers translations, mostly slang and helpful travel phrases, between all four languages, i.e. Japanese, Chinese Mandarin, Korean, and English. Talkman also includes game mode to help players learning and practicing languages. The second release of Talkman is Talkman Euro which provides translations in six languages, that is, English, Italian, Spanish, German, French, and Japanese (or Traditional Chinese) [Wikipedia, 2007]. According to the reviews of Talkman by Metacritic [2006], it appears to be that Talkman receives an average or mixed score. The pros of Talkman involve its novelty and the ability to assist the users as a translator/tutor with interactive talking phrasebook and games functionality. However, the big cons of Talkman are its vocal translation technology and loading time. The former is concerned with the problem of delivering the right phrases in response to the users' input, while the latter is about spending long time to switch between screens. In addition, Talkman does not include dictionary function either.

To sum up briefly, it is noticeable that the commercial dictionary products presented above are now capable of speaking a headword and its definitions with Text-to-Speech technologies developed for their own company. Conversely, the products cannot be

operated to look up word definitions by speaking a word, i.e. voice input. Only the Ectaco X8 series electronic dictionaries can accept voice input to search for word definitions but by spelling a letter not speaking a word, whereas Talkman translator accepts voice input to translate phrases into other foreign languages not for meanings lookup. Therefore, it is quite clear that the issue of using voice input by saying a word to look up word definitions opens an area for further development.

2.3. Institutional research

Institutional research relevant to the speech-based dictionary system is presented in the following sections.

2.3.1. The intelligent retrieval system with speech queries for very large Chinese dictionaries

The intelligent retrieval system with speech queries for very large Chinese dictionaries was a research work by Lin et al. [1997] presenting the intelligent retrieval techniques to retrieve a Chinese word entry from very large Chinese dictionaries using speech queries. To put it clearly, the Chinese word entry mentioned in the paper is a headword and its lexical information such as synonyms, antonyms, word explanations and example sentences. There are two main techniques proposed in the paper. First, it is the integration of Mandarin speech recognition technologies and syllable-based Chinese information retrieval. Second, it is the relevance feedback techniques. The following briefly explains all processes of the retrieval system with the first technique. The Mandarin speech recognition component first recognizes user's speech input based on a syllable level approach and then transcribes it into the most possible syllable string. Next, the syllable-based information retrieval component compares the recognized syllable string to all word entries in a dictionary on the syllable level. The word entries with the syllabic structures of the speech input are finally represented to users from their highest to lowest relevance scores. The second technique was proposed to make the search more powerful, i.e. users can describe a general idea of a desired word. It shows methods to improve the successive queries and update the syllable-based language model for the speech recognition component. That is, by the relevance feedback component, the lexical information of the word entries previously retrieved is automatically or manually extracted into syllabic structure. Then, the syllabic extracted information either is combined with the original query or updates the syllable-based language model for the speech recognition component. The achievement of implementing the retrieval system with proposed techniques was reported. Also, a screen capture of the example retrieval result was illustrated. This indicates the system displays the retrieval results in textual form on screen. In addition, the authors suggested applying the relevance feedback techniques to other textual databases, for instance, a database-specific thesaurus.

In conclusion, this research work succeeded in using speech queries to retrieve Chinese word entries from a very large Chinese dictionary with the proposed techniques. However, only speech input was concerned for solving their problem. Besides, a syllable level approach is used as a basis of all processes. This implies that either speech input (i.e. a syllable, a whole word, or a sentence) the system would then perform a task based on syllable level. Regarding the speech-based dictionary system, a word level approach would be mainly considered to solve my research question. That is users are supposed to know and say exactly a word or phrase to retrieve only its definitions. Nevertheless, those proposed techniques can actually be applied to such the speech-based dictionary system but this issue is beyond the scope of my research. This could be a future development though.

2.3.2. The FreeSpeech project

The FreeSpeech project created by Chesnut [2003] was targeted to implement a voice-controlled fat client application for Pocket PC. The project utilized the combination of speech recognition, natural language processing, and wireless-data technologies. The proposed main ideas can be explained as follows. Firstly, users record their voice input to the application on a Pocket PC. Secondly, the application transmits the recorded input to a web service. Thirdly, the web service processes speech recognition, i.e. it performs an operation to transcribe voice into text with either a specified grammar or dictation approach, or both. Finally, the web service returns the transcribed text to be displayed on the pocket PC. In the project the following development tools were mentioned: CF .NET (Microsoft .NET Compact Framework) Voice Recorder, Web Service Enhancements (WSE) Direct Internet Message Encapsulation (DIME) Web Service, and the Microsoft Speech API (SAPI) Voice Recognition. The application was designed, implemented, and tested based on different user scenarios. Also, the tabular test results and the demonstration of the application were shown. According to the test results, the author claimed that for entering data, the application works much faster with voice input than with a stylus. That implies voice input can solve a problem well on quick access to the required data among a big set of data. In addition, a further development on the Text-to-Speech issue was suggested too.

To summarize, the FreeSpeech project was an application for Pocket PC allowing users to speak freely. This indicates a better performance for entering data with voice input. The project also clearly shows the main ideas, methods, development tools, outcomes, and further development. These findings could be applied to develop the speech-based dictionary application. That is, the speech-based dictionary application could be implemented as a client-server or standalone application. Moreover, the retrieval and presentation of word definitions processes should be undeniably added.

2.3.3. Web services and speech-based applications around VoiceXML

An approach to using VoiceXML applications cooperated with speech web services was presented by Rouillard [2007]. The proposed approach expands the concepts of the FreeSpeech project on speech-based and multimodal applications. In other words, it enables users to speak to any speech-based or multimodal applications without restraint in order to complete their desired tasks. The key success of the approach was to deploy a large vocabulary voice recognition system to create a speech web service. In this case, the speech web service can independently process the transcription of text, a so-called speech dictation system. Consequently, VioceXML applications can handle with unknown words. Based on the proposed approach, a few scenarios were presented to figure out the solutions. One of them mentioned about requesting a word definition and three possible ways to obtain it. According to that scenario, two solutions were presented, i.e. asynchronous and synchronous solution. Asynchronous solution means that VoiceXML applications can cooperate with speech web services in order to send and receive the requested information by URL, while synchronous solution means sending and receiving it by Direct Internet Message Encapsulation (DIME) attachment. The project used the following development tools: VoiceXML platform, .NET Framework, Visual Studio (C# language and Web Services Enhancement), BPEL (Business Process Management Language), and the Oracle BPEL Process manager. A prototype was successfully implemented based on the presented solutions. This confirms the technical feasibility. Still, a need of the robustness improvement was suggested for a further research.

To conclude, the proposed approach expands a domain of user utterance so users are free to speak to any speech-based and multimodal applications. The scenario about requesting a word definition is definitely related to my research topic. The prototype shows the achievement of constructing the speech-based application relevant to a dictionary. This implies that there eventually exists the speech-based dictionary application. Indeed, the findings give very helpful ideas to create such a speech-based dictionary application. Regarding my research topic, grammar-based speech recognition would be considered to increase a speed of user's task, although it limits user utterance. Besides, free development tools would be used to create a prototype based on the principle of this research work.

2.3.4. The dictionary lookup system for mobile devices using spelling recognition

The dictionary lookup system for mobile devices using spelling recognition was a relevant research system invented by Azulai et al. [2007]. It is a patent system claiming the invention of general speech recognition systems and specific methods and systems for querying an electronic dictionary using spelled letter input. Their patent methods can shortly be described as follows. The first method is to accept a speech input. The speech

input means a sequence of spelled letters pronounced by the users. Then, the spelled letter input is analyzed to produce the proximate one or more sequences of spelled letters, that is, a plurality of recognized words. Next, the plurality of recognized words queries an electronic dictionary to retrieve its corresponding dictionary entries. Finally, a list of responsive dictionary entries results is displayed to the users. In other words, a list of possible recognized words along with the corresponding dictionary entries, i.e. a definition or a translation of a word or phrase is presented to users in textual and spoken form. However, the system also supports a multimodal user interface, i.e. using spelling recognition with keypad functions and/or voice commands in order to run the application smoothly. The mobile devices supported by the system refer to mobile phones, portable computers and personal digital assistants (PDAs) but desktop computers could be applied too.

In brief, the principles of the patent system indeed show the efficient approaches to develop dictionary applications, which are definitely suitable especially for travelers, language learners, and disabled people. Moreover, the authors suggested that the same methods can be used in a variety of additional applications, for instance, directory assistance services and name dialing applications [Azulai et al, 2007, pp. 16]. Regarding the research paper, the invention of the system is in progress. This implies that in the near future users can possibly consult a dictionary by speaking, in fact, spelling. Although the methods and systems obviously show spelling-based dictionary lookup in mobile devices, the combination of word-based and spelling-based method can be a further research area. Concerning my research topic, speaking a word or phrase will be a main input method. Thus, the spelling-based input method will not be considered in my research.

Table 1 shows a summary of the reviewed research on speech-based dictionary systems. It first starts with the idea of querying and retrieving data stored in a dictionary database via voice proposed by Crystal. Seventeen years later, de Schryver claimed that the access to electronic dictionaries and the presentation of information via voice were unrealized, although speech technology existed. However, up-to-date commercial dictionary products are capable of presenting word definitions in audible form with Text-to-Speech technologies. Still, in most cases users cannot access to word definitions by voice input but this proves Crystal's idea could be realized little by little. Regarding institutional research, there were more attempts to use speech input to retrieve information from a dictionary. As a result, there has already been one existing speech-based application for a dictionary implemented by Rouillard. Also, there is another spelling-based system for a dictionary proposed by Azulai et al. but still its invention has been in progress. It can be assumed from the review that the representation of word definitions can be done by using Text-to-Speech technology. For looking up word definitions with speech input, a speech recognition system plays an important role. To

offer a variety of user speech input, a large vocabulary speech recognition system with a dictation mode should be employed to a dictionary application, while a spelling speech recognition system should be used to perform an efficient search function. The former easily makes an error-prone, while the latter will not be considered in my research. Hence, a speech recognition system with grammar-based approach will be used in my research.

Research on Speech-based Dictionary Systems	**Main Ideas**	**Input Methods for Look-up Function**	**Output Methods for Presenting Definitions**	**Research Outcomes**
Research article by Crystal [1986]	Present the idea of using voice-activated terminal to access to electronic dictionaries	Speaking	Display information in sound, on screen, or in print	No outcome. Only present the proposal
Research article by de Schryver [2003]	Show lexicographers' dreams towards electronic dictionary revolving around audio and give the degree of realization of those dreams	-	-	Claim that access to electronic dictionaries and presentation of information via voice were unrealized
1. (Speaking) Merriam-Webster's Collegiate Dictionary 11th Edition 2. Merriam-Webster Speaking Dictionary &	Handheld English dictionaries	Typing	1. Display definitions on screen 2. Speaking the definitions with Franklin Text-to-Speech technology	Already available on stores

Thesaurus 3. Speaking Language Master [Franklin, 2007]				
Ectaco 800 series [Ectaco, 2007]	Handheld talking bilingual dictionaries expandable to include over 50 language combinations	1. Typing 2. Touching screen with a stylus when using on-screen keyboard 3. Speaking a phrase with Ectaco speech recognition system available in Phrasebook feature	1. Display definitions or translations on screen 2. Speaking the definitions or translations with Ectaco Text-to-Speech technology 3. Real native-speaker voices in Phrasebook feature	Already available on stores
Ectaco TL-2 series [Ectaco, 2007]	Handheld talking bilingual dictionaries expandable to include over 50 language combinations	1. Touching screen with a stylus when using on-screen keyboard 2. Speaking a phrase with Ectaco speech recognition system available in Audio Phrasebook feature	1. Display definitions or translations on screen 2. Speaking the definitions or translations with Ectaco Text-to-Speech technology 3. Real native-speaker voices in Audio Phrasebook feature	Already available on stores
Ectaco X8 series [Ectaco,	Handheld talking dictionaries	1. Typing 2. Touching	1. Display definitions or	Already available on

2007]		screen with a stylus when using on-screen keyboard 3. Voice input of English words by spelling letters 4. Speaking a phrase with Ectaco speech recognition system available in Voice Phrasebook feature	translations on screen 2. Speaking the definitions or translations with Ectaco Text-to-Speech technology 3. Real native-speaker voices in Voice Phrasebook feature	stores
LingvoSoft dictionaries software for Windows [LingvoSoft, 2007]	Dictionaries software for Windows platform	1. Entering text from keyboard or on-screen keyboard 2. Copying and pasting with a mouse	1. Display definitions or translations on screen 2. Speaking the definitions or translations with LingvoSoft Text-to-Speech technology	Already available on stores
Talkman [Metacritic, 2006]	Translator/tutor application for Sony PlayStation Portable	Speaking a phrase then the device will show a selection of the responsive phrases. After that, pressing a button to select	Speaking the selected phrase in other foreign languages	Already available on stores

		the desired phrase. (this is for translation)		
Research article by Lin et al. [1997]	Present the following techniques to retrieve Chinese word entries from very large Chinese dictionaries with speech queries. 1. Integration of Mandarin speech recognition technologies and syllable-based Chinese information retrieval 2. Relevance feedback techniques	1. Speaking syllable(s) or whole word 2. Describing a general concept of a desired word	Display all retrieved Chinese word entries relevant to a query input on screen	The system was successfully implemented with the proposed techniques
Research article by Chesnut [2003]	Show how to create a voice-controlled fat client application for Pocket PC	Speaking without constraint to receive what was spoken in text form but not to look up word definitions	Display what users said on screen	The demo application was successfully created
Research article by Rouillard [2007]	Propose the approach to using the cooperation between VoiceXML	Speaking without constraint	1. Speaking the definitions with speech synthesis 2. Sending an email	A prototype based on the presented solutions was successfully implemented

	applications and speech web services		containing definitions to a user	
Research article by Azulai et al. [2007]	The patent system claiming methods for dictionary lookup for mobile devices using spelling recognition	Spelling letters including keypad functions and/or voice commands	1. Present the definitions on display of the mobile device 2. Speaking the definitions with text-to-speech generator	Regarding the research paper, the invention is in progress

Table 1. A summary of the reviewed research on speech-based dictionary systems.

3. Spoken dialogue technology

People naturally use speech as a method of interaction but when it comes to communicate with computers, some other methods need to be used instead, such as using a keyboard or mouse. In fact, speech is the most natural way to communicate with computers. Therefore, spoken dialogue technology is required. Spoken dialogue technology involves developing applications that enable people to interact with computers using spoken language. In order to develop spoken dialogue systems, the following components are at least needed to cooperate for the systems: speech recognition, Text-to-Speech synthesis, and dialogue manager. Thus, these tree main components will be required regarding my research topic. It is then important for developers to understand the features and capabilities of each component. Based on that knowledge, the developers can determine what will be suitable for designing and implementing such spoken dialogue systems. In this chapter, I present a theoretical knowledge of the three main components based on the book written by McTear [2004, pp. 79-126] and the course material provided by Turunen and Hakulinen [2005, pp. 3-4, 7-8]. Also, some examples of the current speech technologies and their properties are shown.

3.1. Speech recognition

The most basic of speech recognition technologies is automatic speech recognition (ASR) which is the capability to automatically recognize human speech based on a word-by-word [Cox et al., 2000, pp. 1319]. The speech recognition component is aimed at capturing user's spoken input and transforming it into the most probable word sequence using the given acoustic (feature) and language (structural) models. The following sections explain speech recognition processes, the properties of speech recognition systems, and the performance of speech recognition systems respectively.

3.1.1. Speech recognition processes

1. *Signal processing*. A speech recognition system first starts with extracting the relevant set of features from the acoustic signal, i.e. a user spoken input.
2. *Phoneme and word identification*. This process involves classifying the extracted set of features into phonemes using an acoustic model and merging sequences of phonemes into words using a language model.
3. *Acoustic model*. The extracted set of features are compared to an acoustic model (phonemic level) which defines a sequence of phonemes of each word in advance. By statistical means, the acoustic model can be created from a large corpus of speech [Hakulinen, 2006, pp. 17]. In addition, most of current speech recognition systems use Hidden Markov Models to do pattern matching since they provide methods to create variable patterns of speech.

4. *Language model*. The extracted set of features are compared to a language model on lexical and syntactic levels too. A language model provides a knowledge base to specify permissible sequences of words and is aimed at cooperating with the acoustic model. Context-free grammars and N-gram language models are the two common use language models.
    - Context-free grammars are used to define all possible words and phrases of a user input. These grammars are suitable for spoken dialogue systems with limited domains.
    - N-gram language models are used to give a possible word in a given context based on statistical information on word sequences. They are generally used in large vocabulary applications, for example, to support a dictation method.

3.1.2. Properties of speech recognition systems

The properties of speech recognition systems play an important role for developers. The understanding of those properties helps the developers to decide what type of speech recognizers is suitable for implementing their speech-based applications. From developers' perspective on speech-based application, the characteristics of speech recognition systems which should be considered are summarized in Table 2 [Turunen and Hakulinen, 2005, pp. 3].

Vocabulary and language			
Vocabulary size:	small	middle-size	large
Grammar (LM):	phrases	Context Free Grammar (CFG)	N-gram
Extensibility:	fixed	changeable	dynamic
Communication style			
Speaker:	dependent	adaptive	independent
Speaking style:	discrete	continuous	spontaneous
Overlap:	half-duplex	barge-in	full-duplex
Usage conditions			
Environment:	clean	normal	hostile
Channel quality:	high-quality	normal-quality	low-quality

Table 2. The main characteristics of speech recognition systems.

1. *Vocabulary size and recognition grammars.* The vocabulary size can be ranged from small to large depending on types of applications. A small vocabulary size is around 10-200 words, while a medium size is about thousands of words. Dictation systems, for instance, need very large vocabularies about more than

200,000 words. The recognition grammars are language models and they can be constructed to be fixed, changeable, or dynamic.

2. *Communication style*. The communication style indicates speaker dependency of speech recognition systems. Users need to train a speaker-dependent system in order to recognize their own speech patterns. A speaker-independent system can be used by anyone without training, while an adaptive system can be used with or without training. Speech recognition systems can accept three modes of speaking style. First is a discrete mode requiring a user to pause shortly between each word. Second is a continuous mode which the user can talk more naturally. Last is a spontaneous mode which in practice makes speech recognition systems not robust. The communication style also involves the concurrency of input and output. Only a system or user can speak at any time in half-duplex communication but both of them can speak at the same time in full-duplex. However, with the barge-in capability, users are allowed to interrupt system outputs.
3. *Usage conditions*. The usage conditions involve an environment and a channel that speech recognition systems can be used. The environment can be ranged from clean to hostile, while the channel can be ranged from high-quality to low-quality. Close-talk microphones at clean places can give high-quality, while using a mobile phone at noisy or public places can give low-quality to speech recognition systems.

Table 3 shows some examples of the features of commercial and research speech recognition systems which are considered based on the knowledge presented above. They also indicate most current speech recognition technologies.

Speech recognition systems	**Embedded ViaVoice** [IBM, 2007]	**OpenSpeech Recognizer** [Nuance, 2007a]	**Windows Speech Recognition in Windows Vista** [Microsoft, 2007]	**Sphinx 4** [CMU, 2004]
Vocabulary and language				
Vocabulary size:	Middle-size (500 words) to large (> 200,000 words) depending on	Large	Normal to large	Small size (100 words) up to large size (64,000 words)

	selected software packages (standard, advanced, enterprise)			
Grammar (LM):	Finite state grammars and Statistical language models	Statistical language models (SLMs) and statistical semantic models (SSMs)	Context Free Grammar (CFG)	Phrases, Context Free Grammar (CFG), and N-gram supported
Extensibility:	Dynamic	Dynamic	Fixed to changeable	Changeable
Communication style				
Speaker:	Independent	Independent	Adaptive	Dependent and independent supported
Speaking style:	Discrete and continuous	Continuous	Continuous	Discrete and continuous
Overlap:	- (no information)	Barge-in	Barge-in	- (no information)
Usage conditions				
Environment:	Clean to hostile	Normal to hostile	Clean to normal	- (no information)
Channel quality:	High-quality to low-quality	Normal to low-quality	High-quality to normal	- (no information)

Table 3. The summary features of commercial and research speech recognition systems.

3.1.3. Performance of speech recognition systems

The performance of speech recognition systems can be measured from their accuracy and speed. Accuracy is measured with the word error rate (WER), whereas speed is measured with the real time factor. The WER is calculated as the percentage of

recognition errors and correct words of the words spoken. The WER formula is shown below [Turunen and Hakulinen, 2005, pp. 3].

WER = (deleted + substituted + added words) / words in original message * 100 (3.1)

To interpret the WER of a system, it is also important to consider the difficulty of the recognition task, which involves a speaker (e.g. giving dialect or non-native speech), noise environment, speaking style, and application domains.

3.2. Text-to-Speech synthesis

The speech synthesis or TTS (Text-to-Speech) component basically generates a spoken output from text. The following shows Text-to-Speech processes in brief.

1. The received text is interpreted and normalized to plain text. This process is essential for data such as numbers, abbreviations, mathematical symbols, etc. Meanwhile, the plain text is analyzed to put the prosodic information, i.e. volume, speed, pitch variations, and pauses in order to make speech output sound naturally.
2. The plain text with prosodic information is converted into phonemes.
3. The transcribed phonemes are converted into a speech waveform by speech generation module. This process can be done by three approaches, i.e. concatenative synthesis, formant synthesis, and articulatory synthesis.
    - *Concatenative synthesis*. The concatenative synthesis is based on the concatenation (or stringing together) of segments of recorded speech, which are stored in a speech database. There are various methods managing a combination of the units together.
    - *Formant synthesis.* The formant synthesis is concerned with controlling a production of audio signals, for instance, controlling a parameter of pitch.
    - *Articulatory synthesis*. The articulatory synthesis involves modeling the human vocal tract using complex computation.

Naturalness and intelligibility are the two qualities used to evaluate speech synthesis. Naturalness measures how natural a speech output is comparing to a human speech; the more natural the better is. Intelligibility measures how ease of understanding a speech output is.

The following table displays some examples of commercial and research Text-to-Speech systems. Their features indicate most current Text-to-Speech technologies.

Text-to-Speech Systems	**Features**
AT&T Natural Voices [AT&T, 2002]	- Produce a very realistic human-sounding synthetic speech system - Offers synthesis services in multiple voices and multiple languages for desktop applications - Language and voice-specific dictionaries customization - 2 versions (desktop and server version)
IBM Naxpres [IBM Research, 2007]	- Concatenative synthesis - Include expressive speech synthesizer offering a range of expressions - Synthesized speech available in several languages, such as Arabic, Mandarin Chinese, Cantonese, Chinese-Taiwanese, French, and German
RealSpeak Word [Nuance, 2007b]	- Produce accurate and superb quality speech output from a dictionary of words and idioms using ground-breaking approach - Use coding-by-synthesis architecture that does not need recording - Support very large vocabularies with very low memory - Support extension of vocabularies with little additional memory - Integrability with different platforms
Festival (research Text-to-Speech synthesis) [CSLU, 2004]	- Offer a general framework for building speech synthesis systems - Programmed through shell level, Java, and Emacs interface - Available in British English, American English, and Spanish - Freeware - HTS hidden Markov model based synthesis engine newly Added

Table 4. The summary features of commercial and research Text-to-Speech systems.

3.3. Dialogue manager

A dialogue manager is a center of a spoken dialogue system. This component basically involves managing speech input and output, interaction with users and external

databases, and controlling a dialogue flow. Dialogue initiative is a key aspect in managing dialogues, while dialogue control is a strategy that controls a dialogue flow.

The following sections present the dialogue initiative and the dialogue control in more details.

3.3.1. Dialogue initiative

In general, in spoken dialogue systems the dialogue initiative strategy can be categorized into three groups as follows.

1. *System-initiative dialogue*. A system-initiative dialogue involves directing a dialogue by a system. That is, the system asks question from a user in order to submit a query input to a database. In this way, grammars and a dialogue flow can be defined beforehand and they also help users to reach their target. Moreover, speech recognition can perform better tasks, i.e. producing more accurate results. However, the system-initiative dialogue strategy can produce a long dialogue which results in running a process slowly. This strategy is mostly suitable for information retrieval applications.
2. *User-initiative dialogue*. A user-initiative dialogue is concerned with controlling a dialogue by a user. The user acts as an active participant, i.e. knowing the tasks and asking questions, whereas the system acts as a responsive participant, i.e. answering questions. The strategy is advantageous to let a user have freedom to use the system. However, in the user-initiative dialogue strategy, the system requires the capabilities of natural language understanding with a diverse input. The user needs to know all tasks to complete too. This is hard to implement such a system in practice. An example of a user-initiative dialogue system is a command and control system.
3. *Mixed-initiative dialogue*. The mixed-initiative dialogue strategy lets both participants initiate a dialogue, such as asking questions, requesting confirmations, etc., at different times. A practical mixed-initiative dialogue strategy can be used in a way of beginning user-initiative strategy cooperated with error handling by system-initiative strategy. On the contrary, the mixed-initiative strategy usage can start with the system-initiative strategy and later take the user-initiative strategy when users are familiar with the system. A construction of the mixed-initiative strategy, however, can be complicated. It is suggested that dialogue strategies should be used freely and be able to support other strategies when required.

3.3.2. Dialogue Control

According to different dialogue initiative strategies, the methods to control a dialogue flow can vary. Concerning my research topic, the speech-based dictionary system defined in chapter 1 does not need many dialogues with complex tasks. Thus, the two simple methods for implementing the dialogue control will be presented as follows.

1. *Finite state-based dialogue control.* A dialogue structure based on a finite-state model can be seen as a state transition network. The nodes indicate computer responses, while the arcs indicate user inputs that change the dialogue states. A state shows an action in the dialogue, for example, asking a question. The finite state model offers ease of use and is suitable for well-structured tasks. A main drawback of the finite-state model is the lack of flexibility, so it cannot model such complex and unpredictable tasks.
2. *Frame-based dialogue control*. A dialogue structure is based on the form-filling tasks. Therefore, the dialogue involves asking questions to elicit the required information, and then submitting the queries to a database. The frame-based dialogue control model is suitable for implementing mixed-initiative dialogue strategy. For instance, a user provides the required information and a system asks for the missing information.

To sum up, to implement speech-based applications, spoken dialogue technology is needed. This chapter presents the three main components of spoken dialogue technology and their principles: speech recognition, Text-to-Speech synthesis, and dialogue manager. The properties of speech recognition and Text-to-Speech systems are very important for developers to know and understand since the developers need to consider them and choose the one that suits their application's requirements. Current speech recognition and Text-to-Speech systems presented in chapter 3 indicate the great advancement of each speech technology. Particularly, Text-to-Speech technology now increases more naturalness and intelligibility qualities and is available in many languages. That implies a more potential to integrate these speech technologies into applications in other fields. The developers also need to understand the dialogue management methodologies because they are the key success to control a system to perform efficiently and smoothly. Again, the methods for the dialogue management should be selected to suit the needs of the application. Regarding my second research question, speech recognition and Text-to-Speech technology existed a few decades ago. Recently, they have enough good qualities to employ to implement a speech-based dictionary system. In addition, system-initiative strategy and finite state-based dialogue control are the suitable dialogue management methods for implementing the speech-based dictionary system.

4. Principles of spoken dialogue system development

In this chapter, the principles of spoken dialogue system development will be explained. A software development process of the spoken dialogue system is not much different from a conventional software development process. Basically, the software development process is comprised of the following activities and steps: requirement analysis, functional specification, design, implementation, testing, and evaluation. However, the spoken dialogue development directly involves speech interfaces. Therefore, the development methodologies are definitely different from the ones for graphic user interface (GUI)-based system. The following sections present methodologies and guidelines that are specific to speech-based interfaces of each software development process.

4.1. Requirements analysis

Data collection and requirements analysis are the first step in the spoken dialogue system development. These can be carried out by using the following methods: use case analysis and spoken language requirements analysis. The purpose of the use case analysis is to identify usage requirements of a target system and data for defining processes of a target system. The spoken language requirements analysis involves specifying vocabulary, grammars, and interaction patterns. These two methods can be described in more details as follows.

1. *Use case analysis*. The main issues for the use case analysis are given below. These are helpful for developers to consider and gather usage requirements and all information for the target system.
    - Is speech an appropriate medium for the target system?
    - Will the target system replace or complement an existing system?
    - Who are the users of the target system?
    - What are the motivations for using the target system?
    - What kind of services should the target system provide?
    - What kind of tasks will the user want to complete?
    - What kind of information will the user want to retrieve?
    - How the target system should be designed from the user's perspective?
    - What is the target system behavior when the user communicates with?
    - What are all externally visible behaviors?
    - How the target system is to be used?
    - What are the target system responses with respect to the user's stimulus?

- What are the benefits that the user will receive from the target system?
- How often will the user use the target system?
- Will there be major dialect and accent differences in the areas of deployment?
- How comfortable will the user be with the target system?
- In which type of environment will the user use the target system?

When all answers are completely gathered, it is useful to model the use case diagram to simply understand an overview of the target system functionality based on use cases.

2. *Spoken language requirements analysis*. The spoken language requirements analysis helps the developers to decide which technologies should be used. There are three ways to do the analysis by studying human-human- dialouges, a simulation of human-computer experiments, and a real human-computer operation. A detailed explanation of the three methods is given below.

   - *Human-human dialogues study*. This study can form the basis for the design of human-computer interaction. The human-human dialouges provide a variety of information, e.g. the dialogue and language structures, the vocabulary and grammars used, the dialogue management, etc. The information can be obtained, for example, by observing how people interact with each other to complete their tasks. However, not all results from the human-human dialogues study can be applied to the humban-computer dialogues. They need to be verified before being used in the desing process.
   - *Simulated human-computer experiments study*. The Wizard of Oz method is used in the simulated human-computer interaction study. The method involves the simulation of human-computer interaction which is done all by humans. All required data can be collected and analyzed for a basis of the design from the simulation. To collect data efficiently, the simulation can be performed by simulating either all or some operations of the system.
   - *Real human-computer operation study*. The purposes of the real human-computer operation study are to find the real human-computer interaction and real system functions and to collect data for further improvement of the system. Also, the study is aimed to find suitability and unsuitability of the domain areas for speech-based application.

4.2. Requirements specification

The purpose of the requirements specification is to provide a description of all requirements, including databases, functional as well as non-functional requirements, and design constraints, in a formal document. For interactive speech systems, a detailed description of requirements specification can be found in the research paper by Bernsen et al. (cited in [McTear, 2004, pp. 136]).

4.3. Design

The design phase involves describing data structures of the system, overview of the system, system architecture, detailed design of the system components, and user interface. The outcome of the design phase is to give a basis for the implementation of the system. The design can be divided into two categories: high-level and low-level design.

1. High-level design includes a detail description of all architectural system components and their relationships, the data flow within the system, the user tasks, the dialogue flow and the dialogue elements, e.g. prompts, system help, recognition grammars, navigational commands, and databases.
2. Low-level design includes a detail description of the usability and performance of the system, and error management.

The following sections present guidance in speech-based system design.

4.3.1. Barge-in implementation (full- or half- duplex)

Barge-in implementation is targeted to let the users interrupt system prompts by speaking over the system. Barge-in capability, for instance, can be used when the system presents long lists and this case is definitely recommended. An advantage of barge-in usage is the faster tasks completion. However, the system should also support inactive barge-in for some situations.

4.3.2. Prompts

Prompts design plays an essential role in spoken dialogue systems design since the effective prompts design can reduce system errors and lead to successful interaction. The purposes of the prompts design are to provide all necessary information to guide the interaction and help performing speech recognition, language understanding and dialogue management functionalities smoothly. In prompts design it is recommended that the phrasing of prompts should be created appropriately, for example, short, clear and understandable. There are several techniques used to design prompts. These techniques can be explained as follows.

- *Tapered prompts.* The tapered prompts involve shorteninig the prompts when the same information is requested or presented. Tapered prompts are quite suitable for creating lists.
- *Incremental prompts.* The incremental prompts provide more detailed instructions after a problem occurs.
- *Leading prompts.* The leading prompts provide words that the user can respond to the system.
- *Expanded prompts.* The expanded prompts involve creating an explicit question for the user's second try because it is quite likely that the system could not understand the user's response at the first try.
- *Hints.* The hints give the most possible choices to the user in each dailogue situation.

4.3.3. Grammars

Grammars design involves defining all words and phrases which a user can respond to the system at a specific point in each dialogue. Grammars can be designed to be either simple or complex. A simple grammar is easy to build, while a complex grammar is more difficult to build. Besides, the simple grammar limits the user's responses but the complex grammar offers more flexibility of the user's responses. Hence, it depends on the developers what type of grammars to be used based on a suitability of their application.

4.3.4. Interaction style

System-initiative dialogues can be designed easily with simple grammars but they may take long time to complete the dialogue. Mixed-initiative dialogues provide a user with more flexibility but they need efficient methodologies to construct.

4.3.5. Navigation and menu commands

Navigation and menu can give a successful interaction in speech interface design, for example, for interactive voice response (IVR) systems. The helpful navigation and menu commands should be provided by a system, such as "help", "repeat", and "exit".

4.3.6. System Help

System help can be designed by using a separate dialogue or a sequence of prompts approach. The separate dialogue approach also needs a method to go back to a main dialogue. The sequence of prompts approach needs to have multiple prompts for each turn in the dialogue. In addition, feedback and guidance are necessary to be provided by the system too.

4.3.7. Error management

There are several strategies to correct errors, for example, using an explicit confirmation, implicit confirmation, yes/no question, repeating, spelling, system help and feedback or selective option from a list. Moreover, switching of dialogue management strategies can be used to solve the problems in the interaction. Furthermore, providing various kinds of grammars can solve speech recognition errors and it would be better to design and test the grammars and vocabularies before implementing the application. Also, changing to other modalities, such as telephone keys, can be used to correct errors. This method is efficient in dictation and telephony applications. It is suggested that the design of error correction should avoid all causes which might make errors occur again.

4.4. Implementation

The implementation process is the execution of an idea, model, design, requirements specification, and methodology. In other words, this process is actually programming the code for the system. The implementation of the system can vary depending on the development tools and platforms. For example, selecting VoiceXML platforms, the system will be developed in which conforming to the VoiceXML environment.

4.5. Testing

The testing process is the assessment of the quality of the system, for example, whether or not the system operations meet the requirements specification. It also includes the processes of executing the system in order to find any software bugs. The white box testing and the black box testing are the two traditional approaches to do the testing process. The white box testing covers unit testing and integration testing, whereas the black box testing covers validation testing and system testing. A detail description of the four types of testing is given below.

- *Unit testing*. The unit testing tests the minimal system module. The purpose of the unit testing is to verify if each system unit has been precisely implemented.
- *Integration testing*. The integration testing identifies defects in the interfaces and interaction between integrated modules.
- *Validation testing*. The validation testing examines if the system meets the function and non function requirements specification.
- *System testing*. The system testing tests a fully integrated system to verify if it meets its all requirements specification.

4.6. Evaluation

The evaluation of spoken dialogue systems can be separated into two groups: technology evaluation and usability evaluation. The technology evaluation measures a performance of technical components. Mostly, the technology evaluation is concerned with a measurement of speech recognition performance, i.e. word error rate, and Text-to-Speech performance, i.e. pleasantness and intelligibility. The usability evaluation involves measuring the users' effectiveness with the system and collecting users' opinions about the system, e.g. user satisfaction, cognitive load, and user preferences.

In summary, chapter 4 presents the principles of spoken dialogue systems development described in terms of software development process. Referring to my third research question, these principles, including the proposed methods from the review in chapter 2 and knowledge of spoken dialogue technology can form the basis for the implementation of speech-based system for a dictionary.

5. Speech-based dictionary system development

From the summary of the reviewed research on speech-based dictionary systems presented in chapter 2, some of the proposed methods and development tools can be applied to a speech-based dictionary system development. Additionally, the determination of spoken dialogue technology for a speech-based dictionary system can be done based upon the knowledge base given in chapter 3. Also, the processes of spoken dialogue systems development are described in chapter 4. Consequently, in this chapter I am going to propose my ideas of implementing a speech-based dictionary system based on the knowledge mentioned above, which will be presented in terms of software development process. A prototype of the speech-based dictionary system including its selected testing procedures will be presented too. However, the usability test and the evaluation of the prototype will be left for a future work but the usability plan will be presented in Appendix 2.

5.1. Requirements analysis for the speech-based dictionary system

The following questions and answers indicate the use case analysis for determining the usage requirements and all information needed for the speech-based dictionary system. Table 5 presents the questions asked about the goals of the target system.

Question	**Answer**
Is speech an appropriate medium for the target system?	Comparing to dictionary applications with GUI mode, the target system with speech mode may not be as efficient as those because of the limitations of speech technologies, e.g. speech recognition errors. However, a speech interface proposes its availability at anytime as well as its accessibility. The speech interface is also more suitable in hands-free, eyes-busy environments, and for disabled people, especially visual impaired persons. In addition, speech is a suitable modality for requesting specific information. Consequently, the speech mode is appropriate for requesting a word definition. Considering the limitations of speech technologies, the speech interface is rather appropriate for small-scale dictionary applications expandable to middle-scale applications.

Will the target system replace or complement an existing GUI-based dictionary application?	The target system will replace the existing GUI-based dictionary application because of the following benefits. - It will present the pronounced word definitions so the users will learn how to pronounce words. - It will provide a natural way to obtain the word definitions by speaking a word. (hands- and eyes- free benefit) - It can be adapted for the disabled users' dictionary usage.
How the target system should be designed from the user's perspective?	The target system should be easy to use and should provide fast and accurate performance.
What kind of services should the target system provide?	The speech-based dictionary system should simply provide a word or phrase definition lookup service. However, other lexical information such as synonyms, antonyms, and examples of a word usage can be provided too if available in a reference dictionary database.
Who are the users of the target system?	Anyone who has normal speaking and listening skills.
What kind of tasks will the users want to complete?	- Look up a word or phrase definition by speaking a clear word or phrase into the target system. - Listen to the word or phrase definitions, including synonyms, antonyms, and example sentences of the word or phrase query if available. - Listen to the definitions, including other lexical information if available again. - Look up a new word or phrase. - Exit the target system.
What kind of information will the users want to retrieve?	Generally, the users want to retrieve a definition of a word or phrase with or without additional information, e.g. synonyms, antonyms, and sample sentences.
What is the target system behavior	- Lead a dialogue to reach the users'

when the users communicate with?	goal. - Provide help when the users require. - Handle errors when the users make a mistake.
What are all externally visible behaviors?	- The users ask the target system for a word or phrase definition. - The target system returns the audible word or phrase definition, including synonyms, antonyms, and example sentences of the word or phrase query if available. - The users ask the target system to repeat the result information again. - The target system presents the result information again. - The users ask the target system for a new word or phrase definition. - The target system presents the audible new word or phrase definition, including synonyms, antonyms, and example sentences of the word or phrase query if available. - The target system provides help when the users request. - The target system handles errors when the users make a mistake. - The users handle errors when the target system makes a mistake. - The users exit the target system. - The target system is automatically closed.
What are the benefits that the users will receive from the target system?	- The users will obtain the word or phrase definitions and hear them at the same time so the users will learn how to pronounce words correctly. - The users will be able to do simultaneous tasks, for example, reading a book, typing a document, and listening to a word or phrase definition

	at the same time. - An alternative mode to look up a word or phrase definition.

Table 5. Questions asked about the goals of the speech-based dictionary system and the answers to the questions.

Some questions for analyzing the users of the target system, along with the answers are presented in table 6. The answers can indicate the main characteristice of the target system and the level of expertise the users have with speech applications.

Question	**Answer**
What type of user?	Student, teacher, tourist, etc.
Will many of the users be non-native speakers of the proposed language?	Yes, they will mostly be.
What is user education level?	Junior school, high school, college, University.
What are the motivations for the users to use the target system?	The users prefer to use speech for searching and obtaining a word or phrase definition, including synonyms, antonyms, and example sentences of the word or phrase query if available, and to hear the pronunciation of the word definition.
Will there be major dialect and accent differences in the areas of deployment?	There will be the major accent differences, e.g. US accent, UK accent and non-native accent. However, it depends on how efficient speech recognizer performance will accept accent variations.
How comfortable are the users with the target system?	The users should feel comfortable when using the target system. For example, the users understand the questions asked by the target system and can answer with or without help.
Experience with speech-based system?	None
How much help will be required?	Basic help until familiar with the target system.
Most common tasks performed?	- Look up a definition of a word or phrase - Look up synonyms and antonyms of a word or phrase query - Find an example use of word

Familiarity of user with tasks?	Very familiar with GUI-based system.

Table 6. Questions for analyzing the users of the speech-based dictionary system and the answers to the questions.

The usage profile can indicate the target system and hardware requirements. Table 7 presents some questions asked in relation to a dictionary usage together with the corresponding answers.

Question	**Answer**
Frequency of use?	Rarely to regularly
Availability and accessibility of the target system?	Any time
Input device type?	Microphone
Output device type?	Speaker
In which types of environment will the users use the target system (e.g., a quiet home/office, outdoors, a noisy place)?	Quiet places such as home, office, school, etc.
How the target system is to be used?	The target system is used on a desktop and laptop computer with a microphone and speaker.

Table 7. Questions for the analysis of usage profile for the speech-based dictionary system.

Regarding the speech-based dictionary system introduced in chapter 1, the target system will be operated on a desktop and laptop computer. Therefore, the data collection will be carried out based on how users use dictionary programs on a computer. The real human-computer operation study method can thus be applied for the spoken language requirements analysis. It is obvious that the dictionary programs on computers are GUI-based systems. Thus, the observations of how users communicate with the existing GUI-based dictionary systems can be used for a basis of the analysis. Table 8 shows tasks analysis regarding the observations. The shown tasks are performed repeatedly by the systems. Since there are a large number of GUI-based dictionary systems and they provide the users with different lexical information, these tasks analysis cannot be defined to be compatible with most speech-based dictionary systems.

Dictionary System		
Component	**Functionality**	**Additional detail**
Lookup	Search for definitions	
Presentation	Present definitions with or without synonyms, antonyms, and sample use of word	Definitions presented according to the word classification (n, v, adj, adv, etc.)

Table 8. Tasks analysis for the speech-based dictionary system.

It is noticeable that the existing GUI-based dictionary systems provide a large number of vocabularies. Besides, the users require few actions to interact with a computer to complete their tasks, for instance, typing word query, clicking button to search for definitions, scrolling to read definitions. Hence, the target system should also provide a large number of vocabularies. However, due to the limitations of speech technologies, limited vocabularies should be suitable for the target system. The grammars for the target system should be simple, limited and customizable. In addition, the target system should provide short conversations to reach the user's goals. Below shows a brief example dialogue which can be achieved based upon the collected result of the observations. This can give a quick overview of how a dialogue structure and a dialogue flow will be.

1 System: Welcome to the dictionary lookup system.
2 Say a word or phrase you want to look up.
3 User: Dictionary
4 System: You look up the definitions of dictionary. Is it correct?
5 User: Yes
6 System: The definitions are as follows.
Dictionary noun
1. Definition detail
2. Definition detail
3. Definition detail
Please select one of:
1. repeat the definitions
2. new lookup
3. exit
7 User: Three
8 System: Ok then, good bye.

5.2. Requirements specification for the speech-based dictionary system

This section presents some sample requirements specification using requirements specification template from the software project management course material (see in Appendix 1).

Use case diagram for the target system is illustrated below. It displays an overview of the functionality of the target system from the user's perspective.

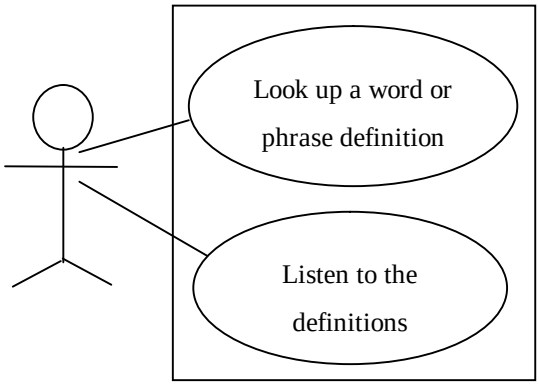

Figure 1. Use case diagram for the speech-based dictionary system.

Function requirements for the speech-based dictionary system are presented in Table 9.

Function	Lookup
Purpose	To look up a word or phrase definition from a database.
Input	A spoken word or phrase.
Handling	The target system accepts speech input from the user and then checks whether it matches the speech recognizer grammar. If valid, the word or phrase query will request the responsive definitions from the database.
Output	Textual definitions of the word or phrase query.
Error handling	If the speech input does not match the speech recognizer grammar, the target system guides the users to speak the query word clearly and loudly or, otherwise, speak another query word.
Function	Presentation
Purpose	To present the spoken word or phrase definition.
Input	Textual definitions of the word or phrase query.
Handling	The target system sends the textual definitions of the word or phrase query to the Text-to-Speech component in order to transcribe them to be in a spoken form.
Output	Spoken definitions of the word or phrase query.

Error handling	If the database connection fails, The target system informs an error message.
Function	Options
Purpose	To provide the options for users to 1. look up a new word or phrase 2. listen to the definitions again 3. exit the target system.
Input	Selected option (e.g. one, two, or three).
Handling	Option 1: The target system asks the users to give the new word or phrase query. Option 2: The target system speaks the definitions again. Option 3: The target system terminates all tasks.
Output	Option 1: The target system prompt for the users to give the new word or phrase query. Option 2: Spoken definitions of the word or phrase query. Option 3: The target system is automatically closed.
Error handling	If each of the provided options is not selected, the target system guides the users to speak the possible user's response.

Table 9. Function requirements for the speech-based dictionary system.

Other requirements in relation to non-function, hardware, software and communication are presented in Table 10.

Nonfunctional requirements	The target system gives three attempts to a user for speaking a word or phrase query. If the target system does not understand the user input at the third attempt, it assumes that there is no the word the user has been asking stored in the database. Therefore, the target system informs the user to query another word or phrase.
Hardware requirements	A desktop and laptop computer with a microphone and speaker.
Software requirements	Depend on the development platforms and tools.
Communication requirements	If the database of the target system runs on a web server, an Internet connection is required.

Table 10. Nonfunctional, hardware, software, and communication requirements for the speech-based dictionary system.

5.3. Design for the speech-based dictionary system

The overall architecture of the speech-based dictionary system can be viewed in Figure 2. The system is composed of a spoken dialogue system that communicates with a user and database. The database can be considered as an integral part since it can be diversely constructed to cooperate with a main system. In fact, the functionality of the spoken dialogue system is mostly determined by the database structure. Thus, in this section a detailed design of the database will not be considered because the database of a dictionary can vary. The design for the speech-based dictionary system will mainly focus on issues relevant to spoken dialogue technology.

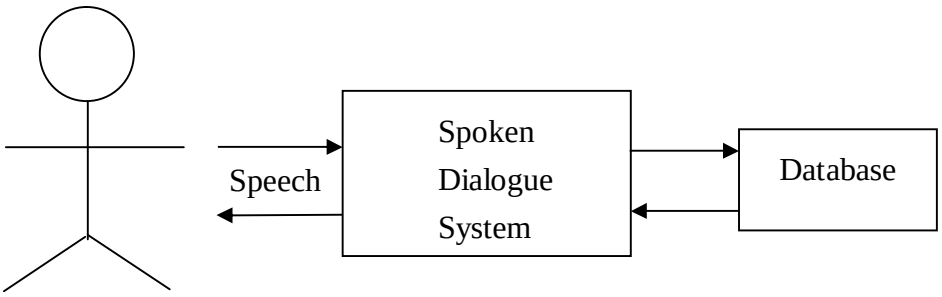

Figure 2. Overall architecture of the speech-based dictionary system.

The user tasks can be illustrated in a task hierarchical diagram (THD). The diagram indicates each user task with its subtasks and also shows the interaction between a user and system from a user's perspective. Figure 3 illustrates a task hierarchical diagram for the speech-based dictionary system. The top level task begins with looking up the definitions of a word or phrase. Then, listening to the results of the word or phrase definitions is the next task. After that, a selection of three options task should be made in order to lead to a further operation.

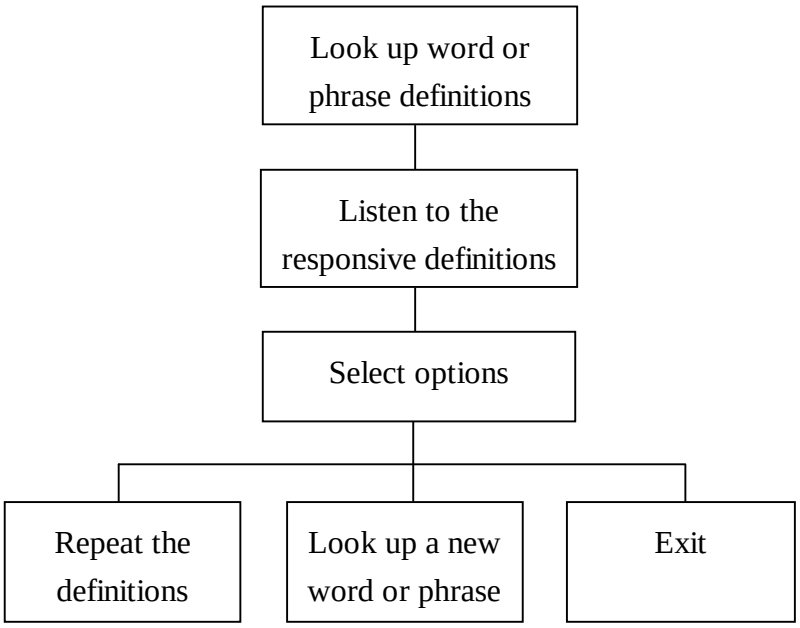

Figure 3. Task hierarchy diagram for the speech-based dictionary system.

Based on the task hierarchy diagram in Figure 3, the dialogue management can be formed for the speech-based dictionary system, i.e. dialogue initiatives, spoken dialogues, dialogue flow, and dialogue grammars. The system-initiative dialogue is suitable for the system because the system can direct the users to obtain word or phrase definitions quickly. Figure 4 shows the overview dialogues and dialogue flow together with the prompts, user response types (e.g. options) and error management using the system-initiative dialogue strategy.

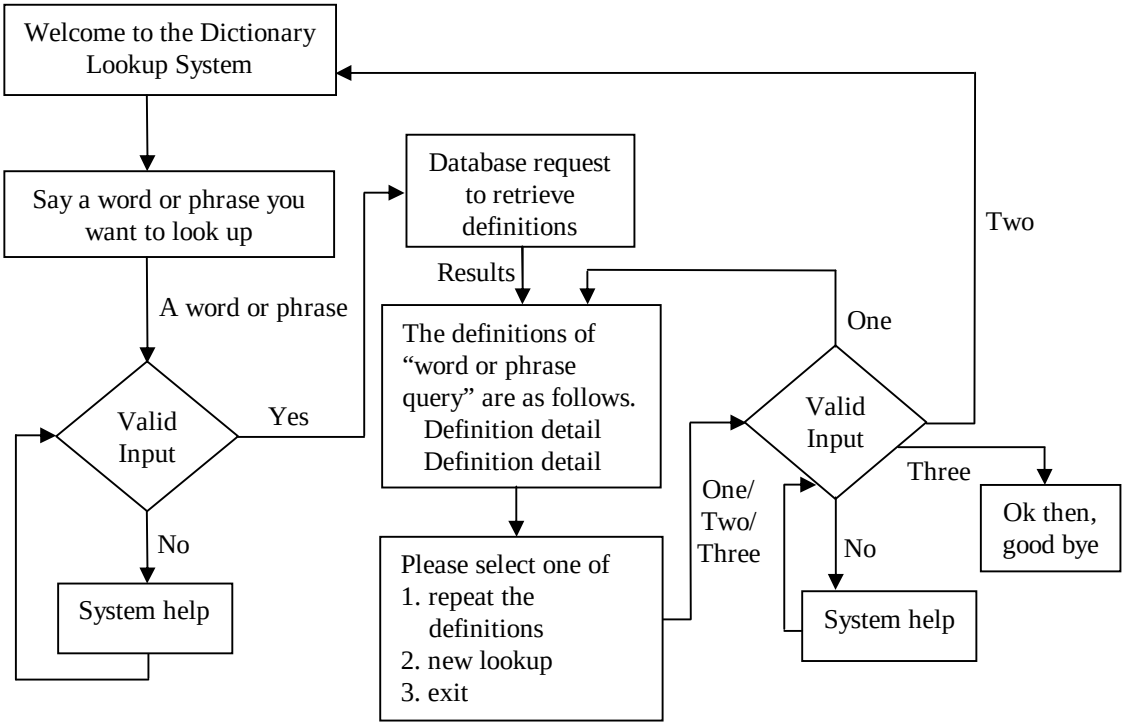

Figure 4. Overview dialogues and dialogue flow for the speech-based dictionary system.

The following shows further design issues for the speech-based dictionary system.

- *Grammars*. Recognition grammar for a word or phrase query is a list of vocabularies available in a database of a reference dictionary, which can be customized. Recognition grammars for selecting options are "one", "two", and "three". Yes/no are the grammars for explicit confirmation and the grammars for users to ask for help are "please I need help", "I need help", "help me please", "help me", and "help please". For barge-in, the grammars are "exit", "quit", "stop", and "cancel".
- *Barge-in*. A dictionary often gives a long list of word definitions since there are lots of words containing several definitions. For the speech-based dictionary system, barge-in would thus be appropriate while the system is presenting the definition results because it is likely that the users prefer to stop listening to the long definitions. Then, with the barge-in capability, the users can do the next desired task such as move back to hear the definitions, look up a new word, or exit the system. Below is a sample dialogue when barge-in is activated.

  User: Stop
  System: You have chosen to stop. Please select one of:
  1. back to listen to the definitions
  2. new lookup
  3. exit
  User: Two

System: Welcome to the dictionary lookup system.
Say a word or phrase you want to look up.

- *System help or error management.* In the first dialogue related to inputting a word or phrase query, reprompt, prompts, and explicit confirmation strategies are used to handle no input, unmatched input and to verify valid input respectively. The following shows example dialogues for error management of the first dialogue.

System: Welcome to the dictionary lookup system.
Say a word or phrase you want to look up.
User: …….. (no input)
System: I didn't hear anything. Say a word or phrase you want to look up.
User: jpeg (unmatched input)
System: No matches found, please try to speak it clearly.
User: jpeg
System: No matches found, perhaps try to say another word or phrase.
User: prejudice (valid input)
System: You look up the definitions of prejudice. Is it correct?
User: …….. (no input)
System: I didn't hear anything. You look up the definitions of prejudice. Is it correct?
User: That’s right. (unmatched input)
System: Say yes if it's the correct word you want to look up. Say no if not.
User: Yes
System: The system is searching for the definitions.

In the dialogues relevant to selecting options, reprompt strategy is used to handle no input and unmatched input, while prompts strategy is used to give help when users request. The sample error management of these dialogues are given below.

System: Please select one of:
1. repeat the definitions
2. new lookup
3. exit
User: …….. (no input)
System: I didn't hear anything. If you want to repeat the definitions, say one. If you want to look up a new word, say two. If you want to exit the system, say three.

User: Nine (unmatched input)

System: If you want to repeat the definitions, say one. If you want to look up a new word, say two. If you want to exit the system, say three.

User: Help

System: Say one to repeat the definitions. Say two to look up a new word. Say three to exit the system.

User: One

System: You have chosen to stop. Please select one of:
1. back to listen to the definitions
2. new lookup
3. exit

User: …….. (no input)

System: You have chosen to stop. Please select one of:
1. back to listen to the definitions
2. new lookup
3. exit

User: Exit (unmatched input)

System: You have chosen to stop. Please select one of:
1. back to listen to the definitions
2. new lookup
3. exit

User: Help

System: Say one to go back to hear the definitions. Say two to look up a new word. Say three to exit the system.

User: Two

5.4. Implementation of the speech-based dictionary system

To implement a speech-based system for a dictionary, a database is a main issue to be carefully considered because each dictionary database provides different information. Therefore, it depends on developers to determine a type of the dictionary database. In this thesis, WordNet lexical database is considered to be a reference database for a prototype of the speech-based dictionary system because it provides a good resource for researchers in linguistics, information retrieval, and several relevant areas [Fellbaum, 1998]. Moreover, it is free to use and can be publicly downloaded. The WordNet database format is grouped as verbs, nouns, adjectives and adverbs all of which consist of the following lexicographic information: definitions, examples, synonyms, and antonyms. Thus, the WordNet database provides fairly enough information to be presented in a spoken form to a user. In WordNet packages, source code files are

provided for researchers in order to modify and customize them to suit their application requirements. However, the development of WordNet database for the prototype of speech-based dictionary system would be skipped in this thesis. Referring to the research paper by Rouillard [2007], an issue on a free dictionary web service was mentioned. Therefore, the free dictionary web service would instead be a solution to be a reference database for the prototype of speech-based dictionary system. Then, VoiceXML and web service would be required for the development environment. Accordingly, the proposed prototype for Windows was implemented as a client-server application by using VoiceXML platform (i.e. OptimTalk Desktop Suite Basic Edition), ACME server, Apache Axis2/Java, and DictService web service. A more detailed description of these development tools is explained below.

- *OptimTalk Desktop Suite Basic Edition*. The OptimTalk Desktop Suite Basic Edition developed by OptimSys [2004] provides developers with tools to implement VoiceXML based applications running on a desktop computer for free of charge. It supports features of, for example, speech recognition and speech synthesis via Microsoft Speech API, speech recognition grammar (SRGS) and semantic interpretation (SISR), and audio input and output. However, VoiceXML 2.1 and Speech Synthesis Markup Language (SSML) are not suppoted in this Basic Edition.
- *ACME server*. The ACME server is a Java-extensible HTTP server created by Acme Laboratories [ACME Labs, 1997]. It serves as a web server and can be used to implement the Servlet API in order to interact with external databases.
- *Apache Axis2/Java*. The Apache Axis 2/Java developed by the Apache software foundation is an open-source project providing the core engine for web services [Apache, 2005]. Axis 2 can be used to create a client Java class which can invoke a required operation from a web service.
- *DictService web service*. The DictService is a free dictionary web service created by Aonaware [2005]. It provides the users with word definitions from a group of dictionary databases one of which is the WordNet (r) 2.0.

Next, the detailed architecture of the prototype can be viewed in Figure 5.

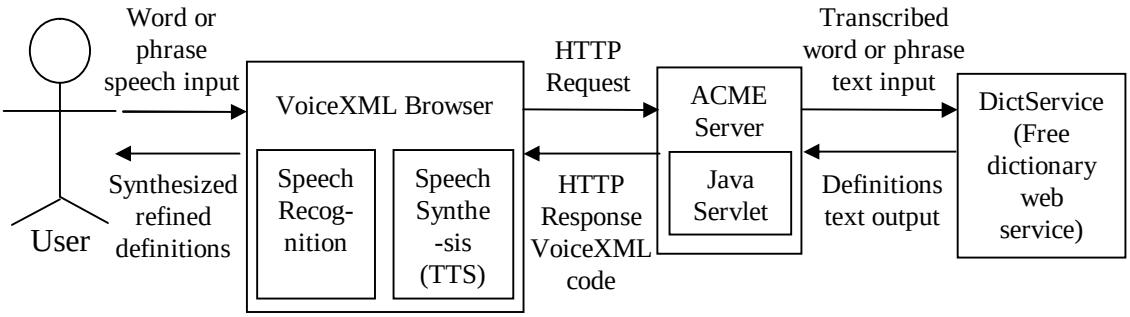

Figure 5. The architecture of the prototype of speech-based dictionary system.

The following describes a process flow in the prototype.

1) A VoiceXML application constructed based on the system-initiative strategy welcomes a user to the dictionary lookup system and asks the user to say a word or phrase he/she desires to look up. This VoiceXML application uses Microsoft English (U.S.) v6.1 Recognizer to recognize the user input and uses Microsoft Text to Speech with LH Michelle voice to communicate the user. If the user speech input is valid, the speech recognizer transcribes the user speech input into text form. If not, the application will guide the user to handle the error until it gets the valid input.

2) The transcribed word or phrase in text form is transmitted to the ACME server via HTTP.

3) In the ACME server, a client function coded in Java Servlet is executed. It first receives the string word or phrase parameter. Next, it connects to the DictService web service to invoke the DefineInDict method and then submits the parameters, i.e. dictId parameter set to “wn” (WordNet) and the string word or phrase parameter. If there are definitions for the string word or phrase input, the DictService web service with DefineInDict method will return the definitions together with synonyms, antonyms, and example sentences from WordNet (r) 2.0 dictionary to the Java Servlet. If not, some error messages will be returned instead.

4) The obtained definitions from the DictService web service are in text form ready to be displayed on screen, so there are some words or characters which are not meaningful by listening. For example, n, v, adj, adv, syn, ant, [,], {, }, and `. Therefore, the obtained definitions are properly refined by the refinement processes in order to be comprehensible. The refinement processes include deleting [, ], {, }, and `, and changing some abbreviations to full names as follows:

   - n -> noun;
   - v -> verb;

- adj -> adjective;
- adv -> adverb;
- syn -> synonym;
- ant -> antonym.

They also include adding the word *Example* following with a number to the example sentences. Then, a VoiceXML document coded in the Java Servlet transmits the refined textual definitions to the speech synthesis (TTS) via HTTP.

5) The speech synthesis (TTS) transcribes the refined textual definitions into speech form and presents them to the user. Then, the VoiceXML application asks the user to select the three menu options, i.e. 1. repeat the definitions, 2. new lookup, and 3. exit. The selection can be done either by barging-in the system while presenting the definitions with saying the word *stop*, *quit*, *exit*, or *cancel* or by choosing the options after the system completes the results presentation.

5.5. Testing for the speech-based dictionary system

In this section I present some testing processes for the prototype of the speech-based dictionary system. This includes unit testing, path testing, validation testing, and system testing. The following shows test case examples and test results of the unit testing and path testing. Each input was tested many times and the actual test results were shown based on recognition accuracy estimated.

Prompt: *Say a word or phrase you want to look up*			
Input	**Expected result**	**Actual result**	**Pass/fail**
(no input)	“I didn't hear anything.”	“I didn't hear anything.”	Pass
adversely	adversely	random words (e.g. city, roughly, knock up)	Fail
get to grips	get to grips	get to grips	Pass
schooner	schooner	tender	Fail
confidential	confidential	confidential	Pass
profane	“No matches found, please try to speak it clearly.”	random words (e.g. flame, vane, telephone)	Fail
zill	zill	wheel	Fail
whereas	whereas	massachusetts	Fail

Table 11. Test case example of the main dialogue (recognition).

Prompt: *You look up the definitions of (recognized word). Is it correct?*			
Input	**Expected result**	**Actual result**	**Pass/fail**
(no input)	"I didn't hear anything." Reprompt	"I didn't hear anything." Reprompt	Pass
yes	Continue to SampleVXMLSevlet	Continue to SampleVXMLSevlet	Pass
no	Back to the first prompt "Say a word or phrase you want to look up"	Back to the first prompt "Say a word or phrase you want to look up"	Pass
That's right	System help	Continue to SampleVXMLSevlet	Fail

Table 12. Test case example of the main dialogue (path).

Input	**Expected result**	**Actual result**	**Pass/fail**
adversely	"The definitions are as follows. **adversely** **adverb** in an adverse manner **example 1** she was adversely affected by the new regulations"	"The definitions are as follows. **adversely** **adverb** in an adverse manner **example 1** she was adversely affected by the new regulations"	Pass
get to grips	"The definitions are as follows. **get to grips** **verb** deal with (a problem or a subject) **example 1** I still have not come to grips with the death of my parents **synonym** come to grips"	"The definitions are as follows. **get to grips** **verb** deal with (a problem or a subject) **example 1** I still have not come to grips with the death of my parents **synonym** come to grips"	Pass
schooner	"The definitions are as follows. **schooner** **noun 1** a large beer glass **noun 2** sailing vessel used in former times"	"The definitions are as follows. **schooner** **noun 1** a large beer glass **noun 2** sailing vessel used in former times"	Pass

confidential	"The definitions are as follows. **confidential** **adjective 1** entrusted with private information and the confidence of another **example 1** a confidential secretary **adjective 2** (of information) given in confidence or in secret **example 1** closet information **example 2** this arrangement must be kept confidential **example 3** their secret communications **synonym** closet (a), secret **adjective 3** denoting confidence or intimacy **example 1** a confidential approach **example 2** in confidential tone of voice **adjective 4** the level of official classification for documents next above restricted and below secret available only to persons authorized to see documents so classified"	"The definitions are as follows. **confidential** **adjective 1** entrusted with private information and the confidence of another **example 1** a confidential secretary **adjective 2** (of information) given in confidence or in secret **example 1** closet information **example 2** this arrangement must be kept confidential **example 3** their secret communications **synonym** closet (a), secret **adjective 3** denoting confidence or intimacy **example 1** a confidential approach **example 2** in confidential tone of voice **adjective 4** the level of official classification for documents next above restricted and below secret available only to persons authorized to see documents so classified"	Pass
zill	"The definitions are as follows. **zill** **noun** one of a pair of small metallic cymbals	"The definitions are as follows. **zill** **noun** one of a pair of small metallic cymbals	Pass

	worn on the thumb and middle finger used in belly dancing in rhythm with the dance"	worn on the thumb and middle finger used in belly dancing in rhythm with the dance"	
whereas	"No definitions found for whereas in WordNet 2.0 dictionary."	"No definitions found for whereas in WordNet 2.0 dictionary."	Pass

Table 13. Test case example of search results (function).

Prompt: *Please select one of: 1. repeat the definitions* *2. new lookup* *3. exit*			
Input	**Expected result**	**Actual result**	**Pass/fail**
(no input)	"I didn't hear anything." Reprompt	"I didn't hear anything." Reprompt	Pass
one	Back to search results	Back to search results	Pass
two	Continue to main dialogue	Continue to main dialogue	Pass
three	"Ok then, good bye" Exit the system	"Ok then, good bye" Exit the system	Pass
one hundred	System help	System help	Pass

Table 14. Test case example of menu options (path).

Prompt: *You have chosen to stop. Please select one of:* *1. back to listen to the definitions* *2. new lookup* *3. exit*			
Input	**Expected result**	**Actual result**	**Pass/fail**
(no input)	Reprompt	Reprompt	Pass
one	Back to search results	Back to search results	Pass
two	Continue to main dialogue	Continue to main dialogue	Pass
three	"Ok then, good bye" Exit the system	"Ok then, good bye" Exit the system	Pass
two hundred	System help	System help	Pass

Table 15. Test case example of barge-in (path).

The validation testing and system testing were accomplished by performing a number of tests at unit and path level. The results of validation testing can be concluded that the prototype positively meets the functional requirements, i.e. presentation and options function, and other requirements but poorly meets the requirement of lookup function which is the core function of the system. However, the presented test results do not provide enough information for an analysis of this problem. Still, the usability test is needed to be carried out to collect more information. Then, the results of system testing can be shortly summarized that the prototype always takes about 1-2 minutes to load the recognition grammar for user's word or phrase queries (10,000 words) when the main dialogue is activated. This causes a slow transaction time. Furthermore, the overall reliability of the prototype is rather low due to the limits of speech recognizer performance and the OptimTalk development tool. For example, the speech recognizer often gives a result based on preceding inputs. A further example is that the barge-in feature of the OptimTalk cannot be configured to be false so the prototype is very sensitive to any input and functions improperly. Also, the OptimTalk restricts a prompt length so the prototype cannot present a spoken long definition to the user.

In summary of chapter 5, the development of speech-based dictionary system was proposed following the software development process. The prototype was implemented to test the feasibility based on the proposed ideas. Some testing processes were conducted but still could not provide enough information for the evaluation of the prototype. However, some main findings from the prototype development perspective can be described as follows. First, the OptimTalk Desktop Suite Basic Edition provides limited use of VoiceXML features, for instance, VoiceXML interpreter, so the prototype cannot properly perform its functionality as it should according to the design phase. In my opinion, the free version of the OptimTalk Desktop Suite is not a suitable

development tool for the speech-based dictionary system. However, free development tools for speech-based applications are rarely available. Indeed, commercial development tools might be good solutions. Second, constructing recognition grammar for a word or phrase query is a crucial issue. In my point of view, grammar and dictation approach are still not good approaches for the speech-based dictionary system. This is because the former spends much loading time if it contains a large number of words, while the latter may not guarantee a speech recognition result whether it is the user's required word query. Spelling approach might be a potential solution. Finally, all of electronic dictionary databases contain textual information which is ready to be displayed on screen. Therefore, free dictionary databases (e.g. from the DICT Development Group [DICT, 1997]) cannot be directly used for the speech-based dictionary system. The developers need an effort to modify and customize them to be audible and understandable information. Some examples are abbreviations (e.g. n, adj, adv, v, pl, c, etc.), special character (`), the word "see also" used to suggest reading more information of relevant word, and so on. An issue on constructing database for speech-based dictionary system is also crucial. There are many points should be considered, for instance, what type of data structure is appropriate for the database, which sorts of lexicographic information are suitable to be heard understandably by users, and customization tool for the database, etc. It is my considered opinion that a database for speech-based dictionary should be carefully constructed in an organized way in order to meet the requirements of lexicographers, developers, and users.

6. Conclusion

The deployment of speech interface in dictionary applications has been much focused on representing definitions aimed at presenting correct pronunciation of words rather on looking up definitions. This is due to more advanced Text-to-Speech technologies than speech recognition technologies. Thus, it is quite common today to see, for example, handheld electronic dictionaries with the ability to speak aloud word definitions. However, it is very rare to see the dictionary applications with speech input functionality in any end products but rather seen a little in research. This thesis proposes the development of speech-based dictionary application and the prototype illustrates a working system based on the proposed ideas. Given this, it can be inferred that it is feasible to implement such a very small speech-based dictionary system with most current speech technology but its efficiency and performance may be lower than ones of other dictionary applications. Concerning scalability and better performance, there remain some issues that need more research and development to find any solution that can fulfill, for instance, constructing structured lexical database and a potential approach for user speech input recognition. Moreover, the issue that will dominate discussion and research on speech-based dictionary system in the future will be a bilingual or multilingual speech-based dictionary, specific definition repeatability and a speech-based dictionary system on different mobile devices such as PDAs and mobile phones. Moreover, an embedded speech-based dictionary system with small vocabulary size, for example, for computer games and translation programs would be an interesting and a worth trying research topic.

References

[ACME Labs, 1997] Acme Laboratories, Class Acme.Serve.Serve http://www.acme.com/java/software/Acme.Serve.Serve.html (checked on December 20th, 2007)

[Aonaware, 2005] Aonaware, Dictionary Web Service http://services.aonaware.com/DictService/ (checked on December 20th, 2007)

[Apache, 2005] Apache Software Foundation, Apache Axis2/Java http://ws.apache.org/axis2/index.html (checked on December 20th, 2007)

[AT&T, 2002] AT&T Corporation, AT&T Natural Voices Text-To-Speech Engines System Developer's Guide Server and Desktop Editions http://www.wizzardsoftware.com/docs/ATTNaturalVoicesTTS14.pdf (checked on December 14th, 2007)

[Azulai et al., 2007] Ophir Azulai, Ron Hoory and Zohar Sivan, Dictionary Lookup for Mobile Devices Using Spelling Recognition http://www.wipo.int/pctdb/en/wo.jsp?IA=WO2007006596&DISPLAY=DESC (checked on October 13th, 2007)

[Chesnut, 2003] Casey Chesnut, FreeSpeech Project http://www.mperfect.net/freeSpeech (checked on March 23rd, 2007)

[CMU, 2004] Carnegie Mellon University, Sphinx-4 A Speech Recognizer Written Entirely in the Java™ Programming Language http://cmusphinx.sourceforge.net/sphinx4/ (checked on December 13th, 2007)

[Cox et al., 2000] Richard V. Cox, Candace A. Kamm, Lawrence R. Rabiner, Juergen Schroeter, Jay G. Wilpon, Speech and Language Processing for Next-Millennium Communications Services. In : *Proc. of The IEEE*. **88**(8), 2000, 1314-1337.

[Crystal, 1986] David Crystal, The Ideal Dictionary, Lexicographer and User http://www.crystalreference.com/DC_articles/Lexicography7.pdf (checked on October 11th, 2007)

[CSLU, 2004] Center for Spoken Language Understanding, Speech Synthesis Research http://cslu.cse.ogi.edu/research/tts.htm (checked on December 14th, 2007)

[de Schryver, 2003] Gilles-Maurice de Schryver, Lexicographers'Dreams in the Electronic Dictionary Age. *International Journal of Lexicography*. **16**(2), 2003, 143-199.

[DICT, 1997] DICT Development Group, Dictionary Server Protocol http://www.dict.org/links.html (checked on December 20th, 2007)

[Ectaco, 2007] Ectaco Inc., Ectaco, Inc. Official Site http://www.ectacoinc.com/ (checked on August 27th, 2007)

[Ehsani and Knodt, 1998] Farzad Ehsani and Eva Knodt, Speech Technology in Computer-aided Language Learning: Strengths and Limitations of a New CALL Paradigm. *Language Learning and Technology*. **2**(1), 1998, 45-60.

[ELLS Technical Work Group, 2002] ELLS Technical Work Group, The E-Language Learning System (ELLS) http://ott.educ.msu.edu/elanguage/ (checked on September 21st, 2007)

[Fellbaum, 1998] Christiane Fellbaum, WordNet: An Electronic Lexical Database http://mitpress.mit.edu/catalog/item/default.asp?ttype=2&tid=8106 (checked on December 20th, 2007)

[Franklin, 2007] Franklin Electronic Publishers Inc., English Dictionaries http://www.franklin.com/handhelds/dictionaries/english/ (checked on August 27th, 2007)

[Hakulinen, 2006] Jaakko Hakulinen, *Software Tutoring in Speech User Interfaces*. Academic Dissertation (PhD). University of Tampere, 2006.

[IBM, 2007] IBM Corporation, IBM Embedded ViaVoice – Features and benefits http://www-306.ibm.com/software/pervasive/embedded_viavoice/about/ (checked on December 12th, 2007)

[IBM Research, 2007] IBM Research, IBM Text-to-Speech Research http://www.research.ibm.com/tts/ (checked on December 14th, 2007)

[Lin et al., 1997] Sung-Chien Lin, Lee-Feng Chien, Ming-Chiuan Chen, Lin-Shan Lee, Ker-Jiann Chen, Intelligent retrieval of very large Chinese dictionaries with speech queries, In: *EUROSPEECH-1997*, 1767-1770.

[LingvoSoft, 2007] LingvoSoft Inc., Language Translation Software – LingvoSoft – Your Gateway to Language Learning http://www.lingvosoft.com/ (checked on August 27th, 2007)

[McTear, 2004] Michael F. McTear, *Spoken Dialogue Technology: Toward the Conversational User Interface*. Springer-Verlag, 2004.

[Metacritic, 2006] Metacritic, Talkman Critic Reviews http://www.metacritic.com/games/platforms/psp/talkman?q=talkman (checked on August 27th, 2007)

[Microsoft, 2007] Microsoft Corporation, Windows Speech Recognition (Windows Vista) http://www.microsoft.com/windows/products/windowsvista/features/details/speechrecognition.mspx (checked on December 13th, 2007)

[Nuance, 2007a] Nuance Communications, Inc., Nuance - OpenSpeech Recognizer http://www.nuance.com/recognizer/openspeechrecognizer/ (checked on December 13th, 2007)

[Nuance, 2007b] Nuance Communications, Inc., Nuance - RealSpeak - Word http://www.nuance.com/realspeak/word/ (checked on 14th December 2007)

[OptimSys, 2004] OptimSys, OptimTalk Desktop Suite http://www.optimsys.cz/products/desktop-suite/introduction.php (checked on December 20th, 2007)

[Rosenfeld et al., 2007] Roni Rosenfeld, Alexander Rudnicky, Stefanie Tomko, Thomas Harris, Universal Speech Interface project http://www.cs.cmu.edu/~usi/ (checked on September 21st, 2007)

[Rouillard, 2007] Jose Rouillard, Web Services and Speech-based Applications around VoiceXML. *Journal of Networks.* **2** (1), 2007, 27-35.

[Simpson, 2003] Jane Simpson, Representing Information about Words Digitally http://www.paradisec.org.au/Simpson_paper_rev1.pdf (checked on May 18th, 2008)

[SLS Group, 2005] Spoken Language Systems Group MIT Computer Science and Artificial Intelligence Laboratory, Galaxy Architecture http://groups.csail.mit.edu/sls/technologies/galaxy.shtml (checked on September 21st, 2007)

[Turunen and Hakulinen, 2005] Markku Turunen and Jaakko Hakulinen, Design and Development of Speech Interfaces Course Material http://www.cs.uta.fi/hci/spi/ddsi/ (checked on August 27th, 2007).

[Wahlster, 2000] Wolfgang Wahlster, Verbmobil: Foundations of Speech-To-Speech Translation http://books.google.com/books?hl=en&lr=&id=RiT0aAzeudkC&oi=fnd&pg=PR5&dq=Verbmobil:+Foundations+of+Speech-To-Speech+Translation&ots=jBhMwQ0HnT&sig=zx2EWMK4n-_lYhG9k5gKU2zGieE#PPP1,M1 (checked on May 7th, 2008)

[Wikipedia, 2007] Wikipedia The Free Encyclopedia, Talkman http://en.wikipedia.org/wiki/Talkman (checked on August 27th, 2007)

[Zhao, 2002] Yong Zhao, The E-Language Learning Project: Conceptualizing a Web-Based Language Learning System. http://ott.educ.msu.edu/elanguage/about/whitepaper1.pdf (checked on September 21st, 2007)

Appendix 1

Requirements specification template

Document template for Requirements Specification
Timo Poranen, TAY/TKT
Dokumentti luotu 8.10.2006
Updated 8.10.2007

This requirements specification template is slightly modified version of the following documents:

Tero Ahtee, Tampere University of Technology:

* http://www.cs.tut.fi/~projekti/dokumentit/maar-sisalto.txt
* http://www.cs.tut.fi/kurssit/8102500/dokumentit/english-documents.txt

The first mentioned link provides good suggestions to take account when writing the plan (in Finnish).

Good books that have useful material are:

* Kotonya, G. and Sommerville, I.: Requirements engineering, Wiley, 1998
* Haikala, I. ja Märijärvi, J.: Ohjelmistotuotanto, Talentum, 2005.
* Sommerville, I. Software Engineering 7, chapters 6 ja 7, Addison-Wesley, 2004.
* Lauesen,S.: Software requirements, styles and techniques, Addison Wesley, 2001.
* Pressman, R.: Software Engineering, McGrawHill, 2005.

This document is required to be complete if your project apply Waterfall model. In other development models it is ok to leave open questions, if you are going to build prototypes / increments / releases to fill open issues and to get feedback from your client. In that case, it is expected that your partially complete requirements specification must be

reviewed earlier and you (depending on your project plan) give in the same review implementation and test plans for the first increment.

The best way to list usability requirements is to use specific usability analysis document:
http://www.cs.uta.fi/uteam/

-> usability-analysis-template.doc

Section 6.2 .... "availability", is, for instance
what is the maximum amount of time for the system to be out of use.
Should the system be available all the time?

- -
REQUIREMENTS SPECIFICATION

Cover page
Version history
Table of contents
Table of figures (if necessary)
Table of tables (if necessary)
Table of appendixes (if necessary)

1. Introduction
 1.1 purpose and scope
 1.2 product and environment
 1.3 definitions, acronyms and abbreviations
 1.4 references
 1.5 overview

2. General description
 2.1 product perspective
 2.2 product functions
 2.3 user characteristics
 2.4 general constraints
 2.5 assumptions and dependencies

3. Data and databases
 3.1 contents of information
  3.1.x item/aspect x

3.2 intensity of use
3.3 capacity
3.4 file(s) and configuration file(s)

4. Functions (requirements)
menu/screen hierarchy
event list
4.x (every function is an subsection)
user interface
about every function;
purpose
inputs
handling
outputs
error handling

5. Interfaces
5.1 hardware interfaces
5.2 software interfaces
5.3 communications interfaces

6. Other requirements
6.1 performance and response times
6.2 security, recovery, availability
6.3 maintainability
6.4 transferability/portability
6.5 operator's task requirements

7. Design constraints
7.1 standards
7.2 hardware constraints
7.3 software constraints
7.4 other constraints

8. Rejected ideas

9. Ideas for further development

10. Open issues (should not exist anymore in "frozen" document)

Appendixes (compulsory in almost all projects!!!)

- User Interface Plan (containing all screens and forms)
- A list of error messages
- Use Cases
- State Diagram (how you move between screens?)
- Other?

Appendix 2

Usability plan

1. Participants

Test participant 1: (occupation)
(Experience with speech application?)
Test participant 2: (occupation)
(Experience with speech application?)
Test participant 3: (occupation)
(Experience with speech application?)
Test participant 4: (occupation)
(Experience with speech application?)
Test participant 5: (occupation)
(Experience with speech application?)

2. Test tasks

Task 1	**Search for the meaning of "adversely".**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will get the meaning of specified word via speech.
End state	Definition results completely represented.
Estimated task time	Less than 3 minute.

Task 2	**Search for the meaning of "get to grips".**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will get the meaning of specified phrase via speech.
End state	Definition results completely represented.
Estimated task time	Less than 3 minute.

Task 3	**Listen to the definitions again.**
Start state	Options selection prompt.
Rationale	The purpose of this task is to find out if the users will be able to hear the repeated definitions of specified phrase.
End state	Definition results completely represented.
Estimated task time	Less than 2 minute.

Task 4	**Search for the meaning of "schooner".**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will get the meaning of specified word via speech, although the specified word has difficulty in pronunciation.
End state	Definition results completely represented. If the participant does not get the results, the moderator will continue to the next task.
Estimated task time	Less than 5 minute.

Task 5	**Search for the meaning of "confidential".**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will get the meaning of specified word via speech, although the specified word has less difficulty in pronunciation than the previous word has.
End state	Definition results completely represented.
Estimated task time	Less than 4 minute.

Task 6	**(It's just too long definitions.) Stop the system and look up a new word definition.**
Start state	Definitions are being represented.
Rationale	The purpose of this task is to find out if the users will be able to quit the system on purpose and search for a new word definition.
End state	System first prompt. If the participant cannot stop the system, the moderator can help by giving an instruction.
Estimated task time	Less than 2 minute.

Task 7	**Search for the meaning of “profane”.**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will notice that the specified word is unrecognized by the system. So the users should then try to look up another word.
End state	System guides the users to look up a new word definition.
Estimated task time	Less than 3 minute.

Task 8	**Search for the meaning of “zill”.**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will get the meaning of specified word via speech.
End state	Definition results completely represented.
Estimated task time	Less than 3 minute.

Task 9	**Search for the meaning of “whereas”.**
Start state	System first prompt.
Rationale	The purpose of this task is to find out if the users will be informed that the specified word has no definitions given by Wordnet 2.0 dictionary.
End state	System informs the users that “no definitions found in Wordnet 2.0 dictionary”.
Estimated task time	Less than 3 minute.

Task 10	**Exit the system.**
Start state	Options selection prompt.
Rationale	The purpose of this task is to find out if the users will be able to exit the system.
End state	System shuts down.
Estimated task time	Less than 30 seconds.

3. Questionnaires

	1	2	3	4	5
It was easy to complete a task using the system.					
It was easy to navigate around the system.					
The system understood what you said.					
The system's speech was easy to understand.					
The system responded in a timely manner.					
The system responded in ways that you would expect.					
The system help was properly provided.					
The system was able to cope with errors.					
You would prefer to use this system rather than a dictionary system with text inputting.					

1. Which parts of the system should be improved?
   Ans:

2. Which functions do you think they are useful for the future development?
   Ans: