

대화 예제와 아젠다를 이용한 음성 인식 오류에 강인한

대화 관리 방법*

이청재, 정상근, 김경덕, 이근배

포항공과대학교 컴퓨터 공학과

{lcj80, hugman, getta, gblee}@postech.ac.kr

Robust Dialog Management with N-best Hypotheses Using Dialog Examples and Agenda

Cheongjae Lee, Sangkeun Jung, Kyungduk Kim, Gary Geunbae Lee
POSTECH Department of COmputer Science and Engineering

요 약

This work presents an agenda-based approach to improve the robustness of the dialog manager by using dialog examples and n-best recognition hypotheses. This approach supports n-best hypotheses in the dialog manager and keeps track of the dialog state using a discourse interpretation algorithm with the agenda graph and focus stack. Given the agenda graph and n-best hypotheses, the system can predict the next system actions to maximize multi-level score functions. To evaluate the proposed method, a spoken dialog system for a building guidance robot was developed. Preliminary evaluation shows this approach would be effective to improve the robustness of example-based dialog modeling.

1. 서 론

음성 대화 시스템(Spoken Dialog System)은 음성 및 자연어를 이용하여 컴퓨터와 의사소통을 하는 차세대 인터페이스로 최근 인간 언어 기술(Human Language Technology)의 발달로 다양한 서비스 분야에서 개발되어지고 있다. 특히 음성 인식(Automatic Speech Recognition) 과 자연어 이해(Natural Language Understanding) 기술 발달로 음성 대화 시스템은 텔레매틱스, 콜센터, 지능 로봇 등 많은 분야에서 이용되고 있으며 21세기의 유비쿼터스 시대를 위한 필수적인 지능형 인터페이스로 각광받고 있다.

그러나 음성 대화 시스템이 실생활에서 사용되기 위해서는 몇 가지 문제점이 존재한다. 대화 관리(Dialog Manager) 관점에서의 가장 심각한 문제 중에 하나는 음성 인식이나 자연어 이해 모듈로부터 발생하는 오류가 대화 관리 모듈로 넘어오는 오류 전파(Error Propagation) 문제이다. 일반적으로 음성 대화 시스템의 대부분의 오류는 음성 인식과 자연어 이해의 오작동으로 인해 발생한다. 이러한 오류는 대화 관리 모듈이 사용자의 의도를 오해하고 왜곡하여 잘못된 시스템 응답을 생성하게 한다. 이러한 문제를 해결하기 위한 근본적인 방법은 음성 인식 기술이나 자연어 이해 기술의 성능의 강

인함을 향상시키는 것이다. 그러나 실제로 완벽한 음성 인식 기술이나 자연어 언어 이해 기술을 개발하는 것은 불가능하며 소음 환경이나 예측하지 못한 사용자 입력이 상존하고 있다. 그러므로 실용적인 음성 대화 시스템 연구 분야에서는 강인한 대화 관리 기술 개발 연구가 활발하게 이루어지고 있다.

대화 관리 관점에서 오류 처리를 위한 방법 중에 하나는 오류 검출(Error Detection) 과 오류 복구 (Error Recovery)를 통해 시스템이 사용자가 오류를 파악하고 수정할 수 있도록 피드백을 주는 것이다 [1,2,3]. 이러한 피드백을 이용하여 사용자는 대화를 통해 잠재된 오류를 수정하여 계속해서 대화를 진행할 수 있다. 대화 관리 관점에서 오류 처리를 위한 다른 방법은 상위 n개의 음성 인식 가설들(n-best hypotheses)를 이용하는 것이다. 실제 부분적으로 관찰 가능한 마르코프 의사결정 과정(Partially Observable Markov Decision Process, POMDP)은 기존의 마르코프 의사결정 과정(Markov Decision Process, MDP) 프레임워크를 확장하여 신뢰 상태(Belief State) 분포 결정을 위해 상위 n개의 음성 인식 가설들을 이용한다. 상위 n개의 음성 인식 결과를 이용하는 것이 강인한 대화 관리 기술 개발에 도움을 준다고 알려져 있다 [4].

본 논문에서는 최근에 제안된 데이터 주도적인 대화 모델링 방법 중의 하나인 예제 기반 대화 모델 (Example-based Dialog Modeling, EBDM)[5]을 기반으로 하여 상위 n개의 음성 인식 가설과 휴리스틱 대화 지식을 이용하여 음성 인식 오류에 강인한 대화 모델 방

* "본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음" (IITA-2008-C1090-0801-0045)

법을 제안 한다. 또한, 휴리스틱 대화 지식을 이용하여 잘못된 대화 진행 오류를 처리할 수 있는 예제 기반 오류 복구 방법도 개발하였다.

본 논문의 구성은 다음과 같다. 2절에서는 기존에 제안된 강인한 대화 관리 기술 방법에 대해서 소개하고, 3절에서는 기존의 대화 예제 기반의 대화 모델 방법을 기술하고 문제점을 고찰한다. 4절에서는 휴리스틱 대화 지식으로 아젠다 기반의 접근법을 제안한다. 5-6절에서는 실제 상위 n개의 음성 인식 가설을 아젠다 지식과 접목하는 방법과 예제 기반 오류 처리 방법에 대해서 제안한다. 7절에서는 실험 방법과 대화 모델 평가 결과를 분석하고 마지막으로 8절에서는 결론 및 향후 연구에 대해서 간단히 기술한다.

2. 선행 연구

최근에 개발되는 많은 음성 대화 시스템들은 사전, 온톨로지 등의 다양한 도메인 지식 정보를 이용한다. 이러한 정보 중에 하나로 도메인 특화된 아젠다(Agenda) 또는 태스크 모델(Task Model)이 사용된다. 이것들은 전체 대화에서 이루고자 하는 복잡한 태스크를 다루기 쉽도록 세부 태스크(subtask)로 분할하여 표현하는 방법이다 [6,7,8]. 이러한 휴리스틱 대화 지식은 대화 모델링, 도메인 지식, 사용자 시뮬레이션, 상태 공간 검색 감소 등의 다양한 목적을 위해 수동으로 제작된다.

예를 들어, CMU(Carnegie Mellon University)에서는 RavenClaw라는 아젠다 기반의 대화 관리 기술을 개발하였다 [7]. 아젠다 기반의 대화 관리 방법은 도메인 특화된 대화 모델을 위해 계층적 태스크 분할을 한 *Dialog Task Specification**과 그것들의 *Expectation Agenda*의 휴리스틱 대화 지식을 이용한다. 각각의 대화 에이전트는 전체 대화 태스크의 특정한 세부 태스크를 다룬다. 최근에는 POMDP 프레임워크에서 나타나는 큰 상태 공간 문제를 해결하기 위해 사용자 목적 트리(user goal tree)와 온톨로지 규칙(ontology rule) 등의 휴리스틱 대화 지식을 이용하는 방법이 개발되었다 [8].

본 논문에서도 아젠다 그래프(Agenda Graph, AG)라는 휴리스틱 대화 지식을 이용한다. 아젠다 그래프는 실제 대화 흐름의 순서를 계층적으로 반영하는 그래프로 기존의 대화 예제 기반 대화 모델 방법에서 나타나는 문제점을 해결하는데 이용된다.

3. 대화 예제 기반 대화 모델링

우리는 과거에 도메인 무관하게 적용할 수 있는 대화 모델링 방법으로 대화 예제 기반 대화 모델링 기술을 제안했다 [5]. 이 방법론은 의미 기반의 대화 예제 데이터베이스(Dialog Example Database)를 이용하여 현재 대화 상태와 가장 가까운 대화 예제를 선택하여 해당하는

적절한 시스템 응답을 생성하는 것이다. 대화 예제 데이터베이스는 의미 분석 및 행위 분석이 된 사람-사람 대화 말뭉치로부터 대화 상태 변수를 이용하여 대화 예제를 인덱싱하여 생성한다. EBDM 프레임워크에서는 적절한 다음 시스템 행위를 결정하기 위해 3가지의 단계가 필요하다.

- 쿼리 생성: 대화 관리자는 자연어 이해 결과와 담화 기록 정보 등을 이용하여 대화 예제 검색을 위한 SQL 조건식을 생성한다.
- 예제 검색: 대화 관리자는 SQL문을 이용하여 현재 대화 상태와 의미적으로 가장 유사한 대화를 선택한다. 검색된 대화 예제가 존재하지 않은 경우에는 특정한 대화 상태 변수를 완화하여 재검색한다.
- 예제 선택: 대화 관리자는 발화 유사도(utterance similarity)를 최대화시키는 대화 예제를 선택한다. 발화 유사도는 발화 패턴과 담화 기록 유사도를 이용하여 계산할 수 있다.

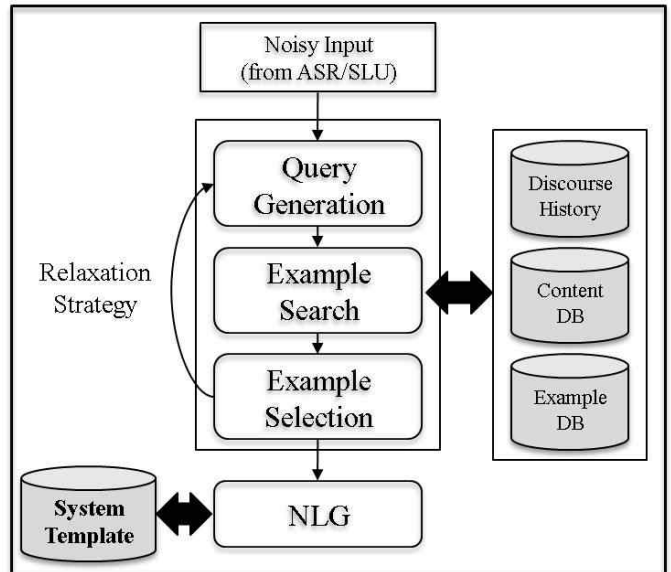


그림 181 예제 기반 대화 모델 전략

<그림 1>는 EBDM 프레임워크의 전반적인 대화 관리 전략을 나타낸다. EBDM 프레임워크는 다양한 도메인에서 음성 대화 시스템을 쉽고 빠르게 개발하기 위한 방법이다 [9]. 그러나 목적 지향 대화를 위해서 두 가지 문제점을 선결해야한다: (1) 태스크 완료를 위한 대화 상태 추적을 해야하며, (2) 음성 인식 오류를 처리하기 위해 n-best 음성 인식 가설을 이용해야한다. 따라서, 우리는 아젠다 그래프라는 휴리스틱 대화 지식을 이용하여 위의 문제들을 해결하였다.

4. 아젠다 그래프

본 논문에서 아젠다 그래프 G 는 도메인 특화된 태스크를 완수하기 위한 세부 대화 흐름을 내포하고 있다. 아젠다 그래프는 루트 노드에서 단말 노드에 가는 가능한

* Dialog Task Specification은 대화 에이전트를 노드로 하는 트리 구조를 가진다.

경로들을 가지고 있으며 이 경로들은 최종 목적을 달성하기 위한 세부 태스크의 순서로 볼 수 있다. 따라서, G 는 특정한 태스크를 완료하기 위한 중간 단계에 해당하는 세부 태스크 노드(N)와 그 노드를 연결하는 유향간선(E)로 이루어진다. 세부 태스크 노드는 3가지 정보를 포함한다: (1) 세부 태스크가 진행되기 위한 선행조건, (2) 노드의 이름과 구분자, (3) 다음 턴에 실행될 수 있는 노드들에 대한 링크. 모든 간선에 대해서 대화 흐름에 대한 선행 지식에 기반하여 천이 확률(transition probability)을 정의했다. 일반적으로 대화 시스템에서 사용자는 목적 달성을 위해 직접적이고 일관된 대화를 진행한다고 가정하면 천이 확률값은 개발자가 기존의 사람-컴퓨터 대화의 실증 분석을 통해 할당할 수 있다.

음성 대화 시스템에서는 성공적인 목적 달성을 위해 사용자와 시스템은 현재까지 진행된 태스크와 다음으로 진행할 태스크에 대해서 알고 있어야 한다. EBDM 프레임워크에서는 아젠다 그래프가 이러한 역할을 할 수 있도록 한다. <그림 2>는 실제 포프(PHOPE)라는 건물 안내를 위한 지능 로봇에서 사용된 아젠다 그래프를 나타낸다. 아젠다 그래프는 대화의 흐름을 직관적이고 유동적으로 표현하기 위해 계층적 트리 구조보다는 방향성 비순환 그래프(Directed Acyclic Graph, DAG)로 표현된다.

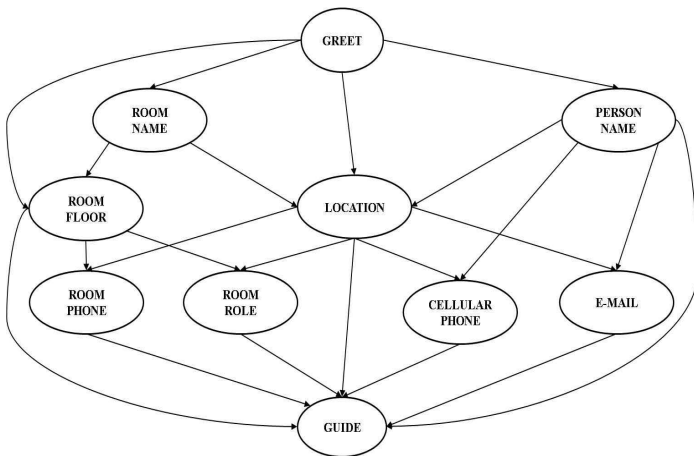


그림 182 건물 안내를 위한 아젠다 그래프

4.1. 대화 예제 매핑

아젠다 그래프 G 의 각 노드 v 는 세부 태스크 진행과 연관된 대화 예제를 포함해야 한다. 그러므로 대화 예제 데이터베이스의 대화 예제를 각 노드에 매핑하는 과정이 필요하다. 대화 예제 매핑을 위해 너비 우선 탐색(Breadth-First Search, BFS) 방식으로 아젠다 그래프를 탐색하여 각 노드의 선행 조건과 각 대화 예제의 선행 조건을 비교하여 참인 경우에는 노드에 매핑이 되도록 한다. 매핑된 대화 예제는 해당 노드에서 나타날 수 있는 발화 패턴 및 대화 상태를 나타낸다.

4.2. 담화 해석

우리는 Lochbaum의 담화 분석 알고리즘[10]을 이용하여 저장된 포커스 스택(Focus Stack)에 존재하는 최근 세부 태스크와 현재 발화가 어떠한 관련이 있는지를 알기 위해서 담화 분석 알고리즘을 구현했다. 본 논문에서의 포커스 스택은 현재까지 선택된 노드들을 저장한 것으로 담화 구조를 내포하고 있다. 포커스 스택은 매 사용자 발화마다 업데이트되며 최상위의 노드가 바로 직전에 선택된 노드로 포커스 노드(focus node)라고 한다. 이전 발화와 현재 발화의 담화 속에서의 연관성을 분석하기 위해 5가지의 담화 상태(Discourse State)를 정의했다.

- **NEW_TASK**: 새로운 아젠다를 완수하기 위해 새로운 대화를 시작함. (루트 노드의 자식 노드)
- **NEW_SUB_TASK**: 현재 아젠다에 포함되는 새로운 세부 태스크를 실행함. (형제 노드)
- **NEXT_TASK**: 다음에 진행할 수 있는 세부 태스크를 실행함. (자식 노드)
- **CURRENT_TASK**: 현재 수행된 세부 태스크를 다시 한 번 더 실행함. (현재 노드)
- **PARENT_TASK**: 이전 세부 태스크에 대한 대화 정보를 수정함. (부모 노드)

괄호 안의 노드의 위치는 포커스 스택의 최상위 노드인 포커스 노드에 대한 상대적 위치를 나타낸다. 만약 **NEXT_TASK**가 선택되어지면, 선택된 노드를 포커스 스택에 삽입을 하고 대화를 계속해서 진행을 하게 된다. **NEXT_TASK**는 대화 진행이 일관된 사용자 발화를 의미한다. 즉, 예측되지 않은 포커스 전환이 일어나지 않고 현재 사용자 발화가 이전 시스템 발화에 강한 상관관계를 가질 때 나타나게 된다. 다른 4가지 경우에도 다양한 담화 상태를 포함할 수 있다. 예를 들어, **NEW_TASK**의 경우에는 루트 노드의 자식 노드가 선택되어지는 경우로 새로운 대화가 시작된다는 것을 의미한다. 그러므로 기존의 포커스 스택을 비우고 선택된 노드를 최상위 노드로 삽입한다. **CURRENT_TASK**의 경우는 같은 세부 태스크를 반복 수행하는 것으로 잘못된 대화 정보를 수정하거나 다른 정보를 검색하는 것을 의미한다.

위의 5가지의 담화 상태에 포함되지 않은 경우에는 담화 해석 알고리즘에 의해 현재 사용자 발화는 거부된다. 현재 사용자 발화가 음성 인식이나 자연어 이해의 오류로 인해 노이즈가 발생했다고 가정하기 때문이다. 이러한 경우에는 오류 복원을 통해 다음에 사용자가 할 수 있는 세부 태스크나 사용자 발화 템플릿을 제공하여 사용자가 쉽게 대화를 진행할 수 있도록 한다.

<그림 3>는 담화 해석 알고리즘을 위한 의사 코드(pseudo code)를 나타낸다. 여기서 S, H, G는 각각 포커스 스택, 음성 인식 가설, 아젠다 그래프를 나타낸다. 담화 해석 알고리즘을 통해 가능한 다음 노드들 중에서 탐욕 알고리즘(greedy selection)을 이용해 최적의 노드를 선택하게 된다.

5. 노드 선택과 예제 선택

| | |
|---|---|
| <pre> INTERPRET(<S,H>,G) ≡ C ← GENERATE(<S,H>,G) if C = 1 then return discourse state in C else c* ← SELECT(<S,H>,C) return selected discourse state in c* </pre> | <pre> GENERATE(<S,H>,G) ≡ return a union set of : i) NEW_TASK(<S,H>,G) ii) NEW_SUB_TASK(<S,H>,G) iii) NEXT_TASK(<S,H>,G) iv) CURRENT_TASK(<S,H>,G) v) PARENT_TASK(<S,H>,G) </pre> |
| <pre> NEXT_TASK(<S,H>,G) ≡ C ← θ E ← retrieved examples of H foreach c ∈ E foreach c_e ∈ {children of top(S)}(G) if e is an example of c_e then C ← C ∪ {c_e} return C </pre> | <pre> SELECT(<S,H>,C) ≡ c* = argmax_c ωS_H(H) + (1-ω) S_D(c ∈ C S) return c* </pre> |

그림 183 담화 해석 알고리즘 의사 코드

음성 인식기에서 제공되는 n-best 음성 인식 가설들 중에서 아젠다 그래프를 이용하여 최적의 음성 인식 가설을 선택하기 위해서 다양한 점수 함수를 정의하였다. 현재 대화 관리자는 루트 노드부터의 천이보다는 바로 직전의 노드부터의 순간 천이에 의존하는 탐욕 알고리즘을 이용하였다.

우선, 각각의 음성 인식 가설(h)을 검색하여 담화 해석 알고리즘을 이용해 선택 가능한 후보 노드(c)를 생성한다. 주어진 후보 노드들 중에서 노드 점수를 계산하여 가장 높은 점수의 노드를 선택한다. 그 다음에 선택된 노드에 존재하는 대화 예제들 중에서 예제 점수를 계산하여 현재 사용자 발화와 가장 유사한 예제를 선택하게 된다.

최적의 노드 선택을 위한 노드 점수는 음성 인식 가설의 신뢰 점수와 노드 천이 확률값의 선형보간법을 이용한다. 이 때, $NEXT_TASK$ 의 경우에는 노드 천이 확률값으로 아젠다 그래프의 천이 확률값을 이용할 수 있으나 다른 담화 상태 경우에는 천이 확률값이 존재하지 않는다. 그러므로 현재 노드에서 가능한 천이 확률값 중에서 가장 작은 값을 선택하고 후보 노드와의 거리를 계산하여 거리만큼 패널티를 주는 방법을 이용한다. 이것은 목적 지향 대화에서 일반적인 사용자는 일관된 대화를 진행하여 다른 담화 상태로 넘어갈 확률은 $NEXT_TASK$ 보다 낮을 것이라고 가정한다.

최적의 노드가 선택된 후에는 예제 점수를 이용하여 선택된 노드에 매핑된 예제들 중에서 최적의 대화 예제를 선택한다. 예제 선택은 대화 예제 검색 시에 사용된 쿼리 조건들이 얼마나 맞느냐를 나타내는 검색 점수와 기존의 EBDM 프레임워크에서 사용된 발화 유사도 점수를 이용한다.

6. 오류 복구 전략

음성 대화 시스템에서 오류는 항상 존재할 수 있으며 검출된 오류를 복구하기 위해서 오류 복구 전략(error recovery strategy)을 이용한다. 오류 복구 전략은 대화 시스템이 사용자에게 적절한 피드백을 제공하고 사용자가 잘못된 대화 정보를 수정하는 확인(confirmation) 대화 혹은 재발화(Rephrasing) 대화를 진행하는 것이다. EBDM 프레임워크에서는 예제 기반의 오류 복구 전략을

이용한다 [3]. 예제 기반의 오류 복구 전략에서 오류 검출은 현재 사용자 발화와 유사한 대화 예제가 존재하지 않는 경우를 오류로 가정한다. 이것은 음성 인식 오류나 자연어 이해 오류로 인해 검색 조건에 오류가 포함될 수 있기 때문이다. 이러한 경우에 대화 관리자는 *UtterHelp* 피드백을 제공한다. *UtterHelp*는 이전 담화 기록을 이용해 현재 나올 수 있는 대화 예제를 검색하여 현재 발화와 가장 유사한 발화 템플릿을 제공하는 것이다. 그러므로 사용자는 제공된 발화 템플릿을 이용하여 재발화를 할 수 있다. 제공된 발화 템플릿은 음성 인식이나 자연어 이해, 대화 관리 모듈에 훈련되어져 있어서 대화 처리 성능을 향상시킬 수 있다.

본 연구에서는 기존의 예제 기반의 오류 복구 전략에 *AgendaHelp*를 추가한다. *AgendaHelp*는 아젠다 그래프를 이용하여 담화 해석 알고리즘이 실패한 경우, 현재 포커스 노드의 $NEXT_TASK$ 에 해당하는 자식 노드의 노드명을 이용하여 사용자가 다음에 진행할 수 있는 세부 태스크 명을 제공하는 것이다. 즉, 사용자가 다음에 어떤 태스크를 해야할 지 모르는 경우에 대화 시스템이 다음으로 가능한 세부 태스크에 대한 정보를 제공하여 사용자가 쉽게 사용할 수 있도록 하는 것이다. 예를 들어, “SYSTEM: 다음으로 <위치 검색>, <전화번호 검색> 등을 할 수 있습니다.”와 같은 시스템 발화를 제공할 수 있다.

7. 실험 및 결과

본 논문에 소개된 방법에 대한 실험을 위해 PHOPE를 위한 음성 대화 시스템을 개발하였다. PHOPE는 건물 안내를 위한 지능 로봇으로 건물 내에 존재하는 방들의 정보와 구성원의 정보를 제공하고 사용자가 요청 시에는 직접 목적지로 안내를 할 수 있다. 대화 말뭉치 수집을 위해 10명의 학부생들이 주어진 시나리오를 바탕으로 약 120개의 사람-사람 대화를 수집하였다 <표 X>. 그리고 건물 안내를 위한 아젠다 그래프를 작성하여 EBDM 프레임워크와 통합하였다.

표 1 대화 말뭉치 정보

| 도메인 | 건물 안내 도메인 |
|---------------|-----------|
| 대화 수 | 120 |
| 사용자 발화 수 | 512 |
| 대화 당 사용자 발화 수 | 4.26 |
| 발화 당 이형태소 수 | 6.65 |

대화 시스템 평가 방법은 사용자 평가와 시뮬레이션 평가가 있다. 실제 사용자 평가는 대화 시스템을 실제 환경에서 평가할 수 있는 장점이 있지만 객관성이 떨어지고 비용이 많이 드는 단점이 있다. 그러므로 우리는 사용자 시뮬레이터(user simulator)를 개발하였다 [11]. 사용자 시뮬레이터는 현재 대화 상태에서 적절한 사용자의 의도를 추측하고 의도에 맞게 사용자 발화를 생성하게 된다. 생성된 발화는 음성 인식 노이즈 채널(ASR noisy channel)을 통해 음성 인식 오류를 포함한 발화로 변환

된다. 시뮬레이션 실험을 통해 음성 인식 성능에 따른 대화 모델 성능을 측정하였다. 시뮬레이션 평가는 자동으로 생성된 1000대화를 이용하여 평가를 하였다. <표 X>는 시뮬레이션 결과를 나타낸다. <표 X>에서 음성 인식 결과는 상위 1개(1-BEST)와 상위 10개(10-BEST)를 이용하였다. 그리고 대화 모델에서 아젠다 그래프를 사용한 경우(+AG)와 그렇지 않은 경우(-AG), 오류 복구 전략을 사용한 경우(+ER)와 그렇지 않은 경우(-ER)에 대해서 각각 실험을 하였다. 일반적으로 대화 모델 평가는 성능 측면에서 실제 대화 성공률(Task Completion Rate, TCR)과 비용 측면에서 대화당 평균 사용자 발화 수를 측정하게 된다. 대화 성공률이 높을수록 사용자가 현재 대화 시스템을 이용하여 주어진 목적을 달성할 확률이 높다는 것을 의미한다. 또한, 대화당 평균 사용자 발화 수가 낮을수록 빠른 시간에 대화를 마칠 수 있다는 것을 의미한다.

표 2 대화 시뮬레이션 결과

| 단어 인식 오류율 (%) | 시스템 | 대화당 사용자 발화 수 | 대화 성공률 (%) |
|---------------|--------------------|--------------|--------------|
| 10 | 1-BEST (-AG, -ER) | 5.85 | 81.01 |
| | 1-BEST (-AG, +ER) | 5.90 | 83.50 |
| | 10-BEST (-AG, -ER) | 6.43 | 92.66 |
| | 10-BEST (-AG, +ER) | 6.17 | 93.87 |
| | 1-BEST (+AG, -ER) | 5.85 | 67.94 |
| | 1-BEST (+AG, +ER) | 5.90 | 70.75 |
| 20 | 10-BEST (+AG, -ER) | 6.55 | 90.85 |
| | 10-BEST (+AG, +ER) | 6.39 | 91.75 |

<표 2>를 보면, 실제 음성 인식 결과를 상위 10개를 이용하고 아젠다 그래프와 오류 복구 전략을 동시에 사용한 경우에 대화 성공률이 93.87%(WE=10%), 91.75%(WER=20%)로 가장 높았다. 이 결과로부터 1-best 음성 인식 결과를 사용하는 것보다 n-best 음성 인식 결과를 사용하는 것이 효과적인 것을 알 수 있다. 또한, 아젠다 그래프와 오류 복구 전략 등을 이용하여 기존의 EBDM 방법을 음성 인식 오류에 강인하게 만들 수 있음을 알 수 있다.

8. 토의 및 향후 연구

본 논문에서는 기존의 아젠다 기반의 대화 모델 방식을 EBDM 방법에 접목하여 음성 인식 오류에 강인한 대화 모델을 구현하는 방법을 제안했다. 아젠다 그래프는 수동으로 구축하는 비용이 들 수 있지만 대화 시스템에서 대화 상태를 추적하거나 담화 해석을 하는데 도움을 줄 수 있다. 시뮬레이션 실험을 통해 본 연구에서 제안한 방법이 대화 성공률을 향상을 얻었다.

향후 연구 계획으로는 제안된 방법에 사용된 아젠다 그래프를 제작하는 비용을 줄이기 위해서 사용자 발화 클러스터링 기법을 통한 반자동 아젠다 그래프 생성 방법을 고안 중이다.

참고 문헌

- [1] McTear, M., O'Neil, I., Hanna, P., and Liu, X. Handling errors and determining confirmation strategies-An object-based approach. *Speech Communication*, 45(3):249-269. 2005.
- [2] Torres, F., Hurtado, L.F., Garcia, F., Sanchis, E., and Segarra, E. Error Handling in a Stochastic Dialog System through Confidence Measure. *Speech Communication*, 45(3):211-229. 2005.
- [3] Lee, C., Jung, S., and Lee, G.G. Example-based Error Recovery Strategy For Spoken Dialog System. *Proceedings of the IEEE Automatic Speech Recognition and Understanding Workshop*, 538-543. 2007.
- [4] Williams, J.D. and Young, S. Partially Observable Markov Decision Processes for Spoken Dialog Systems. *Computer Speech Language*, 21(2):393-422. 2007.
- [5] Lee, C., Jung, S., Eun, J., Jeong, M., and Lee, G.G. A Situation-based Dialogue Management using Dialogue Examples. *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*, 69-72. 2006.
- [6] Rich, C. and Sidner, C.. Collagen: A Collaboration Agent for Software Interface Agents. *Journal of User Modeling and User-Adapted Interaction*, 8(3):315-350. 1998.
- [7] Bohus, B. and Rudnicky A. RavenClaw: Dialog Management Using Hierarchical Task Decomposition and an Expectation Agenda. *Proceedings of the European Conference on Speech, Communication and*

Technology, 597–600. 2003.

[8] Young, S., Schatzmann, J., Weilhammer, K., and Ye, H..

The Hidden Information State Approach to Dialog Management. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, 149–152. 2007.

[9] Lee, C., Jung, S., Jeong, M., and Lee, G.G. *Chat and*

Goal-oriented Dialog Together: A Unified Example-based Architecture for Multi-domain Dialog Management. Proceedings of the IEEE Spoken Language Technology Workshop, 194-197. 2006.

[10] Lochbaum, K.E. 1998. *A Collaborative Planning Model of Intentional Structure. Computational Linguistics*, 24(4):525–572.

[11] Jung, S., Lee, C., Kim, K., Lee, G.G. *An integrated*

dialog simulation technique for evaluating spoken dialog systems. Proceedings of the COLING 2008 workshop on Speech Processing for Safety Critical Translation and Pervasive Applications. 2008.