Speech Recognition System Using DHMMs Based on Ubiquitous Environment

Jong-Hun Kim[1], Un-Gu Kang[2], Kee-Wook Rim[3], and Jung-Hyun Lee[1]

[1] Department of Computer Science & Engineering Inha University
Yonghyun-dong, Nam-gu, Incheon, Korea
jhkim@hci.inha.ac.kr, jhlee@inha.ac.kr

[2] Department of Information Technology Gachon University of Medicine and Science
Yeonsu-dong, Yeonsu-ku, Incheon, Korea
ugkang@gachon.ac.kr

[3] Department of Computer and Information Science
Sun Moon University, Chung-Nam, Korea
rim@sunmoon.ac.kr

Abstract. Most commercialized speech recognition systems that have a large capacity and high recognition rates are a type of speaker dependent isolated word recognition systems. In order to extend the scope of recognition, it is necessary to increase the number of words that are to be searched. However, it shows a problem that exhibits a decrease in the system performance according to the increase in the number of words. This paper defines the context information that affects speech recognition in a ubiquitous environment to solve such a problem and designs a new speech recognition system that demonstrates better performances than the existing system by establishing a word model domain of a speech recognition system.

1 Introduction

The necessity of the interface between humans and machines according to the development of information and communication technologies is required. In particular, speech recognition technologies are necessary to satisfy the natural communication between various devices in ubiquitous environments and most easy interfaces. These speech recognition technologies extract the linguistic information and sound information included in the voice between humans and save the extracted information to transfer it to machine by applying proper practices in order to understand the meaning included in this information.

Types in speech recognition have been developed as an isolated word recognition method that recognizes separately spoken words, continuous pronunciation recognition method that recognizes continuous pronunciations, and voice understanding that recognizes conversational sounds. The final goal of these speech recognition technologies is to understand every voice in all environments. However, the high performance commercial system is mainly represented as a speaker dependant isolated word system.

R. Obermaisser et al. (Eds.): SEUS 2007, LNCS 4761, pp. 213–222, 2007.

Representative commercial speech recognition systems are Voice Scribe 1000 Dragon Dictate by Dragon System, Voice Command by IBM, and Speech Command by Texas Instruments. These systems are speaker dependant isolated word recognitions systems and able to recognize about 1000 words. These systems show certain significant decreases in their performances, such as recognition speed and rate, according to the increase in words.

Therefore, this paper attempts to design a speech recognition system (SRS) using user's context information in speech recognition services. The factors that affect the performance of a speech recognition system will be configured as context information and determined using Ontology. Information can be obtained using the noise measurement, Radio Frequency Identification (RFID) Tag, and RFID Reader, and then accurate context information can be recognized using Ontology Database and inference engine. The system proposed in this study is designed based on Open Service Gateway Initiative (OSGi), which is a type of ubiquitous middlewares, in order to obtain real-time context information and provide the obtained information to applications. Also, the speech recognition algorithm used in this system is a Hidden Markov Models (HMMs) in which this algorithm overcomes the disadvantage in an isolated word speech recognition system and increases the performance by configuring a HMM Domain according to the context. In the results of the test for the user who registered in the system, it showed a high speech recognition rate in a home network system.

2 OSGi

Open Service Gateway Initiative (OSGi) is an organization that establishes standards on the transmission of multi-services that independently home networks and information domestic appliances through access networks by defining network technology and common open architecture structures. OSGi was founded on March 1999, consisted of 15 businesses, and was expanded to include more than 50 software, hardware, and service provider companies.

OSGi is a nonprofit organization that not only defines the API between middlewares and application programs but also plays a role in the separation between specified application programs and middlewares. Standards established by OSGi provide dynamic services for devices with small capacity memories using the platform independence of Java and network mobility of execution codes. In particular, it is an open architecture network technology that can support various network techniques, such as Bluetooth, Home Audio/Video Interoperability (HAVi), Home Phoneline Networking Alliance (PNA), Home Radio Frequency (RF), Universal Serial Bus (USB), Video Electronics Standards Association (VESA), and other networks. It also provides management and connection functions for most products. These include set-top boxes, cable modems, routers, warning systems, power management systems, domestic appliances, and PCs, in which the Java based gateway consists of Java environments, service frameworks, device access management functions, and log services that include the connection technology for these elements when access and new services are required. The OSGi service platform displayed in Fig. 1 consists of the OSGi framework and standard services.

Three major entities of the OSGi are Service, Bundle, and Framework. Service includes Java interfaces that perform specific functions, actually implements objects, and is a component that is accessed through a predefined service interface. A single application can be configured through the cooperation of several services and is able to request services during run-time. Bundle is a functional distribution unit that provides services. Framework is an execution environment that manages the life cycle of the Bundle. Bundle is a service set and a component unit that uses the service registered in service registries. The implementation of Service can be performed physically, distributed, and sent to the Framework through the Bundle in logical units. Bundle exists as JAR files. A JAR file includes more than one service implementation object, resource files, and manifest files. The manifest file represents the service provided by each Bundle and other services that are used to implement Bundle. Finally, Bundle can be implemented or terminated using the Start and Stop function in the Framework.

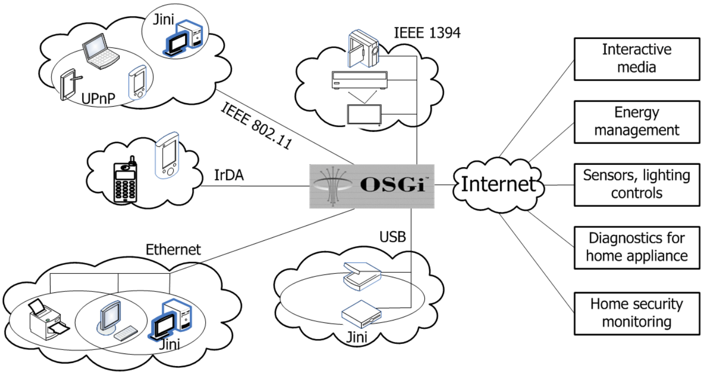


Fig. 1. The Overview of OSGi

3 Domain-Separated Hidden Markov Models (DHMMs)

A pattern recognition method is generally used for speech recognition and is classified as a Dynamic Time Warping (DTW) method that uses a template-based pattern matching and Hidden Markov Model (HMM) method employing a statistical pattern recognition method. HMM is an algorithm that was founded on mathematics. It was introduced in the field of speech signal processing in 1975 and widely applied from isolated word recognition to the spontaneous speech recognition. This algorithm can be classified as a learning and recognition process under the assumption that the time series pattern in speech feature vectors is modeled after the Markov process. In addition, a method that is combined with the HMM is widely used at the present time due to the increase in the amount of calculations even though neural network based methods are also used in speech recognition.

This study applies an HMM that uses a speech recognition algorithm as a pattern recognition method according to the domain. The Baum-welch method is used as a learning method for the HMM. In addition, probability for the HMM is calculated using a Vitervi algorithm.

3.1 HMM Topology

The parameters used in the HMM consist of the transition probability between states, output probability subordinated to states, and initial presence probability of states. The parameter of the HMM can be simply expressed as Eq. (1).

$$\lambda = < A, B, \pi > \tag{1}$$

A : State-transition probability distribution

B : Observation symbol probability distribution

π : Initial state distribution

This study applied a modified Bakis model that included five different states as illustrated in Fig. 2 in order to express Eq. (1). The major characteristics of the HMM model topology can be determined as five STATEs, First-Order-Markov Chain model, and self migration potential and next state potential probability for each STATE. The last STATE is a DUMMY STATE that has no transition probability in which each STATE has 512 OBSERVATION SYMBOL probabilities.

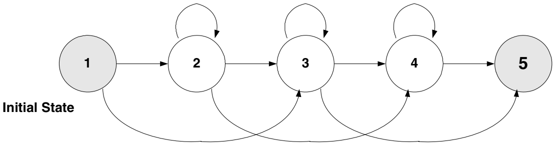
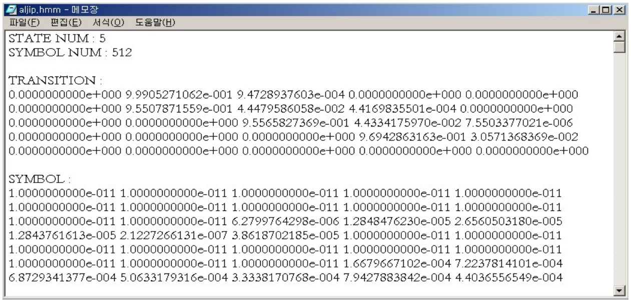


Fig. 2. HMM Topology

Fig. 3 illustrates the internal probability of the "aljip" HMM, a type of computer application words.



```
STATE NUM : 5
SYMBOL NUM : 512

TRANSITION :
0.0000000000e+000 9.9905271062e-001 9.4728937603e-004 0.0000000000e+000 0.0000000000e+000
0.0000000000e+000 9.5507871559e-001 4.4479586058e-002 4.4169835501e-004 0.0000000000e+000
0.0000000000e+000 0.0000000000e+000 9.5565827369e-001 4.4334175970e-002 7.5503377021e-006
0.0000000000e+000 0.0000000000e+000 0.0000000000e+000 9.6942863163e-001 3.0571368369e-002
0.0000000000e+000 0.0000000000e+000 0.0000000000e+000 0.0000000000e+000 0.0000000000e+000

SYMBOL :
1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.0000000000e-011
1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.0000000000e-011
1.0000000000e-011 1.0000000000e-011 6.2799764298e-006 1.2848476230e-005 2.6560503180e-005
1.2843761613e-005 2.1227266131e-007 3.8618702185e-005 1.0000000000e-011 1.0000000000e-011
1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.0000000000e-011
1.0000000000e-011 1.0000000000e-011 1.0000000000e-011 1.6679667102e-004 7.2237814101e-004
6.8729341377e-004 5.0633179316e-004 3.3338170768e-004 7.9427883842e-004 4.4036556549e-004
```

Fig. 3. Five States and 512 Observation Symbols of HMM

3.2 HMM Domain

3.2.1 Context Information Domain

This study configured the HMM domain according to speaker information, utterance location, and used objects. Speaker information and location were verified using RFID sensors installed in user devices and in the home. The state information domain was configured according to the space (Balcony, Bathroom, Bedroom, Guestroom, Kitchen) in the home where detailed configurations were performed in accordance with applied objects (Computer, Television, Radio, Refrigerator, Washing machine, Electric Lamp). State information on the speaker and noise in the Context Manager and Service Manager were transmitted to the Speech Recognition Manager.

3.2.2 Observation Sequence Domain

The speech recognition prototype implemented in this study was based on a word model. The length of the observation sequence produced by the length of the utterance changed because the word model shows the same utterance unit and recognition unit. Therefore, the recognition rate decreased in the application that had HMM topology similar to the isolated word that registered different utterance lengths. Furthermore, recognition speed exhibited a decrease due to the increase in the number of recognition words. The system used in this study configured the domain according to the length of the observation sequence and selected the domain from the HMM of the objective domain using the length information of the observation sequence from the input speech signal.

3.2.3 Syllable Number Domain

This study configured a state information domain, observation sequence domain, and domain for the number of syllables in order to improve the performance of the speech recognition system. Vowels in speech showed a periodical property different from the consonants. Thus, it is possible to improve the performance of the speech recognition system using the domain for the number of syllables through a reliable detection process for vowels. In addition, it is possible to develop a speech recognition system according to the unit of phonemes. The number of syllables can be produced by the analysis of the frequency of isolated words and formant feature extraction data.

4 Speech Recognition System Design

This chapter designed and implemented the speech recognition system (SRS) that was able to recognize correct speech by estimating context information in a Java-based OSGi framework using the context definition.

Fig. 4 presents the diagram of the overall system. The SRS designed in this paper analyzed and suggested various data transferred from context recognition sensors and established it as information to recognize correct speech through a recognition process. In order to perform this process, the SRS consisted of a Context Manager, Service Manager, and Speech Recognition Manager.

The system proposed in this study used an ontology inferencer Jena 2.0 and developed an OSGi gateway using the Knopflerfish 1.3.3, an open architecture source project which implemented a service framework.

4.1 Context Manager

The configuration of context information for the speech recognition system (SRS) consists of user information (sex, age), noise, object, and location information.

User information, nosie, object and location information can be predefined as ontology, and data can be input from sensors. Noise data can be transferred using an OSGi framework and communication from a noise measurement device used in real-time Zigbee communication. User information, use object and location information can be traced using an RFID Tag which is attached to a user.

Table 1 presents the definition of context information as different spaces, such as class 2 for sex, class 5 for age, class 3 for noise, class 6 for object and class 6 for location information, in order to build an ontology model.

In particular, the service area is limited to homes, and the users' location is limited to the Balcony, Bathroom, Bedroom, Guestroom, Kitchen, and Livingroom.

Table 1. Configuration and Definition of Context Information

Sex	Age		Noise		Object	Location
class	**num.**	**class**	**Num.(dB)**	**class**	**class**	**class**
Male	0~7	Infant	20~39	Low	Computer Television Radio Refrigerator Washing machine Electric Lamp	Balcony Bathroom Bedroom Guestroom Kitchen Livingroom
	8~11	Child				
	12~17	Young Adult	40~59	Normal		
Female	18~61	Adult				
	62~	Old Adult	60~	High		

The context of the SRS based on the context information used in this study is defined as Web Ontology Language (OWL) that is used on a Semantic Web in order to configure and express exact contexts and various relationships.

The Context Manager transfered data generated by events to a context analyzer and that data was transfered to an OWL inference engine. The OWL inference engine transferred data received from the context manager to the Service Manager in which data was transformed as information using an OWL inferencer including OWL ontology object database.

4.2 Service Manager

The Service Manager consisted of a Bundle Service that provided speech recognition service as a bundle in a Simple Object Access Protocol (SOAP) Service, OSGi framework installed device in order to transfer information received from the OWL inference engine to the SRS, and an Application and Bundle Manager Service that supported the management of the mobility of bundles.

4.3 Speech Recognition Manager

A speech recognition manager extracts observation sequences from the feature extracted voice data and produces the optimum state sequence and probability value by applying a Viterbi algorithm in the HMM. The HMM that has the largest probability value in such obtained probability values will be applied to recognize voices. This system improves the search speed and recognition rate of the HMM according to the position that is the context information of HMMs and applied objects.

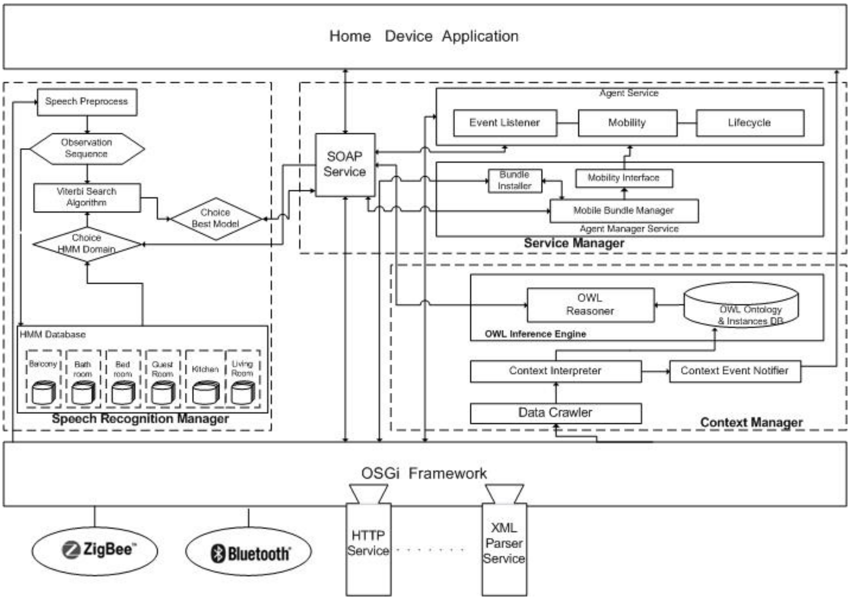
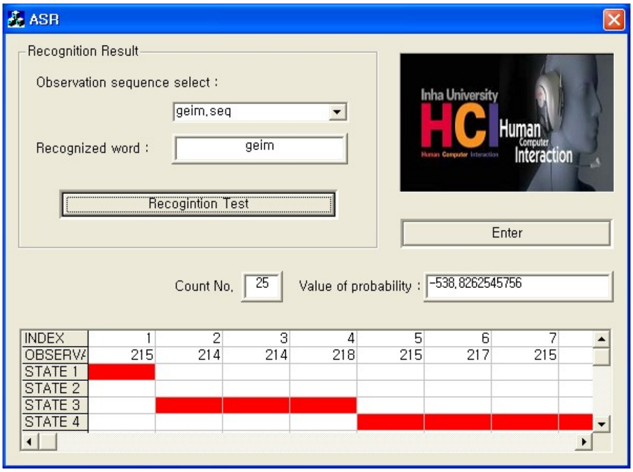


Fig. 4. The Speech Recognition System Using Context Information

5 System Evaluation

In order to test the efficiency of the speech recognition system proposed in this paper, the test was applied using 50 words that were usually used to control computers and electronic appliances and recorded in a normal housed hold by three speakers. The data was sampled by 16kHz and transferred as 16bits using an A/D converter.

The accuracy of the HMM and recognition algorithm was tested on 25 words used in computer applications. Fig. 5 shows the selection of the word that exhibited the highest probability among 25 sample words by applying an observation sequence from the observation sequence used to test the HMM. Fig. 6 illustrates the difference in recognition rates of the conventional Hidden Markov Models (HMMs) and Domain-separated Hidden Markov Models (DHMMs).



Recognition Result

Observation sequence select : geim.seq

Recognized word : geim

Count No. 25 Value of probability : -538.8262545756

INDEX	1	2	3	4	5	6	7
OBSERV/	215	214	214	218	215	217	215
STATE 1	🟥						
STATE 2							
STATE 3		🟥	🟥	🟥			
STATE 4					🟥	🟥	🟥

Fig. 5. Model Recognition Application for Selection of the HMMs

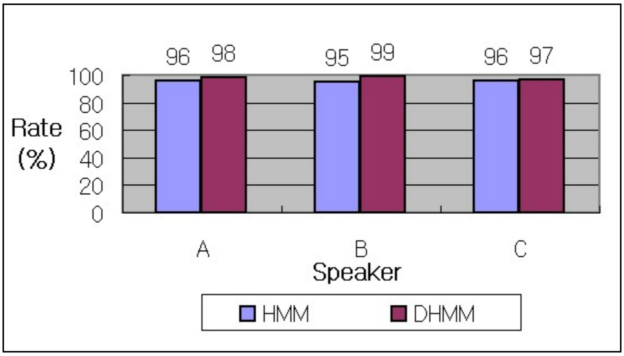


Fig. 6. Speech Recognition Rate of HMM and DHMM

6 Conclusions

Commercial isolated word speech recognition systems used to control the existing application or electronic appliances demonstrate high recognition rates in a limited environment because these systems use only speaker's voices. This was due to the fact that it can't identify the state of utterances and utterance goals of the user. In addition, the searching time of word models will increase, if the number of subject words increased due to the use of a word model. Also, it represents a low recognition rate due to the increase in the number of words.

This study obtains personal information of utterances in a ubiquitous environment and designs a speech recognition system that improves the performance of such a system by investigating utterance goals through the position and applied device. Thus, the context information of utterances was configured in accordance with sex, age, noise, applied object, and position and defined as Ontology. This system configured a word model domain according to the position and object in order to recognize proper information for the applied context information. The actively obtained context information in an OSGi based context recognition manager becomes important information in the selection of a word model domain to recognize voices. In the results of the performance test for the system proposed in this study, it demonstrated a high recognition rate in all positions in a home network environment.

It is necessary to precisely model the given context using various sensors in order to accurately verify the intention of utterances and apply it to a voice recognition system in future. Also, it is necessary to develop a system that has no limitations in noise environments, sex, and age by adding a model, which defines noises, sex, and age.

Acknowledgement

This research was supported by the Ministry of Information and Communication (MIC), Korea, under the Information Technology Research Center (ITRC) support program supervised by the Institute of Information Technology Assessment (IITA).

References

1. Dobrev, P., Famolari, D., Kurzke, C., Miller, B.A.: Device and Service Discovery in Home Networks with OSG. IEEE Communications Magazine 40(8), 86–92 (2002)
2. Rabiner, L.R.: A Tutorial on Hidden Markov Models and Selected Application in Speech Recognition. Proc., IEEE, 77(2), 257–286 (1989)
3. Rabiner, L.R., Levinson, S.E., Sondhi, M.M: On the Application of Vector Quantization and Hidden Markov Models to Speaker Independent, Isolated Word Recognition. The Bell System technical Jounal 62(4) (1983)
4. Weiser, M.: The Computer for the Twenty-first Century. Scientific American 265(3), 94–104 (1991)
5. Brown, P.J., Bovey, J.D., Chen, X.: Context-Aware Application: From the Laboratory to the Marketplace. IEEE Personal Communication, 58–64 (1997)
6. Sohrabi, K., Gao, J., Ailawadhi, V., Pottie, G.: A Self-organizing Sensor Network. In: the Proceedings of the 37 Allerton Conference on Communication, Control, and Computing, Monticello, Illinois (September 1999)
7. Bellavista, P., Corradi, A., Stefanelli, C.: Mobile Agent Middleware for Mobile Computing. IEEE Computer 34(3) (March 2001)
8. Liu, T., Martonosi, M.: Impala: A Middleware System for Managing Autonomic, Parallel Sensor Systems. In: ACM SIGPLAN Symp. Principles and Practice of Parallel Programming (June 2003)
9. Romer, K., Schoch, T., Mattern, F., Dubendorfer, T.: Smart Identification Frameworks for Ubiquitous Computing Application. In: IEEE International Conference on Pervasive Computing and Communication, IEEE Computer Society Press, Los Alamitos (2003)
10. W3C. Web Ontology Language, http://www.w3.org/2004/OWL/
11. Strang, T., Linnhoff-Popien, C.: A Context Modeling Survey. In: UbiComp 1at International Workshop on Advanced Context Modelling, Reasoning and Management, Nottingham, pp. 34–41 (2004)

12. Chen, H., Finin, T.: An Ontology for Context-aware Pervasive Computing Environments. The Knowledge Engineering Review archive 18(3), 197–207 (2003)
13. Chen, H.: An Intelligent Broker Architecture for Pervasive Context-aware Systems. PhD thesis, University of Maryland, Baltimore County (2004)
14. Rodriuez, M., Favela, J.: A Framework for Supporting Autonomous Agents in Ubiquitous Computing Environments. In: CICESE, Ensenada, Mexico (2002)
15. Carroll, J.J., Reynolds, D.: Jena: Implementing the Semantic Web. Recommendations HP Labs, Bristol UK (2005)
16. JADE, Jave Agent Development Framework, http://jade.tilab.com/
17. Want, R., Hopper, A., Falcao, V., Gibbons, J.: The Active Badge Location System. ACM Transactions on Information Systems 10, 91–102 (1992)
18. Gu, T., Pung, H.K., Zhang, D.Q.: An Ontology-based Context Model in Intelligent Environments. In: Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference, pp. 270–275 (2004)
19. Bagci, F., Schick, H., Petzold, J., Trumler, W., Ungerer, T.: Support of Reflective Mobile Agents in a Smart Office Environment. In: Proceedings of the 18th International Conference on Architecture of Computing Systems, pp. 79–92 (2005)
20. Dermatas, E., Fakotakis, N., Kokkinakis, G.: Fast Endpoint Detection Algorithm for Isolated Word Recognition in Office Environment. In: Proc., ICASSP-91, Toronto (April 1991)
21. Lee, S., Lee, S., Lim, K., Lee, J.: The Design of Webservices Framework Support Ontology Based Dynamic Service Composition. In: Lee, G.G., Yamada, A., Meng, H., Myaeng, S.-H. (eds.) AIRS 2005. LNCS, vol. 3689, pp. 721–726. Springer, Heidelberg (2005)