

대화형 일정관리 에이전트의 기능 향상을 위한 스크립트 설계 방안

임수정* 최봉환* 임성수*, 조성배*

*연세대학교 컴퓨터과학과

e-mail:{soojung, bitbyte, lss}@sclab.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Script Design Method for Functional Improvement in Conversational Schedule Management Agent

Soojung Lim*, BongWhan Choe*, Sung-Soo Lim*, Sung-Bae Cho*

*Dept of Computer Science, Yonsei University

요 약

최근 개인의 업무 능률 향상을 위해 일정 관리를 대행해주는 지능형 에이전트에 대한 연구가 진행 중이다. 사용자는 쉽고 친숙한 인터페이스 환경을 제공하며 유연하고 풍부한 의사전달 과정인 대화를 보다 선호하게 되는데, 본 논문에서는 CAML을 사용하여 효율적인 대화 스크립트 설계 방법을 제안한다. 스크립트 설계 시, 예외처리에 시간이 많이 소요되는 등 기존 연구에서의 문제점을 해결하기 위해 스크립트는 도메인 분석, 대화흐름 설계, 대화로그 수집 및 대화 스크립트 생성, 외부 함수 정의, 실제 일정관리 에이전트에서의 적용의 단계를 거쳐 생성되며, 생성된 스크립트는 실제 대화 에이전트를 통해 일정관리의 기능을 수행한다. 10명의 사용자를 대상으로 비교평가를 통해 제안하는 스크립트 설계방법이 우수함을 보였고, 사람 간 대화와의 유사성 역시 높음을 보였다.

1. 서론

일정관리는 업무 능률을 향상시킬 수 있도록 회의, 약속 등에 대한 일정을 작성, 변경, 조정하는 것으로 체계적이고 효율적인 관리를 위해서는 많은 시간과 노력이 필요하다[1]. 하지만 기존 연구들의 대부분은 사용자가 마우스 클릭이나 키보드 타이핑을 통해 자신의 일정을 시스템에 단 방향으로 입력해야 하고, 시스템의 구조가 복잡할 경우 시스템 사용을 위한 지식이 필요하여 일정관리 자체를 위한 시간이 많이 소요되기도 한다. 사용자는 쉽고 친숙한 인터페이스 환경과, 시스템과 상호작용 가능한 양방향의 서비스를 선호하게 되는데[2], 그 중 대화는 보다 유연하고 풍부한 의사 전달을 하는데 유용하다. 자연어 대화에 기반을 둔 지능형 에이전트는 사용자의 질의에 대한 서비스를 제공하기 위해 이에 필요한 행동을 일련의 규칙에 따라 스크립트에 기술한다. 이때 인간의 대화가 가지는 단점(주제 및 내용이 복잡(Cycle, Subdialogue, branch 등)함, 너무 많은 정보 및 쓸모없는 데이터를 포함[3])을 해결하기 위해, 대화 스크립트는 이해하기 쉬워야 하고, 최대한 발생 가능한 다양한 주제를 다뤄야 하며, 토픽 전환이 자유로워야 한다는 조건을 갖는다. 실 예로 D. Chiu는 대화 에이전트를 위한 좋은 스크립트의 조건으로 이해하기 쉽고 디버그 및 유지가 쉬우며, 재사용성이 높아야 한다고 강조하고 있다[4].

전통적인 대화형 에이전트는 스크립트 내 패턴매칭 기술을 통해 사용자의 질의에 대응하는 답변을 제공한다. 또

이를 바탕으로 보다 복잡한 분석과 의도처리를 통해 다양한 어플리케이션에 대화형 인터페이스를 적용하기도 한다. 따라서 본 논문에서는 대화형 일정관리 에이전트의 기능 향상을 위한 CAML 기반의 스크립트 설계 방안을 제안한다. CAML은 XML을 기반으로 하여 이해가 쉬우며, 대화 에이전트 설계자가 도메인에 필요한 템플릿 변수를 직접 명시 할 수 있어 재사용성이 높고, 도메인에 필요한 기능을 수행할 수 있도록 사용자 정의 함수를 동작시킬 수 있어 디버그 및 유지가 쉽다[5]. 스크립트의 설계는 크게 다섯 가지 단계로 도메인 분석, 대화흐름 설계, 대화로그 수집 및 대화 스크립트 생성, 외부 함수 정의, 실제 일정관리 에이전트에서의 적용으로 나눌 수 있다. 생성된 스크립트는 대화형 일정관리 에이전트를 통해 일정 추가, 변경, 삭제, 검색 등의 기능을 수행하게 하며, 위의 설계 방법을 거치지 않고 생성된 스크립트와의 설계 시간, 만족도 및 사람 간 대화와의 유사성 비교를 통해 제안하는 방법의 유용성을 검증하였다.

2. 관련 연구

2.1 지능형 대화 에이전트

지능형 에이전트 분야는 연구 및 개발 영역이 빠르게 확산되고 있으며, 보다 효율적이고 사용자 중심적인 인터페이스 설계에 초점을 맞추고 있다. 지능형 에이전트의 효과적인 상호작용 수단으로 자연어 대화를 통해 사용자와의 양방향 커뮤니케이션을 가능하게 하는 대화 에이전트

의 연구가 최근 지속적으로 진행되고 있다[6]. 대화 에이전트는 사용자가 원하는 정보를 단순한 키워드나 메뉴 등의 방식으로 얻는 것이 아니라, 자연어로 된 문장을 입력하게 함으로써, 사용자에게 친숙한 인터페이스 환경을 제공한다. 자연어를 통한 대화는 유연성, 명료성, 표현력 면에서 뛰어난 장점을 가지고 있기 때문에 인간과 컴퓨터간의 상호작용 수단으로 적합하며[7], 인간과 컴퓨터간의 상호작용을 통해 보다 자연스러우며, 복잡한 정보를 효율적으로 처리할 수 있게 해준다. 따라서 대화를 통해서 사용자와의 상호작용을 가능하게 하는 대화 시스템을 개발하는 것은 바람직하며, 이는 복잡한 시스템에서 더욱 효과적인 사용자 인터페이스가 된다[8].

2.2 대화 에이전트를 위한 스크립트 생성

자연어 대화에 기반을 두어 사용자와 상호작용을 하는 지능형 에이전트는 질의에 대한 서비스를 제공하기 위해 이에 필요한 행동을 일련의 규칙에 따라 스크립트에 기술한다. D. Chiu는 대화기반의 인터넷 검색 에이전트를 위한 스크립트 설계 시, 기본 언어 구축 및 제어 흐름 정의, 정보추출, 대화 형태 정의, 예외 처리의 과정을 거쳐 스크립트를 생성 하였다[9]. 또 N. Belkin 등은 정보검색 시스템을 위한 스크립트 설계 시, 정보 처리를 위한 대화형태 정의, 대화 원형(Prototype)을 위한 기본 스크립트 설계, 정보검색 시 필요한 사건 정의, 실제 검색에 적용의 단계를 거쳤다[10]. 하지만, 질의응답을 어휘 수준에서 처리하여 미리 설계된 패턴에 정확히 일치하는 경우에만 답변을 제공하기 때문에 대화의 유형이 단순하고, 얼마나 많은 양의 스크립트를 보유하느냐로 그 성능이 결정되기 때문에 스크립트 구축에 상당한 시간을 필요로 하며, 예외 처리에도 시간이 많이 소요된다는 단점이 있다.

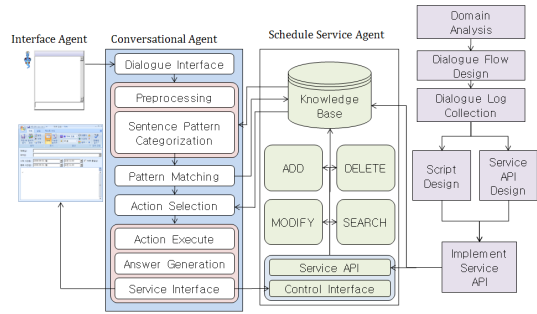
본 논문에서는 도메인 분석, 대화흐름 설계, 대화로그 수집 및 대화 스크립트 생성, 외부 함수 정의, 실제 일정관리 에이전트에서의 적용의 다섯 단계를 통해, 보다 체계적이고 효율적으로 스크립트를 생성하는 방법을 제안한다.

3. 대화형 일정관리 에이전트와 스크립트 설계

스크립트 설계 프로세스를 포함한 대화형 일정관리 에이전트의 전체 시스템 구조는 그림 1과 같다. 먼저 사용자 질의가 입력되면, 형태소와 패턴을 분석하여 사전에 정의된 스크립트 데이터베이스로부터 해당 질의와 그에 따른 답변을 매칭하게 된다. 이때 스크립트는 제안하는 다섯 가지 설계 단계를 거쳐 생성이 되며, 스크립트 내에서 함수 호출(Function call)을 통해 일정관리 모듈에서 적합한 서비스를 수행하게 된다. 본 논문에서는 대화형 일정관리 에이전트에 적합한 스크립트를 설계하는 것에 초점을 맞추고, 마스터 플랜(Master plan)에 따라 하향 설계(Top-down)방식으로 스크립트를 설계하는 방법을 제안한다.

3.1 도메인 분석

도메인 분석은 스크립트 설계의 첫 단계로 대화 에이



(그림 1) 에이전트의 구조 및 스크립트 설계 프로세스

전트를 일정관리 시스템에 적용하기 위한 일정관리 도메인의 분석 과정을 말한다. 도메인 분석은 주어진 환경을 정의하고 보다 다양한 서비스를 제공하기 위해 선행되어야 한다. 일정관리의 경우, 크게 일정 추가, 변경, 삭제, 검색의 4가지로 구성되며 각각의 도메인은 반드시 포함되어야 하는 요소 및 포함 가능한 요소로 정의된다. 표 1은 일정관리 도메인에 대한 정의를 나타낸다.

<표 1> 일정 관리 도메인 및 포함 요소

도메인	요소
추가	장소명 유형 이름 참여자 목록 일정 시작날짜 일정 끝날짜 일정 시작시간 일정 끝시간
변경	일정 ID 장소명 참여자 목록 유형 이름 시작일 끝일 시작 시각 끝 시각
삭제	일정ID 목록
검색	장소명 유형 이름 참여자 목록 날짜 범위 시작 날짜범위 끝 날짜범위 시작시각 날짜범위 끝시각

3.2 대화흐름 설계

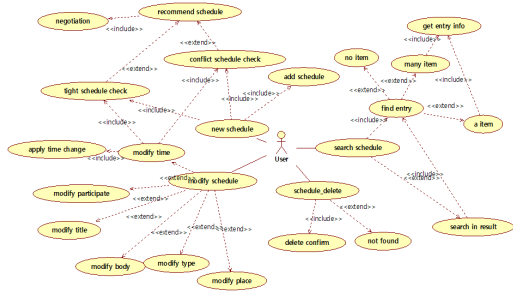
대화 흐름은 스크립트 설계시의 전체적인 마스터 플랜(Master plan)을 나타내는 것으로 전체 일정 관리, 즉 일정 추가, 변경, 삭제, 검색 시 발생 가능한 모든 흐름을 나타낸다. 실 예로 새로운 스케줄이 입력되었을 시, 새로운 스케줄을 입력하는 과정에서 충돌 및 타이트한 스케줄을 체크하고, 이를 해결할 수 있는 적합한 스케줄을 추천받은 후 사용자의 선택을 통해 스케줄을 입력하게 된다. 그림 2는 대화 토픽 간 흐름 Use case를 나타낸다.

3.3 대화로그 수집 및 대화 스크립트 생성

질의 결과는 해당 질의의 답변으로 발생가능한 질의 값에 대해 정의한다. 이는 실제 사람 간의 충분한 대화 로그 수집 후 이상적인 대화의 형태로 정의할 수 있다.

스크립트는 스크립트 언어 및 템플릿 변수를 통해 정의 되는데, 스크립트 언어는 대화의 범주를 나타내거나, 스크립트가 속한 주제나 사용자 입력의 패턴을 명시하고, 답변을 출력하는 항목들을 포함한다. 템플릿 변수는 개체의 범주에 대해 명시를 하거나 해당 변수가 가질 수 있는 값 등을 명시한다[5].

스크립트 언어 내에서 <script>는 대화의 범주를 나타내고, <topic>은 스크립트가 속한 주제를 명시한다.



(그림 2) 대화 흐름 Use case

<necessary_var>는 스크립트 행동 수행에 있어서 필수적으로 필요한 정보 기술을 명시하며, <candidate_var>는 행동 수행에 있어서 있으면 도움이 될 만한 변수에 대해 명시한다. 사용자의 입력은 <query>를 통해, 스크립트의 행동은 <action>을 통해 정의한다. 답변은 [answer_string]의 형태로 <answer>를 통해 출력되며, <function>은 해당 도메인에 관련된 기능을 수행할 수 있도록 한다. 또 템플릿 변수 내에서 <object>는 개체의 범주에 대해 명시하고, <variable>은 추출할 변수에 대해, <value>는 해당 변수가 가질 수 있는 값을, 마지막으로 <default>는 스크립트의 <answer> 태그에서 템플릿 변수 값이 존재하지 않을 때 사용될 값에 대해 명시한다. 그림 3은 생성된 스크립트의 예를 보여준다.

```

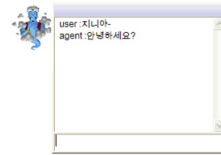
<script>
<topic> 일정입력 </topic>
<topic>
<necessary_var>
<variable> $일정$시간$ </variable>
<variable> $일정$대상$ </variable>
<variable> $일정$날짜$ </variable>
</necessary_var>
<query> $일정$날짜$ $일정$시간$에 $일정$대상$와 약속이 있어. </query>
<action>
<function> schedule_input|$일정$시간|$일정$대상|$일정$날짜$
<return value = true>
<answer> 일정을 등록하였습니다. </answer>
</return>
<return value = conflict>
<function> ques_schedule_input_conflict|$internal$answers$ </function>
<answer> $internal$answers$ </answer>
<change_topic> 일정을 등록 </change_topic>
</return>
</function>
</action>
<topic_question> 일정 추가 하시겠습니까? </topic_question>
</script>
    
```

(그림 3) 생성된 스크립트의 예

3.4 외부 함수 정의

외부 함수는 실제 일정관리 에이전트의 기능 수행을 위해 스크립트 내에서 정의되어야 한다. 외부함수는 스케줄 입력, 검색, 삭제, 검색과 관련한 5개의 스케줄 함수와 데이터 포맷 정렬, 시간 정렬 등의 18개의 라이브러리 함수로 구성되어 있다. 표 2는 스케줄 변경에 대한 외부 함수 정의를 나타내는데, 일정 ID, 장소명 등의 인자 값들과 return 값의 형태에 대해 정의한다. 또한 지니 캐릭터와의 연동을 통해 에이전트가 현재 수행하고 있는 항목들을 애니메이션을 통해 보여주게 되는데, 이는 그래픽 적으로 사용자에게 보다 친숙한 환경을 제공한다. 지니 캐릭터는 Acknowledge, Alert, Announce, Blink 등 74개의 다양한 Action과 SHOWCMD, HIDECMD, MOVECMD 등 8개의

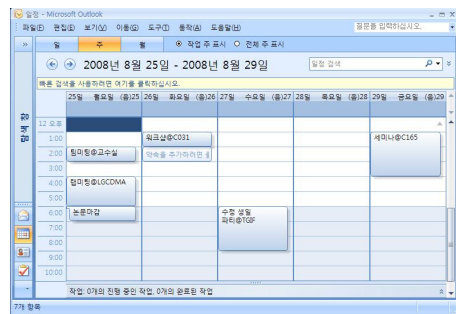
명령 태그로 구성되어 있어, 보다 상황에 적합한 애니메이션을 보여준다. 그림 4는 대화형 일정관리 에이전트의 어플리케이션 및 지니 애니메이션과의 연동 예를 보여준다.



(그림 4) 지니 애니메이션과의 연동 예

3.5 실제 일정관리 에이전트에서의 적용

입력된 사용자 질의는 일정관리 에이전트와 연동되어 해당하는 기능을 수행하게 된다. 그림 4의 어플리케이션을 통해 사용자가 일련의 기능에 대한 질의를 입력할 경우, 이는 대화처리 모듈을 거쳐 MS Outlook과 연동되어 해당 질의에 대한 기능을 수행하게 된다. 그림 5는 실제 대화형 일정관리 에이전트를 통해 스케줄이 등록된 화면을 보여준다.



(그림 5) 실제 스케줄 등록 화면

4. 실험결과

4.1 시나리오 설계

제안하는 방법의 사용성 평가를 위해 표 2와 같은 시나리오를 설계하고, 이에 따라 대화형 일정관리 에이전트를 사용하도록 하였다. 시나리오는 1주일간의 일과를 기준으로, 일정을 추가, 변경, 삭제, 검색하도록 하는 기본 업무와 특정 상황에서 대화형 일정관리 에이전트의 추천을 반영하여 의사결정을 하는 업무로 구성되어 있다.

<표 2> 사용성 평가를 위한 시나리오의 일부

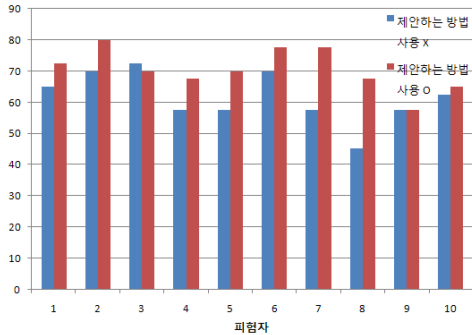
Scenario 1	Conversation example
1. 추가: 내일, 점심식사, 장소, 12시	내일 점심 12시에 생수와 점심을 먹기로 했어. → 네, 풍후 하였습니다.
2. 검색: 27일 일정	27일 일정이 어떻게 되지? → 오후 8시부터 9시까지 수경 생일파티가 IGIE에서 있습니다. 술, 그림 오후 4시에 LG CDMA에서 팀미팅을 하는 걸로 해줘. → 알겠습니다.
3. 추가: 팀미팅, 오후 4시, LG CDMA	28일 목요일에 가족들과 여행을 가기로 했어. → 네, 알겠습니다.

4.2 사용성 평가

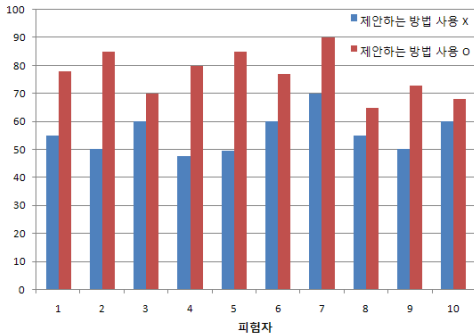
제안하는 방법의 유용성을 보이기 위해 20~29세의 남·여 대학생 10명을 대상으로 위의 설계과정을 따르

지 않은 스크립트와 제안하는 설계과정에 의해 생성된 스크립트에 기반을 둔 대화형 일정관리 에이전트를 사용 후 만족도를 SUS(System Usability Scale)의 항목을 사용하여 평가 하도록 하였다. 또 시스템 사용 후, 각각의 스크립트내의 대화가 얼마나 사람간의 대화와 근접한지를 평가하도록 하였다.

스크립트 설계 시 걸리는 시간은 위의 과정을 따랐을 경우, 예외처리를 위한 설계 시간이 줄어 보다 효율적임을 보였다. 또 그림 6은 두 시스템을 사용 후 느끼는 만족도 여부에 대한 평가로, SUS 10개 항목을 사용하여 평가한 결과를 100점 만점으로 환산하여 보여준다. 제안하는 설계방법을 따른 스크립트를 사용한 대화형 일정관리 에이전트의 만족도는 평균 70.5점으로 그렇지 않은 경우 보다 (평균 61.5) 보다 우수한 것을 확인하였다.



(그림 6) 사용자 만족도 평가 결과



(그림 7) 사람 간 대화와의 유사성

그림 7은 사람과의 대화와 비교했을 때, 제안하는 방법을 통해 설계된 스크립트가 얼마나 사람의 대화와 비슷한지, 얼마나 정확성을 갖는지에 초점을 맞추어 평가한 결과를 보여준다. 평균 유사도는 77.1점으로 그렇지 않은 경우에 비해(평균 55.7)인간의 대화와 보다 유사함을 보였다.

5. 결론 및 향후 계획

본 논문에서는 쉽고 친숙한 인터페이스 환경과, 시스템과 상호작용 가능한 대화형 일정관리 에이전트의 기능 향상을 위한 스크립트 설계 방법을 제안한다. 스크립트의

설계는 다섯 가지 단계인 도메인 분석, 대화흐름 설계, 대화로그 수집 및 대화 스크립트 생성, 외부 함수 정의, 실제 일정관리 에이전트에서의 적용을 거쳐 생성되며, 이는 실제 대화 에이전트에서 자연어 질의와 답변 쌍 매칭을 통해 일정관리 서비스를 제공하게 된다. 실험 결과 위의 설계과정을 따르지 않은 스크립트에 비해 설계시간 및 만족도면에서 우수함을 보였으며, 사람 간 대화와의 유사성 역시 높음을 보였다.

참고문헌

[1] S.-S. Lim and S.-B. Cho, "User adaptive schedule management agent using Bayesian networks," *Proc. of KCC*, vol. 33, no. 1(B), pp. 103-105, 2006.

[2] J.-H. Hong and S.-B. Cho, "A two-stage Bayesian network for effective development of conversational agent," *Fourth International Conference on Intelligent Data Engineering and Automated Learning*, pp. 2-3, 2003.

[3] C.-S. Lee and C.-Y. Pan, "An intelligent fuzzy agent for meeting scheduling decision support system," *Fuzzy Sets and Systems*, vol. 142, no. 3, pp. 467-488, 2004.

[4] S. Bocionek, "Software secretaries : learning and negotiating personal assistants for the daily office work," *In Proc. IEEE Int. Conference on Systems, Man and Cybernetics*, pp. 7-12, 1994.

[5] S.-S. Lim, and S.-B. Cho, "CAML: Conversational agent markup language for application-independent dialogue system," *9th China-India-Japan-Korean Joint Workshop on Neurobiology and Neuroinformatics*, pp. 70-71, 2007.

[6] J. Allen, D. Byron, M. Dzikovska, G.Ferguson, L. Galescu, and A. Stent, "Towards conversational human-computer interaction," *AI Magazine*, vol. 22, no. 4, pp. 27-38, 2001.

[7] K.-M. Kim, J.-H. Hong, S.-B. Cho, "Semi-structure information extraction for the construction of adaptive web knowledge base of conversational agents," *Proc. of Korea Cognitive Science*, pp. 197-201, 2004.

[8] N. Kushmerick, "Gleaning the web," *IEEE Intelligent Systems*, vol. 14, no. 2, pp. 20-22, 1999.

[9] D. K. W. Chiu, "A script language for generating internet-bots," *12th International Conference on Database and Expert Systems Applications (DEXA2001)*, pp. 667-671, 2001.

[10] N. J. Belkin, C. Cool, A. Stein, and U. Thiel, "Cases, script, and information-seeking strategies: On the design of interactive information retrieval systems," *Expert Systems with Applications*, vol. 9, pp. 379-395, 1995.