A Hierarchical Domain Model-Based Multi-Domain Selection Framework for Multi-Domain Dialog Systems

Seonghan Ryu[1] *Donghyeon Lee*[1] *Injae Lee*[1] *Sangdo Han*[1] *Gary Geunbae Lee*[1]
Myungjae Kim[2] *Kyungduk Kim*[2]
(1) Pohang University of Science and Technology
(2) Samsung Electronics
{ryush,semko,lij1984,hansd,gblee}@postech.ac.kr
{koong.kim,kduk.kim}@samsung.com

ABSTRACT

We proposed a hierarchical domain model (HDM)-based multi-domain selection framework (MDSF) for multi-domain dialog systems. The HDM-based MDSF statistically detects one or more candidate domains and heuristically determines one or more final domains from among the candidate domains. The HDM is used in both the candidate domain detection and final domain determination components. Multi-domain dialog systems that employ the HDM-based MDSF provide service to one or more domains at the same time, whereas traditional multi-domain dialog systems provide service to only one domain at a time. To validate the HDM-based MDSF, we developed a multi-domain dialog system for TV program, video-on-demand, and TV device domains. The experimental results show that the HDM-based MDSF correctly selects one or more domains and enables multi-domain dialog systems to provide more accurate and rapid dialog service than traditional multi-domain dialog systems.

TITLE AND ABSTRACT IN KOREAN

다중 도메인 대화 시스템을 위한 계층적 도메인 모델 기반의 다중 도메인 선택 프레임워크

본 논문은 다중 도메인 대화 시스템을 위한 계층적 도메인 모델 기반의 다중 도메인 선택 프레임워크를 제안한다. 계층적 도메인 모델 기반의 다중 도메인 선택 프레임워크는 통계적 방법으로 한 개 이상의 후보 도메인을 검출하고, 규칙으로 후보 도메인 중 한 개 이상의 최종 도메인을 결정한다. 후보 도메인 검출 및 최종 도메인 결정 단계에서 계층적 도메인 모델이 사용된다. 기존의 다중 도메인 대화 시스템이 한 번에 한 개의 도메인에 대한 서비스만을 제공하는 반면, 계층적 도메인 모델 기반의 다중 도메인 선택 프레임워크를 적용한 다중 도메인 대화 시스템은 한 번에 한 개 이상의 도메인에 대한 서비스를 제공한다. TV 프로그램, 주문형 비디오, TV 장치에 대한 다중 도메인 대화 시스템에 대한 실험을 통해 계층적 도메인 모델 기반의 다중 도메인 선택 프레임워크는 한 번에 한 개 이상의 도메인을 정확하게 선택할 수 있고, 다중 도메인 대화 시스템이 기존 다중 도메인 대화 시스템에 비해 정확하고 신속한 대화 서비스를 제공할 수 있게 함을 확인할 수 있었다.

KEYWORDS : Multi-domain dialog system; Multi-domain selection; Hierarchical domain model; Candidate domain detection; Final domain determination
KEYWORDS IN KOREAN : 다중 도메인 대화 시스템; 다중 도메인 선택; 계층적 도메인 모델; 후보 도메인 검출; 최종 도메인 결정

Proceedings of COLING 2012: Posters, pages 1049–1058,
COLING 2012, Mumbai, December 2012.

1 Introduction

A dialog system is a natural and effective interface between humans and machines because dialog is a natural method of human communication. Recently, multi-domain dialog systems that provide service to multiple domains have become widely employed in real-life situations (Allen et al., 2000; Komatani et al., 2006; Larsson and Ericsson, 2002; Pakucs, 2003). Multi-domain dialog systems that employ the distributed architecture first select a domain based on a user utterance, and then execute the domain-specific processes of the selected domain (Lin et al., 1999). Therefore, previous research has focused on the correct selection of a single domain based on a user utterance (Çelikyilmaz et al., 2011; Ikeda et al., 2008; Nakano et al., 2011).

However, to our knowledge, no previous research has focused on the selection of one or more domains at the same time for multi-domain dialog systems that provide service to closely related domains. For example, suppose that a multi-domain dialog system provides service to a TV program and video-on-demand (VOD) domains. When a user asks "*Are there any animation programs?*" the system should select both the TV program and the VOD domains. In contrast, when the user says "*Play it.*" in the middle of a dialog for the VOD domain, the system should select the VOD domain based on the dialog history, although the user utterance could be accepted by both the TV program and the VOD domains. However, traditional multi-domain dialog systems have no method of selecting one or more domains at the same time.

In this paper, we proposed a hierarchical domain model (HDM)-based multi-domain selection framework (MDSF). The HDM-based MDSF selects one or more domains at the same time. The HDM-based MDSF consist of two processes: statistically detecting one or more candidate domains based on a user utterance and heuristically determining one or more final domains from among the candidate domains based on the previous domains and the type of the dialog act of the user utterance. The HDM is used in both the candidate domain detection component and the final domain determination component. We developed a multi-domain dialog system using the HDM-based MDSF for TV program, VOD, and TV device domains to validate the HDM-based MDSF.

This paper is organized as follows: Section 2 briefly introduces related work. Section 3 introduces multi-domain dialog systems that employ the MDSF. Section 4 describes the detailed method of the HDM-based MDSF. Section 5 demonstrates the experimental results of the HDM-based MDSF. Finally, we draw conclusions and make suggestions for future work.

2 Related work

Most research on domain selection has focused on selecting a domain correctly. To avoid erroneous domain switching, a two-stage domain selection framework determines whether the previous domain is continued, and then selects another domain only if the previous domain is determined to not be continued (Nakano et al. 2011). To cope with speech recognition errors and grammatically incorrect user utterances, a robust domain selection method integrates topic estimation results and dialog history (Ikeda et al., 2011).

Most research on domain selection has not considered the scenario of encountering a user utterance that can be served by several domains together at the same time. In contrast, we consider multi-domain dialog systems that provide service to one or more domains at the same time. Therefore, we proposed the HDM-based MDSF.

3 Multi-domain dialog systems

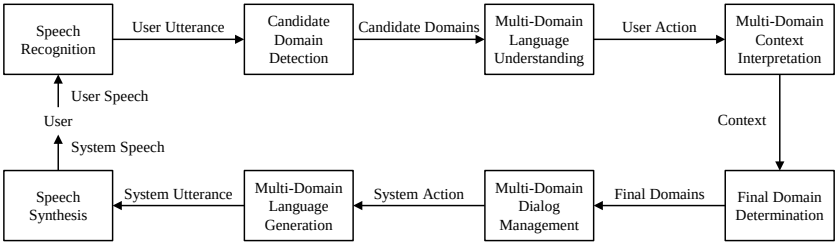

Figure 1 - The architecture of a multi-domain dialog system.

A dialog system is a computer software program that provides natural and effective interaction between humans and machines (McTear, 2002). Users ask dialog systems for services using natural language, and the dialog systems respond using natural language. Some multi-domain dialog systems select one or more domains based on a user utterance and provide service to the selected domains at the same time. The architecture of these multi-domain dialog systems (Figure 1) consists of eight main components:

- **Speech recognition**: the recognition of a user utterance from a user speech.
- **Candidate domain detection**: the detection of one or more candidate domains based on a user utterance.
- **Multi-domain language understanding**: the classification of a dialog act and recognition of a named entity sequence based on a user utterance for the candidate domains.
- **Multi-domain context interpretation**: the determination of either continuing a previous context or setting a new context for the candidate domains.
- **Final domain determination**: the determination of one or more final domains from among the candidate domains.
- **Multi-domain dialog management**: the management of dialog flow by deciding a next system for the final domains.
- **Multi-domain language generation**: the generation of the textual representation of a system action for the final domains.
- **Speech synthesis**: the synthesis of a system speech from the system utterance.

The MDSF consists of the candidate domain detection and final domain determination components. If the MDSF misunderstands the domains of a user utterance, the multi-domain dialog system would perform unexpected behaviors. Therefore, the MDSF should correctly select one or more domains and enable the multi-domain dialog system to provide service to one or more domains at the same time.

Turn	Speaker	Utterance	Domain
1	User	Play “The Closer.”	TV program
	System	Do you mean a TV program or a VOD?	TV program and VOD
2	User	TV program.	TV program
	System	The TV program has been started.	TV program

Table 1 - The dialog in a single-domain scenario in a multi-domain dialog system.

For example, in the dialog in a single-domain scenario in a multi-domain dialog system (Table 1), a user tells the system "*Play* '*The Closer*'." in the first turn. The system understands the domains of the user utterance as being either TV program or VOD, and then asks the user to select the desired domain. This is because playing both TV program and VOD at the same time is impossible to the system.

Turn	Speaker	Utterance	Domain
1	User	Are there any animation programs?	TV program and VOD
	System	This is the list of the related TV programs: (…). This is the list of the related VODs: "Ice Age", (…).	TV program and VOD
2	User	Who starred in "Ice Age"?	VOD
	System	No such TV program is available. Denis Leary, (…) starred in the VOD.	TV program and VOD
3	User	I want to watch it.	VOD
	System	The VOD has been started.	VOD

Table 2 - A dialog in a multi-domain scenario in a multi-domain dialog system.

In contrast, in the dialog in a multi-domain scenario in a multi-domain dialog system (Table 2), a user asks the system "*Are there any animation programs?*" in the first turn. The system understands the domains of the user utterance as being both TV program and VOD and presents the user with the list of related TV programs and VODs. In the second turn, the user asks the system "*Who starred in 'Ice Age'?*" The system understands the domains of the user utterance as being both TV program and VOD. However, the system presents the user with only the stars of the VOD because no such TV program is available in the system. In the third turn, the user says "*I want to watch it.*" The system understands the domains of the user utterance as being either TV program or VOD. However, by considering dialog history, the system regards the domains of the user utterance as being VOD without asking a domain to the user; the system then plays the VOD.

4 Hierarchical domain model-based multi-domain selection framework

4.1 Hierarchical domain model

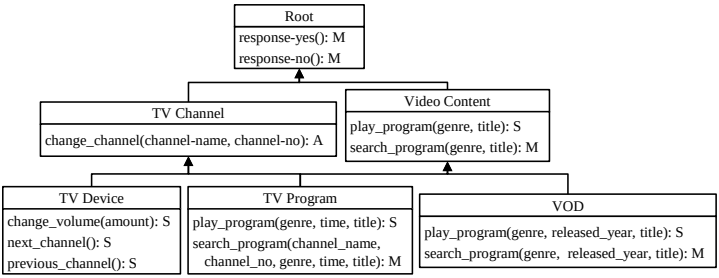

Figure 2 - An example of the hierarchical domain model. TV device, TV program, and VOD are the base domains; root, TV channel, and video content are the expanded domains. (M stands for MULTIPLE; S stands for SINGLE; A stands for ARBITRARY)

We used the HDM in both the candidate domain detection and final domain determination components. The HDM is a formal description of the capabilities of domains and the hierarchical relationships among the domains (Figure 2). The capability of a domain means the dialog acts of the domain, the types of the dialog acts, and the parameters for the dialog acts. The characteristics of the dialog acts of each type are as follows:

- ***MULTIPLE***: the dialog acts can be served by multiple domains at the same time.
- ***SINGLE***: the dialog acts should be served by only one domain.
- ***ARBITRARY***: the dialog acts should be served by only one domain, but the result of the action is equal in all domains.

In the HDM, each domain is either a base domain or a virtual expanded domain. A base domain is the basic unit of functionality designed for the multi-domain dialog system. A virtual expanded domain has multiple child domains, which inherit the definition of the virtual expanded domain. A domain can define a new dialog act or redefine an existing dialog act of the parent domain by adding more parameters to the dialog act. When the domain does not redefine the inherited dialog act of the parent domain, the dialog act does not need to be explicitly described.

4.2 Candidate domain detection

The candidate domain detection component takes a user utterance for its input and detects one or more candidate domains for its output. The candidate domain detection component consists of the in-domain verification components of all the domains; the output of the candidate domain detection component is an integration of the outputs of the in-domain verification components.

4.2.1 Training phase

The basic method for training an in-domain verification component of the candidate domain detection components is to use an in-domain corpus as a positive example and out-domain corpora as negative examples. The in-domain verification component is then trained using a keyword-based approach or a feature-based approach (Chelba et al., 2003; Komatani et al. 2006).

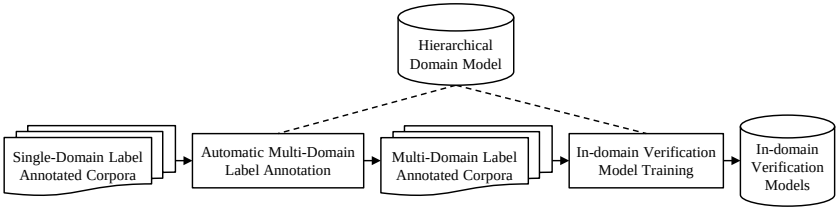

Figure 3 - Candidate domain detection component training.

However, the domain of the corpus to utterances belong cannot be used directly to train the in-domain verification component. This is because some user utterances of the other domains are not negative examples but are positive examples when the domains are closely related to each other. For example, a user utterance "*Are there any animation programs?*" in the TV program corpus is a positive example of both the TV program and VOD domains. Therefore, multi-domain labels on corpora should be automatically annotated before training in-domain verification components (Figure 3).

In the automatic multi-domain label annotation process, multi-domain labels for all user utterances in corpora are automatically annotated using the HDM. More specifically, when several multi-domains can accept a user utterance by considering the dialog act and the named entity sequence of the user utterance, the multi-domain label of the user utterance is annotated as the most general one from among the multi-domains. For example, a user utterance "*Do you have action?*" [*search_program*(*genre*='*action*')] can be accepted by the video content, the TV program, and the VOD domains. The video content domain is the most general domain from among these domains; therefore, the multi-domain label of the user utterance is annotated as video content.

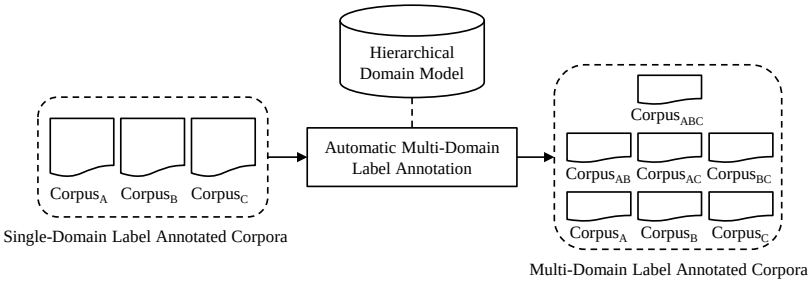

Figure 4 - An example of automatic multi-domain label annotation for domain A, B, and C.

After the automatic multi-domain label annotation, the multi-domain labels of positive examples of a single-domain are the domain or its parent domains; the multi-domain labels of negative examples of the single-domain are remaining domains. For example, when the single domains are A, B, and C, the automatically annotated multi-domain labels are A, B, C, AB, AC, BC, and ABC (Figure 4). The multi-domain labels of the positive examples of single-domain A are A, AB, AC, and ABC; the multi-domain labels of negative examples of single-domain A are remaining domains.

4.2.2 Decoding phase

The candidate domain detection component takes user utterance for its input and detects one or more candidate domains for its output. The candidate domain detection component integrates the outputs of the in-domain verification components of all the domains. The in-domain verification component of each domain verifies whether the user utterance can be accepted by the domain.

4.3 Final domain determination

The final domain determination component takes the candidate domains, a dialog act, a named entity sequence, and a context interpretation result for its input and determines one or more final domains from among the candidate domains for its output. Two cases exist in final domain determination according to the relationship between a set of previous domains and a set of candidate domains.

Case 1: When the previous domain set is a proper subset of the candidate domain set and the multi-domain context interpretation component continues a previous context, the final domain determination component ignores the candidate domains and determines the previous domains as

the final domains. This is because the dialog history implies that the domains do not changed in this case. For example, a user utterance "*Play it.*" can be accepted by both the TV program and VOD domains, but domain switching should not occur for the user utterance in the middle of dialog in the TV program domain.

In addition, to avoid unnecessarily asking a domain, the failed domains are not considered to be previous domains in the next turn. This is because the intended domain of a user utterance within a continued context is only successful domains. For example, suppose a user asks "*Do you have animation programs for adult?*" and no such TV program is available. The system should then inform the user that no such TV program is available and present the user with the list of related VODs. When the user says "*Play the first one.*" in the next turn, the intended domain of the user is VOD not both TV program and VOD.

Case 2: Otherwise, the final domain determination component determines final domains from among the candidate domains based on the type of dialog act described in Section 4.1.

- ***MULTIPLE***: determines all candidate domains as final domains.
- ***SINGLE***: asks the user to select one domain from among the candidate domains.
- ***ARBITRARY***: determines an arbitrary candidate, but priority is given to the previous domain.

5 Experiments

5.1 Candidate domain detection

We used 5-fold cross validation to evaluate the candidate domain detection component of the proposed HDM-based MDSF using the corpora of three base domains, which consist of 2628 user utterances. In the corpora, 52.6% of user utterances belong to only one domain; the others belong to more than one domain. The multi-domain label answers were annotated by hands for evaluation. For the evaluation metrics, we used precision, recall, and F-1 score. We used the Maximum entropy classifier (Ratnaparkhi, 1998) to implement the in-domain verification components of the proposed candidate domain detection component. The baseline is the traditional domain detection component that the domains of the corpora to the user utterances belong are used directly to train the domain detection component.

Component	Precision	Recall	F-1 score
Baseline	97.1%	65.2%	78.0%
Proposed	95.6%	96.2%	95.9%

Table 3 - The result of the candidate domain detection experiments.

The proposed candidate domain detection component had much higher accuracy, recall, and F-1 score, but slightly lower precision than did the baseline component; the recall increased from 65.2% to 96.2%, the precision decreased from 97.1% to 95.6%, and the F-1 score increased from 78.0% to 95.9% (Table 3). The recall of the baseline component was too low because it made numerous false negative errors; i.e. it cannot detect the domains to which a user utterance may refer when the user utterance can be accepted by more than one domain. In contrast, the recall of the proposed candidate domain detection component was high because it made very few false negative errors.

5.2 Multi-domain dialog systems

We used human user experiments to evaluate the multi-domain dialog system that employed the proposed HDM-based MDSF to validate its effectiveness. We exclude the speech recognition component and speech synthesis component in the experiments because these components are independent to the domain. We asked 10 student volunteers to complete 10 dialog tasks involving TV program, VOD, TV device, or combinations of them. For the evaluation metrics, we used successful turn rate (STR), task completion rate (TCR), and average turn length (ATL). STR indicates the average success turn rate of user utterances; TCR indicates the average success rate of the tasks; ATL indicates the average turn length of the dialogs. We excluded the top-most and the bottom-most outliers for each task. The baseline is the traditional multi-domain dialog system that selects only one domain at a time.

System	STR	TCR	ATL
Baseline	55.0%	58.8%	4.7
Proposed	91.1%	95.0%	3.5

Table 4 - The result of the multi-domain dialog system experiments.

The proposed system had higher STR and TCR and lower ATL than did the baseline system; the STR increased from 55.0% to 91.1%, the TCR increased from 58.8% to 95.0%, and the ATL decreased from 4.7 to 3.5 (Table 4). More specifically, the STR, the TCR, and the ATL of each task were improved in all tasks. The STR and the TCR of the proposed system were high because the HDM-based MDSF correctly selects the domains of interest of users. The ATL of the proposed system was low because the HDM-based MDSF enables the proposed system to provide service to one or more domains at the same time.

Conclusion and future work

Is this paper, we proposed the HDM-based MDSF. The experimental results show that the HDM-based MDSF correctly selects one or more domains and enables multi-domain dialog systems to provide more accurate and rapid dialog service than traditional multi-domain dialog systems. To our knowledge, this paper is the first work on the selection of one or more domains in multi-domain dialog systems.

We plan to research multi-domain user simulation. A simulated user experiment is a useful method for evaluating dialog systems with large number of dialogs because a human user experiment is time-consuming and expensive; however, no existing user simulator can simulate users within multi-domain dialog systems that employ the MDSF. Therefore, multi-domain user simulation is an important part of future research on domain selection.

Acknowledgments

This research was supported by the Basic Science Research Program through National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2011-0027953).

This research was supported by the MKE(The Ministry of Knowledge Economy), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2012-H0301-12-3002).

References

Allen, J., Byron, D., Dzikovska, M., Ferguson, G., Galescu, L., and Stent, A. (2000). An architecture for a generic dialogue shell, *Natural Language Engineering*, 6(3): 213-228.

Çelikyilmaz, A., Hakkani-Tür, D. Z., and Tür, G. (2011). Approximate inference for domain detection in spoken language understanding, In *Proceedings of the Interspeech 2011*, pages 713-716, Florence, Italy.

Chelba, C., Mahajan, M., and Acero, A. (2003). Speech utterance classification, In *Proceedings of the ICASSP 2003*, pages 69-72, Hong Kong, China.

Ikeda, S., Komatani, K., Ogata, T., and Okuno, H. G. (2008). Extensibility verification of robust domain selection against out-of-grammar utterances in multi-domain spoken dialogue system, In *Proceedings of the Interspeech 2008*, pages 487-490, Pittsburgh, Pennsylvania, USA.

Komatani, K., Kanda, N., Nakano, M., Nakadai, K., Tsujino, H., Ogata, T., and Okuno, H. G. (2006). Multi-domain spoken dialogue system with extensibility and robustness against speech recognition errors, In *Proceedings of the SIGdial 2006*, pages 9-17, Sydney, Australia.

Larsson, S. and Ericsson, S. (2002). GoDiS – issue-based dialogue management in a multi-domain, multi-language dialogue system, In *Proceedings of the ACL 2002 Demonstration Abstracts*, pages 104-105, Philadelphia, Pennsylvania, USA.

Lee, C., Jung. S., Kim, S., and Lee, G. G. (2009). Example-based dialog modelling for practical multi-domain dialog system, *Speech Communication*, 51(5): 466-484.

Lin, B., Wang, H., and Lee, L. (1999). A distributed architecture for cooperative spoken dialogue agents with coherent dialogue state and history, In *Proceedings of the ASRU 1999*, Keystone, Colorado, USA.

McTear, F. M. (2004). Spoken Dialogue Technology: Towards the Conversational User Interface, Springer.

Nakano, M., Sata, S., Komatani, K., Matsuyama, K., Funakoshi, K., and Okuno, H. G. (2011). A two-stage domain selection framework for extensible multi-domain spoken dialogue systems, In *Proceedings of the SIGdial 2011*, pages 18-29, Portland, Oregon, USA.

Pakucs, B. (2003). Towards dynamic multi-domain dialogue processing, In *Proceedings of the Interspeech 2003*, pages 741-744, Geneva, Switzerland.

Ratnaparkhi, A. (1998). Maximum entropy models for natural language ambiguity resolution, Doctoral Dissertation, University of Pennsylvania, Philadelphia, USA.