Information retrieval system using SMS

Shreyas M.Patankar
P.V.G.'s C.O.E.T.
Pune-411009
Maharashtra.

Rajkalpesh R.Jaiswal
P.V.G.'s C.O.E.T.
Pune-411009
Maharashtra.

Sunil R.Sansare
P.V.G.'s C.O.E.T.
Pune-411009
Maharashtra.

ABSTRACT
A natural language dialogue system is composed of many parts: natural language process, dialogue management, data query, etc. Adding a new service function to traditional complex dialogue systems, which is based on a mixed initiative paradigm, for example Question Answering (QA) service for E-commerce website, requires in general a big effort for redesigning dialogue management modules and other important modules. It can balance between overall dialogue management and particular agent service. Active people often do not have the time or ability to study in traditional, face-to-face classroom settings. In addition, learners may need access to content or expertise as they go about their daily activities. Mobile learning with cell phones offers a unique response to this need. Mobile phones can support many kinds of learning, including items of glossary, language learning tips, examination preparation notes, short course summaries, answers to exercises, etc. We discuss the advantages brought by the new architecture, including more extensibility, more effective, platform insensitivity, wider adaptability etc., and describe in de- tail how to implement the system modules. A system for querying information and knowledge by the use of SMS in a mobile learning environment is considered. Our system is applicable for various kinds of organizations such as schools, colleges, companies, etc. The main concept behind this project is to design software that can retrieve requested data from our own knowledge base and send it to user.

General Terms
Theme recognition algorithm, frame based dialogue control algorithm

Keywords
SMS, natural language dialogue system, natural language process, dialogue management, data query, complex dialogue systems, knowledge base

1. INTRODUCTION
The popularity of mobile learning is growing along with the rapid increase in use of personal mobile devices and wireless networks[3] .There are three main types of mobile learning:

a) Collaborative learning in a wireless environment,

b) Utilization of mobile devices for mentoring and scaffolding purposes and,

c) Distribution of learning materials, such as test questions, to mobile terminals. Self-learning environment in which mobile phones and personal computers are used to complement each other. There have been several studies of self-learning environments that use mobile phones. However, these studies either focus only on mobile phones as the client device and not on access from personal computers [3], focus on mobile phones used only to receive simple mentoring messages, the learning material is often developed with proprietary specifications, preventing content from being shared and reused. This paper describes a self-learning system that supports both mobile phones and personal computers as client devices. The system has a learner adaptation capability spanning both mobile phone-based and personal computer-based learning phone-based and computer-based learning environments by enabling sharing of learning material. The design goals of the system are:

1) To provide a standard-based mobile learning infrastructure independent of device characteristics (often different from mobile phone to mobile phone or from carrier to carrier) using existing e-learning standards [3].

2) To enable offline learning using mobile phones.

3) To implement a learner adaptation functionality with which learning materials and learners status from mobile phones and personal computers are shared and the learners status is reflected in the next learning activity from both environments. Mobile phones are already becoming much more than devices for transmitting the human data wirelessly.

QA is defined to be a system for discovering answers to an open-domain natural language question from a large repository. Along with the popularization of Internet, World Wide Web becomes a very useful knowledge base for general purpose QA systems [1].

Many companies provide online services for querying and question answering. Since QA systems can answer peoples question automatically, even in natural language way with a natural language process (NLP) component, we could apply QA system to E-commerce and other QA service. If we only construct domain-specified QA system for each of the services, most of the QA systems have a similar framework: they generate a query out of the natural language question, perform various query actions and then pinpoint the exact answer. But because different service programs may have different data structures, query parameters and even different platforms, systems vary in these three aspects.

In this paper, we focus on multi-theme question answering system, which is a combination of QA systems added with theme recognition and theme management.
Services of many E-commerce websites vary from time to time. In other words, programs are added or removed frequently. If we use traditional systems, it will be awkward when we plan to add a new service, which means

to develop an entire new QA system for the specified usage. But in our system, because of the existence of dialogue and theme management module, new services only need to register in it and provide some theme information, and then the new service will be available. When the server receives a query request, it will recognize its theme and then send the query command to query processing module. Query processing module can do data processing themselves and then send the result back.

Mobile device also need QA systems of this architecture. In modern society, mobile devices become more and more popular. PDA, cell phone and pocket PC appear frequently in people's hands. A traditional QA system needs a huge database including word dictionary, rules and other data, which makes it impossible to install a QA system on a mobile device because of its small storage capability. But in our system, thanks to the application of Web Services, a mobile device doesn't need to store all the data in its memory.

The rest of this paper is organized as follows. Design policies are discussed in Section 2, Overview of the system is described in section 3, Design of the system is described in section 4, System implementation is described in Section 5, Conclusion and future work is described in section 6.

2. DESIGN POLICIES

The designing goals are:

1. To understand the purpose and goal of system [5].

2. Differentiate the whole system according to its functionality. There are following design considerations:-

(a)System design

In this first level it is decided that which modules are needed for the system, the specification of these modules, and how these modules should be interconnected [6].

(b)Detailed design

In the second level, the internal design of the modules, or how the specification of the module can be satisfied is decided.

To meet the design goal, which is to enable offline learning using mobile phones, an offline content browsing function for mobile phones must be provided. This browsing function should be capable of displaying as much content as possible. For tests, it should be able to accept answers, score them, and give feedback without communicating with the mobile/server.

3. OVERVIEW OF THE SYSTEM

The system takes in a natural language (NL) question in English from the user. This question is then passed to a query processing module, which includes a Part-of-Speech (POS) tagger and a grammar analysis module. Dialog control module is the core of the system, which cooperates with other modules, including dialogue theme recognition, semantic recognition, data extraction and an interface to access services.

4. DESIGN OF THE SYSTEM

The system consists of a GSM Module (GSMM), a Dialogue Control Module (DCM), a Querying Processing Module (QPM) and a Knowledge Base (KB), as depicted in Fig. 1. By receiving from and sending to users messages through the GSM network, GSMM acts as the interface between the mobile infrastructure and the rest of the system. The processed requests are then transferred to DCM, which handles the content of the requesting messages and produces suitable querying tasks for QPM, which in turn searches and matches information from KB or Internet to produce suitable answering messages for the users according to their requesting messages[4].

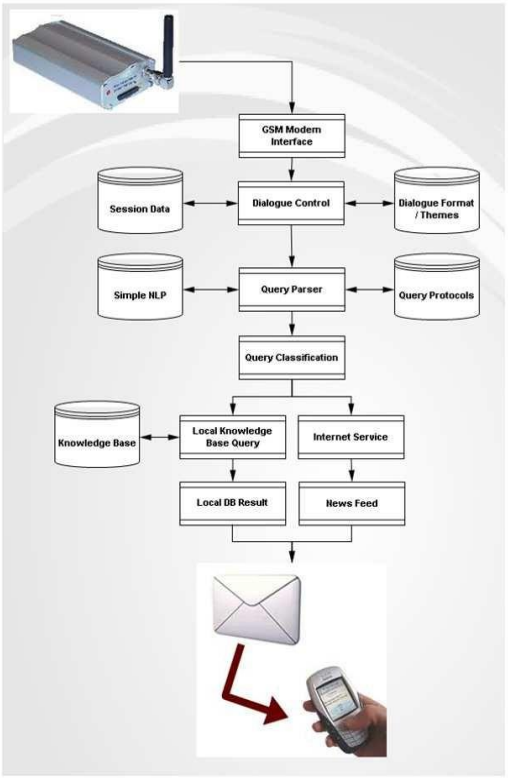

Fig 1: Architecture of the system

5. SYSTEM IMPLEMENTATION

5.1 Query processing module

The natural language query given by the user is first tagged by a Part-of-Speech tagger. In a sentence, a words Part-of- Speech correlates to its position in the sentence. For example, an adjective always appears before a noun. We can use Hidden Markov Model (HMM) to describe the Part-of-Speech possibilities [1].WordNet is useful for this step, because it includes almost all of the words possible Part-of-Speech.

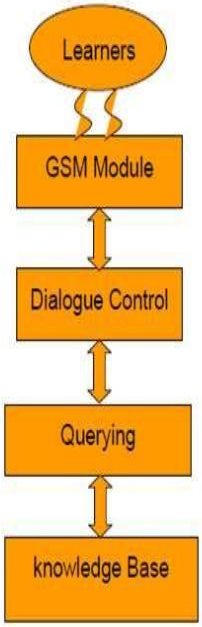

Fig 2: System structure

Fig 3: The flowchart of dialogue control and NLP

5.3 Semantic recognition and data extraction module

This module extracts data information from user input. It is considered that most of the information is contained in verb, noun and adverb. The analysis of these words is the most important step for semantic recognition.

Given a system question Which software do you want to download?, there are two possible ways to answer:

After a sentence has been tagged, the result will be processed by a syntax analysis module. Syntax analysis is defined to be the process to find the grammar tree for tagged sentence. Chart Parsing is adopted to generate the grammar tree.

5.2 Dialogue theme recognition module

For every registered domain, a set of describing words is predefined. For example, if user input includes following words, then he may have the intention to use employment consultation: employ, job opening etc. Similarly, when people want to download some software, they will talk about the words software, download etc.

What is also possible is that several describe words have been contained in a users sentence. Given the sentence I want to get some employment information about software developing, employment and software are both contained. So the system should confirm the user's intention, by asking do you want employment consultation or software download service? And then the user will give a more clear answer. The algorithm could be described as following:

The algorithm starts from the first registered theme T_1, and then for each registered theme T_t, looks over its describing words dictionary. If any of the describing words w is substring of user's input I, T_t will be added to the possible theme set Theme. After the loop, if the number of themes in Theme is only one, then the theme is the intention theme, or else a further confirmation will be taken to make sure user's intention by call Confirm (Theme).

1. $T_t = T_1$
2. while $t \le sizeof(registered\ \ themes)$
3. {
4. for each describe word w of T_t
5. if w is substring of I
6. add T_t to $Theme$ and break
7. $t = t+1$
8. }
9. if $sizeof(Theme)=1$ return $Theme$
10. else return $Confirm(Theme)$

1) Use only one word, like Eclipse or office. In this case, the word will be the value of asking question simply.

2) Use a complete sentence, like the software name is Eclipse. In this case, both the value phrase Eclipse and the description phrase software name should be recognized.

5.4 Dialogue control module

This module is the core part of the system, which manages all the registered theme and control dialogue flow. In the inter-communication between user and system, this module controls all other modules to recognize user intention, dialogue theme, extract data etc.

The control process could be summarized by the following algorithm:

It is totally different comparing to traditional dialogue pro- cess algorithm. $_{At}$ first, dialogue theme is recognized as S_i according to user input I. Then it starts in the initial state S_{i1} of theme S_i. S_t denotes the system state at turn t. The function NextAction determines the next action A_t to be invoked, and the function NextState updates the state variables with the external observations. After that, a new user input I_t should be analyzed to recognize its intention S_j. If S_i is not the same as S_j, then system state should be updated to the initial state S_{j1} of theme S_i. The process is repeated until a final state S_F is reached.

1. $S_j = ThemeRecognize(I)$
2. $S_t = S_{i1}$
3. while $S_t \neq S_F$
4. {
5. $A_t = NextAction(S_t)$
6. invoke A_t
7. O_t = environment response to A_t
8. $S_{t+1} = NextState(S_t, A_t, O_t)$
9. $t = t+1$
10. $S_j = ThemeRecognize(I_t)$
11. if $S_t \neq S_j$ then $S_t = S_{j1}$
12. }

5.5 User interface

This module is responsible for exchanging information be- tween users and the system. Since we didn't include a speech recognizer, graphic user interface (GUI) is adopted as the interface.

6. CONCLUSION AND FUTURE WORK

Mobile learning is just beginning. Only now are technologies emerging that will support basic learning applications, but even these technologies offer the ability to develop meaningful, effective learning objects for language learning. New technologies will support learning applications of greater sophistication. By drawing on previous experience, integrating effective, research-based methods, and melding those with the emerging capabilities of mobile phones, the possibility exists of developing compelling mobile learning content to meet the needs of language learners worldwide.

The development of new features of this technology should be guided of principles of good curriculum design and pedagogy for teaching English. New features need to support needs for:

1. Greater interactivity with the content, through the ability to submit student responses [2].

2. Access to teachers, librarians, and other learners.

3. Ability to interact with other learners, including playing games, conversation, and project-based learning, preferably using the phones capabilities

Because of its natural human-computer interaction style, question answering will be applied in more and more fields in future.

7. ACKNOWLEDGEMENT

As per the current scenario in day-to-day life, students require some small device oriented architecture for learning in less time to get the requested information. So, we tried to give students this facility in affordable cost.

8. REFERENCES

[1] Chen, Z., Wen, D. (2005).”A New Web-Service-Based Architecture for Question Answering.” *Proceedings of IEEE International Conference on Natural Language Processing and Knowledge Engineering* (IEEE NLP-KE05), 179-183, Nov.2005

[2] Collins, T. G. (2005). ”English Class on the Air: Mobile Language Learning with Cell Phones.” *Proceedings of the Fifth IEEE International Conference on Advanced Learning Technologies* 2005

[3] K.Nakabayashi, T.Hoshide, M.Hosokawa, T.Kawakami, K.Sato,”Design and Implementation of a Mobile Learning Environment as an Extension of SCORM 2004 Specifications,” *IEEE International Conference on Advanced Learning Technologies* 2007.

[4] Stone, A. (2004).”Mobile Scaffolding: An Experiment in Using SMS Text Messaging to Support First Year University Students.” *Proceedings of the IEEE International Conference on Advanced Learning Technologies 2004.*

[5] McTEAR, M. F. (2002). ”Spoken Dialogue Technology: Enabling the conversational user interface.” *ACM Computing Surveys*, 34(1), 90-169

[6] Wang, C.-Y., Liu, B.-J. et al. (2003). ”Using Mobile Techniques in Improving Information Awareness to Promote Learning Performance.” *Proceedings of the The 3rd IEEE International Conference on Advanced Learning Technologies* (ICALT’03)