

근사 패턴매칭을 이용한 대화형 도우미 에이전트의 개발

Development of a Conversational Help Agent Using Approximate Pattern Matching

김 수 영* 조 성 배*
(Soo-Young Kim) (Sung-Bae Cho)

요약 인터넷의 성장에 따라 많은 웹사이트가 생기고, 더 많은 정보가 웹사이트에 등록되었다. 웹사이트에 등록되는 정보가 많을수록, 사용자가 원하는 정보를 얻기가 쉽지 않다. 따라서, 사용자가 원하는 정보를 쉽게 찾을 수 있도록, 웹사이트 내에 전문(full-text) 검색엔진을 도입하기도 한다. 본 논문에서는 사용자가 자연어를 이용한 대화를 통해 웹사이트 내의 정보를 습득할 수 있도록 하는 대화형 도우미 에이전트를 개발한다. 제안한 방법은 전통적인 자연어 처리 기법이 아닌 인공지능의 패턴 매칭에 기반한다. 사용자가 문장을 입력하면, 한글 전처리과정을 통해 사용자의 문장을 분석하고, 이미 작성되어 있는 지식과의 매칭을 통해 사용자에게 알맞은 대답을 제시한다. 지식은 XML 형식으로 저장되며, 사용자가 입력한 문장과 아주 똑같지 않더라도, 어느 정도의 유사도를 가지고 대답을 이끌어낼 수 있다. 실험결과 동일한 의미를 가진 다양한 형태의 문장을 입력했을 경우에도 동일 패턴임을 인식하여, 사용자가 원하는 결과를 낼 수 있었다.

Abstract As Internet grows, many web sites have been built, therefore much information has been registered. Because the web sites have more information, it is more difficult that the user can find the information wanted. Therefore, to get information that user wants easily, the full-text engine may be embedded to the web site. This paper is about developing the help conversational agent for a user to find the information that he wants through conversation with agent. The proposed method is based on the pattern matching of artificial intelligence, not natural language processing. If a user inputs any sentence, the help conversational agent responds to the sentence through preprocessing and pattern matching with knowledge. The knowledge is built with the XML format. With the approximate pattern matching, the agent picks up the appropriate response with some degree of similarities. At the experiment, some different sentences with the same meaning have been entered, then the agent recognized them as the same pattern, and it made a correct answer.

1. 서론

인터넷이 급속도로 성장함에 따라 인터넷의 웹사이트 숫자도 늘어나고, 많은 정보가 등록되고 있다. 인터넷에 많은 정보가 등록되면서, 사용자의 정보획득을 위해 다양한 하이퍼링크를 제공하고 있으나, 대개 웹사이트의 구조

는 정보제공자가 통일된 표준 없이 구성하기 때문에 인터넷 사용자들이 낯선 웹사이트에서 원하는 정보를 쉽고 빠르게 찾는 것이 쉽지 않은 경우가 많다. 이를 해결하기 위해, 웹사이트 내에 전문(full-text) 검색엔진을 도입하여, 사용자가 자신이 원하는 정보를 대표하는 키워드(keyword)를 입력함으로써, 검색을 할 수 있도록 하고 있다[8]. 그러나 정보검색 시스템의 대상문서의 양이 증가하면, 키워드 검색결과로 나오는 문서의 양 또한 늘어나게 된다. 그리고 결과로 나온 문서들을 일일이 읽어야 사용자가 원하는 답을 찾을 수 있는 번거로움이 있다. 여러 연구에서, 사용자의 의도를 나타내는 테에는 키

* 연세대학교 컴퓨터과학과
키워드 : 대화, 에이전트, 패턴매칭
주소 : 120-749 서울시 서대문구 신촌동 134
연세대학교 컴퓨터과학과 소프트 컴퓨팅 연구실
전화 : 02-2123-3877 Fax : 02-365-2579
E-mail : adus@candy.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

워드보다는 자연어(natural language)를 사용하는 것이 훨씬 효과적이며 밝혀져 있다[5,12]. 본 논문에서는 사용자가 원하는 정보를 키워드가 아니라, 자연어로 된 문장을 입력하게 함으로써, 사용자에게 좀 더 친숙한 인터페이스를 제공한다. 입력받은 문장을 분석하여, 사용자가 원하는 답을 문서들의 집합이 아니라, 구체적인 문장으로써 답해주는 대화형 에이전트를 개발한다. 특정 도메인에서 사용할 수 있도록 지식을 정의하고, 사용자가 입력한 문장을 한글 전처리과정과 매칭을 통해 구축된 지식 중 가장 근사한 답을 낼 수 있도록 한다.

2. 관련연구

최초의 대화형 로봇으로는 1966년 MIT대학의 Weizenbaum교수가 만든 ELIZA가 있다[9]. ELIZA는 사람과 기계사이의 자연어 의사소통을 연구하기 위해 만들어진 프로그램으로, 사용자가 던진 문장을 분석하여, 대답에 필요한 키워드를 추출한 후, 이미 프로그램 되어 있는 문장에 키워드를 치환하여 대답한다.

ALICE (Artificial Linguistic Internet Computer Entity)라는 대화형 로봇은 자연스러운 대화를 이끌어 내기 위하여, 패턴매칭 기법을 사용하였다[13]. 대화에 사용되는 지식, 즉 패턴은 AIML (Artificial Intelligence Markup Language)라는 XML 형태의 파일로 저장되어 있다. AIML 파일에 사용자가 물어볼 만한 질문과 그에 해당하는 대답을 기술한다. ALICE는 사용자가 물어보는 문장과 AIML 파일에 패턴으로 기술된 문장을 비교하여, 하나의 와일드 카드(*)를 허용하는 범위 내에서 정확하게 매칭 되는 것만을 선택하여 답을 해주는 시스템이다.

인터넷에서 사용되고 있는 검색엔진에 많은 진보가 이루어지고 있다. 특히, 사용자 질의를 자연어로 입력받고, FAQ에 기반하여 검색결과를 내는 검색엔진도 있다. 대표적인 검색엔진으로 Askjeeves.com이 있는데, FAQ 스타일 검색엔진의 장점은 키워드기반 검색엔진이 수천 개의 결과를 내는 반면, 상호작용과 카탈로그 방식을 통해 더 정확한 결과를 내 준다는 것이다. 이러한 스타일의 검색엔진은 웹 기반 기술 지원과 같은 제한된 도메인에서 광범위한 성공을 보여주고 있다[11].

국제적인 정보검색평가대회인 TREC (Text REtrieval Conference)에서는 1999년 TREC-8에서 Question-Answering Track(Q&A Track)을 내 놓았다[7]. Q&A Track은 대용량의 문서집합에서 사용자가 원하는 정보를 문서단위가 아닌 작은 문단이나 문장 또는 정확한 단어로 제시해 주기 위한 것이다. 사용자는 자연언어로 질문을

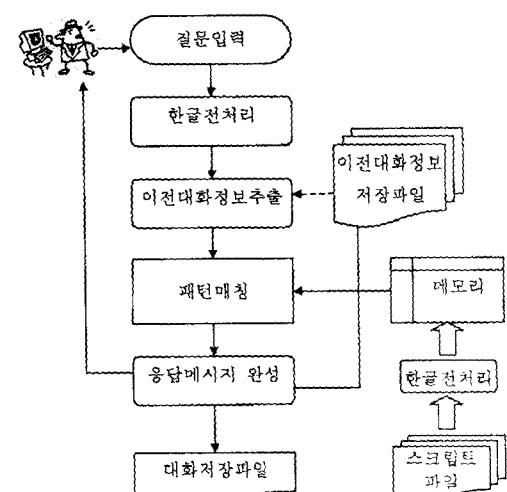
하게 되고, 시스템은 이에 대한 해답을 정확하게 얻어내는 것을 목적으로 한다. 그러나 질문문장의 의미를 정확하게 파악하는 일은 아직도 쉽지 않은 일이고, 그 답을 찾는 일 또한 어려운 일이다. 따라서, 질문의 의미를 정확하게 파악하기보다는 질문에 나타나는 패턴을 찾아서, 구축된 지식과 패턴 매칭을 하는 방법을 사용하기도 한다.

이와 같은 맥락에서, 사용자가 입력한 자연어 문장을 순차적 패턴 매칭 기법을 이용하여 대화형 로봇을 개발한 바 있다[1]. 이 로봇은 "패턴-답변" 형태로 저장되어 있는 스크립트 파일을 지식으로 하여, 사용자에게 알맞은 대답을 해주고 대화를 이끌어 가는 대화형 로봇이다. 그러나 순차적 패턴 매칭 방법은 사용자가 입력한 문장 패턴과 지식으로 구축되어 있는 패턴들을 순차적으로 비교함으로써, 어순만 약간 바뀌어도 같은 패턴으로 인식하지 못하는 문제가 있었다.

본 논문에서는 이러한 문제를 해결하고자, 순차적 패턴 매칭 기법에서 더 나아가, 한글 전처리과정과 지식처리를 이용하여, 사용자의 입력문장과 지식으로 구축된 패턴이 어순이 다르더라도, 근사하게 매칭시켜 답을 해줄 수 있는 대화형 에이전트를 개발한다.

3. 대화형 도우미 에이전트

대화형 도우미 에이전트의 구조는 (그림 1)과 같다. 우선 사용자가 질문을 입력하면 한글 전처리기는 사용자의 질문을 규격화하고 개념어를 추출한다. 이전대화 정보추출에서는 이전 대화 정보를 추출하여 문맥유지를



(그림 1) 대화형 도우미 에이전트의 구조

수행한다. 패턴매칭에서는 사용자의 입력정보와 메모리 상에 올려진 질문-대화 패턴을 비교하여 매칭되는 것을 찾는다.

응답 메시지는 사용자에게 돌려지며 이러한 대화 정보는 대화저장 파일에 남겨진다.

3.1 한글 전처리

대화형 도우미 에이전트에서 이루어지는 문장은 한국어 대화체문장으로서, 문어체에 비해 단어의 축약, 조사의 생략, 수정 또는 반복발화, 간투어 등의 특성으로 인해 분석하는데 많은 문제점을 포함하고 있다[2,4]. 이러한 대화체 문장을 분석하기 위한 가장 대표적인 방법이 개념기반 분석기법이다. 개념기반 분석기법은 강건성을 가장 큰 장점으로 하며, 비문법적인 요소를 많이 포함하고 있는 자연발화 처리에 유리한 기법 중 하나로 평가되고 있다[6,10].

본 논문에서는 특정 도메인내에서 발화될만한 문장을, 그리고 대화를 통해 남겨진 문장을 분석하여, 문장 내의 의미 있는 단어들을 개념으로 정의하고, 개념을 중심으로 지식을 만들어 매칭을 수행하게 한다.

한글 전처리에서는 사용자가 입력한 문장을 시스템이 다룰 수 있는 문장으로 규격화하는 작업을 수행한다. 한글에서는 영어와 달리 조사와 용언의 어미 등 변화가 복잡하므로, 주된 의미에 영향을 미치지 않는 조사나 보조용언, 동사의 어미들을 제거하여 주된 의미만 가지고 매칭을 한다. 한글 전처리 과정은 부호(따옴표, 쉼표, 마침표, 느낌표, 물음표 등) 제거, 동의어처리, 형태소분석, 개념화로 구성된다. 동의어 처리부분에서는 동의어목록 을 가지고 같은 의미를 가진 단어를 지식에 쓰인 단어로 치환한다. 동의어 처리 후 형태소 분석을 하여 개념 을 뽑아내게 된다. 문장의 의미에 가장 많은 영향을 주는 명사, 용언의 어근, 의문사만을 추출한다. 의문사는 "누구", "무엇", "어디", "몇", "어떻게", "왜" 등으로 형태 소 분석시 대명사나 부사 등으로 분석되지만, 질문문장에서 중요한 역할을 하므로, 따로 정의하여 개념으로 추출될 수 있도록 한다. 즉 "주소록은 어떻게 만드나요?"라는 문장은 부호제거 단계를 거쳐 "주소록은 어떻게 만드나요"로 변하고, 동의어 처리 후 형태소분석을 하게 된다. (그림 2)는 형태소 분석을 한 결과이다[3].

주소록은: 주소록/NNIN2+은/PPAU
어떻게 : 어떻게/ADCO
만드나요 : 만들/VBMA+나요/ENTE

(그림 2) 형태소분석

형태소분석 결과를 바탕으로 문장을 개념화하면 "주소록", "어떻게", "만들"이라는 3가지의 단어가 나오게 된다. '주소록'은 보통명사(NNIN2), '어떻게'는 부사(ADCO)지만, 의문사로 취급하고, '만들'은 동사(VBMA)이므로 개념어로 추출된다.

3.2 지식처리

도우미에이전트가 대화를 나눌 수 있는 지식은 XML 형식을 가진 스크립트파일에 저장된다. 지식의 기본단위는 FAQ에서 쓰이는 각각의 "질문-답변"의 쌍이다. 지식은 질문패턴, 답변, 관련 URL로 구성된다. 질문패턴은 사용자가 입력할 만한 질문으로 같은 의미를 가진 질문을 여러 개 기술할 수 있다. 이렇게 함으로써, 같은 의미지만 다양하게 나타나는 질문들을 하나의 패턴으로 처리할 수 있다. 답변은 질문패턴에 해당하는 답변이며, 대화의 지루함을 없애기 위해 다양한 형태의 답변을 넣을 수 있다. <random></random>태그를 사용하여, 여러 개의 답변중 하나를 임의로 추출하여 답변할 수 있는 것이다. 관련 URL은 답변과 관련된 URL로 도우미에이전트 사용자 인터페이스의 특정 부분에 표시함으로써, 사용자의 이해를 도울 수 있다.

스크립트파일에 저장되는 질문패턴은 자연어로 된 완전한 문장이며, 메모리에 로딩될 때에는 한글 전처리 과정을 거쳐 개념화한 형태로 바뀌게 된다. (그림 3)은 스크립트 파일의 예이다.

```
<qna>
<pattern>
<li>보낸 메일을 취소할 수 있나요?
<li>보낸메일을 취소하는 법은?
</pattern>
<answer>
인터넷메일은 전송된 메일을 PC통신에서처럼 전송취소를 할 수가 없습니다.
</answer>
<link>http://candy.yonsei.ac.kr/mail/faq/faq1.html</link>
</qna>
```

(그림 3) 스크립트 파일의 예

(그림 4)는 도우미 에이전트가 사용하는 스크립트 파일을 BNF(Backus-Naur Form)로 나타낸 것이다. BNF는 기본적으로 치환구조를 가지고 있다. 왼쪽의 성분이 오른쪽의 성분들로 대치되는 것이다. 이때 <>는 하나의

```

<doumi파일> ::= <qna><qna>*
<qna> ::= <pattern><pattern>*<previous><answer>
<question> ::= <PATTERN> <질문패턴> </PATTERN>
<previous> ::= <PREVIOUS> <이전도우미답변> </PREVIOUS>
<response> ::= <RESPONSE> <response_comp> </RESPONSE>
<response_comp> ::= <random> | <answer> | <anchor>
<random> ::= <RANDOM> <li><li>* </RANDOM>
<li> ::= <L> <answer> </L>
<answer> ::= <한글> | <reserv_pron> | <한글>
<link> ::= <LINK> <URL> </LINK>
<질문패턴> ::= <한글> | * | <한글>
<이전도우미답변> ::= <한글>

```

(그림 4) 스크립트 BNF

카테고리를 나타내고 *는 반복회수를 결정한다. 즉, A*는 A가 0번 이상 반복되는 것을 나타낸다. A|B는 A또는 B가 나타남을 의미한다.

3.3 선행 대화 정보 추출

BNF에 보면, <answer></answer> 태그 사이에 <reserve_pron>이 있는데, <reserve_pron>은 일종의 대명사를 지칭하는 것으로 <set_it>, <set_he>, <star> 등이 있다. 로봇과 사용자가 대화하는 중간 중간에 로봇은 사용자가 대답한 문장 중에서 "그것", "그사람"등에 해당하는 것을 저장해 놓을 수 있다. 다음의 스크립트를 살펴보자.

```

<qna>
<pattern>에디슨은 누구입니까?</pattern>
<answer>
<set_he>에디슨</set_he>은 미국의 발명가로 평생 미국에서 살면서 연구활동을 했습니다. 에디슨은 세계에서 가장 많은 것을 만든 발명가로 1천 93개의 발명품에 대한 특허를 얻었습니다.
</answer>
</qna>

```

(그림 5) 스크립트 예

위의 문장에서 <answer></answer>부분을 살펴보면, "에디슨" 단어의 양 옆에 <set_he></set_he>을 볼 수 있다. 이것은 "에디슨"이란 사람 이름을 "그사람"이라는 대명사로 지정해서 메모리에 기억해 놓기 위함이다.

사용자가 "에디슨은 누구입니까"란 질문 후에 "그의 대표적인 발명품은 무엇이 있나요?"라고 질문을 했다면, 로봇은 "그"를 "그사람"이라는 대명사로 인식하고, "에디

슨"으로 치환하여 적절한 패턴을 찾는 작업을 한다.

<set_it>태그는 사람이 아니라 사물을 대명사로 지정해서 쓰려는 태그이며, <set_he>와 같은 방식으로 사용된다. <star>는 위에서 언급한 것처럼, 사용자의 문장 중 와일드카드(*)에 해당하는 단어를 기억하고 있다가 대답시에 치환하여 사용한다. 이런 식으로 이전 대화문장과 대명사를 이용하여, 문맥을 어느 정도 유지하면서 사용자와 대화를 이끌어간다.

사용자가 대화형 도우미에이전트와 대화를 주고받을 때, 사용자가 원하는 답을 주려면, 단발성의 질문에 대해서뿐만 아니라, 그 전에 했던 대화와의 문맥유지가 필요하다. 본 시스템에서는 이를 위해서, 사용자와 대화했던 직전의 대화를 저장하고, 다음 대화의 패턴 매칭에 사용될 수 있도록 한다. 바로 전 대화의 내용에 필요한 질문패턴이 있다면, 스크립트에서는 <previous></previous> 태그를 이용하여 기술하고, 패턴 매칭시에 활용하게 된다. 예를 들어, 다음 대화를 살펴보자.

도우미 : 당신의 이름은 무엇입니까?
사용자 : 내 이름은 홍길동이야.
도우미 : 반갑습니다. 홍길동님! 무엇을 도와드릴까요?

(그림 6) 대화의 예

도우미가 위와 같은 대답을 하기 위해서 다음과 같이 스크립트가 작성되었다. 위의 대화에서, 사용자가 입력한 문장 "내 이름은 홍길동이야"는 아래 스크립트의 패턴 중 "이름은 *이야"에 매칭이 되고, 도우미에이전트가 직전에 했던 대화문장 "당신의 이름은 무엇입니까?"와 <previous></previous>태그 사이의 문장이 일치하여, <answer></answer>부분에 명시된 답을 출력하게 된다. 여기서 <star>는 <reserve_pron>중의 하나로, 사용자가 입력한 문장에서 패턴의 *에 해당하는 단어를 뜻하며, 대답 생성시 해당 단어로 치환된다.

```

<qna>
<pattern> 제 이름은 *입니다. </pattern>
<pattern> 이름은 *이야</pattern>
<pattern> * 입니다. </pattern>
<pattern> * </pattern>
<previous> 당신의 이름은 무엇입니까? </previous>
<answer>반갑습니다. <star>님! 무엇을 도와드릴까요?
</answer>
</qna>

```

(그림 7) 스크립트의 예

3.4 패턴매칭

사용자가 인터페이스를 통해 입력한 문장과 메모리에 로딩되어 있는 질문 패턴들과 비교하는 작업을 한다. 패턴 매칭 전에 사용자의 입력문장은 한글 전처리 과정을 통하여 개념화되어 있고, 마찬가지로 지식도 스크립트 파일에서 한글 전처리 과정을 거쳐 개념화되어 있다. 사용자가 입력한 문장의 개념단어집합 $C = \{c_1, c_2, \dots, c_n\}$, 지식의 질문패턴집합 $P = \{P_1, P_2, \dots, P_k\}$, $P_j = \{pc_1, pc_2, \dots, pc_m\}$ 라고 하면, 입력 문장에 대한 질문패턴 P_j 와의 스코어는 다음과 같이 구한다.

$$S_j = \sum_{i=1}^k exst(c_i, P_j)/m + a \times g(D)$$

$exst(c_i, P_j)$ 는 P_j 의 집합내에 c_i 와 같은 단어가 있으면 1을 반환, 그렇지 않으면 0을 반환하는 함수이다. m 은 패턴 j 에 나타난 개념의 수이고, $g(D)$ 는 `<previous>` `</previous>` 태그가 명시된 경우에 직전대화에서 도우미 에이전트가 말한 문장과 previous에 명시된 문장이 같을 때 1을 반환하는 함수이다. 상수 a 를 높은 값으로 설정함으로써, 문맥을 이용한 대화가 이루어질 수 있도록 한다.

입력문장에 대해 모든 질문패턴과의 스코어를 구한 후 가장 높은 점수를 획득한 질문패턴을 선택하여, 그에 해당하는 응답을 내보내게 된다. 이렇게 하면, 패턴의 개념 단어들과 입력문장의 단어들의 순서가 스코어에 영향을 미치지 않는다. 이것은 한국어문장내에서 각 단어의 어순이 자유로운 점을 고려한 것이다. 그러나, 계산된 최대 스코어가 너무 작으면, 매칭의 정확도가 떨어지므로, 어느 수준까지 정확도를 고려할 것인지를 정해야 한다. 본 논문에서는 2/3를 넘는 경우에만 매칭이 성공한 것으로 하였다. 2/3라는 수치는 경험적으로 결정하였다.

3.5 응답메시지 생성 및 대화저장

패턴 매칭을 통해 적절한 <패턴-대답>의 쌍이 찾았으면, 사용자에게 보낼 응답메시지를 완성해야 한다. <패턴-대답>에서 <대답>은 복수 개일 수도 있고, 한 개일 수도 있다. <random></random>태그로 뭉친 복수 개인 대답은 그 중에서 한가지의 대답을 임의로 뽑아서 사용자에게 대답을 해준다. <reserve_pron>이 있는 경우에는 해당 대명사를 메모리에서 읽어서 치환하여 대답을 한다. <link>태그가 있으면, 태그에 명시된 웹 페이지를 특정 부분에 표시한다.

패턴 매칭을 통해 적절한 <패턴-대답>의 쌍을 찾지 못한 경우에는, 답변을 준비하지 못했다는 예러 메시지를 출력한다.

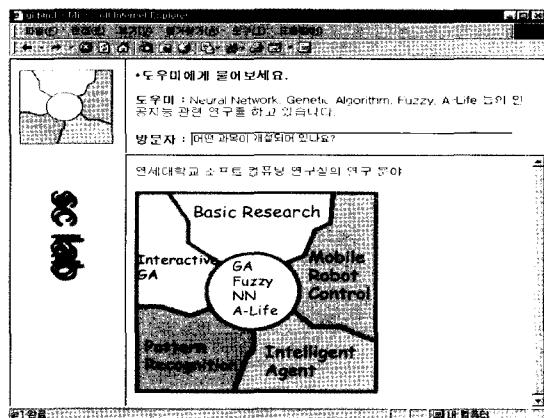
도우미에이전트와 사용자가 주고받은 대화를 대화 서

장 파일에 저장한다. 향후에 도우미에이전트 관리자가 대화 저장 파일에 기록된 대화를 바탕으로, 도우미 에이전트의 지식을 추가하거나 수정할 수 있도록 한다. 즉, 저장된 대화중 사용자의 질문이나 대화에 관해 에이전트가 잘못 해석하거나, 입력되지 않은 패턴이 있으면, 도우미 에이전트의 관리자가 스크립트파일을 수정할 수 있다.

4. 실행결과

4.1 사용자 인터페이스

본 시스템은 특정 사이트의 한 프레임에 위치하여, 사용자가 문장을 입력할 수 있는 텍스트 박스와 도우미에이전트가 대답을 할 수 있는 공간으로 이루어져 있다. 스크립트 파일에 <link>가 명시되어 있으면 오른쪽 아래 프레임에 Link에 명시된 URL의 페이지가 다스플레이된다.



(그림 8) 도우미에이전트 UI

4.2 대화의 예

연구실의 홈페이지를 소개하는 도우미에이전트와 사용자와의 대화에서 일어날 수 있는 질문에 대해 도우미에이전트가 축적된 지식을 바탕으로 하여, 정확한 대답을 낼 수 있는지 살펴보자.

```
<qna>
<pattern> 연구실의 위치는 어디입니까? </pattern>
<pattern> 연구실은 어디에 있습니까? </pattern>
<answer> 연세대학교 제3공학관에 있습니다. </answer>
</qna>
```

(그림 9) 스크립트의 예

위와같이 대화형 도우미에이전트의 지식이 구축되어 있다면, 위 질문패턴들은 한글 전처리과정을 거쳐 메모리에 로딩될 때 다음과 같이 저장된다.

패턴 1 - "연구실의 위치는 어디입니까?"

-> "연구실", "위치", "어디"

패턴 2 - "연구실은 어디에 있습니까?"

-> "연구실", "어디", "있"

다음의 사용자 질문에 대해, 위의 지식과 매칭이 일어날 수 있는지 살펴보자.

질문 1: 연구실이 어디 있나요?

질문 2: 어디에 연구실이 있습니까?

질문 3: 연구실 위치는?

각 질문과 패턴과의 스코어를 계산해보자.

질문1, 질문2 : 한글 전처리 과정을 끝낸 후의 개념
집합-> { "연구실", "어디", "있" }

패턴 1과의 스코어 : 2/3

패턴 2와의 스코어 : 3/3

질문3 : 한글 전처리 과정을 끝낸후의 개념집합 →
{ "연구실", "위치" }

패턴 1과의 스코어 : 2/3

패턴 2와의 스코어 : 1/3

질문 1과 질문 2는 개념의 위치만 바뀌어 대화가 이루어진 것으로, 개념집합은 같게 된다. 위와 같이 스코어가 계산되어서, 질문 1,2,3은 위에 기술된 스크립트의 지식패턴에 매치되어, 그에 해당하는 대답을 도출하게 된다. 실험결과 특정 도메인에서 다양한 질문에 대해 제시한 예문과 비슷한 결과를 얻을 수 있었다. 그럼 10은 대화의 예와 사용한 지식베이스를 보여준다.

5. 결론 및 향후연구

본 논문에서는 특정 도메인에 대해 사용자의 질문에 대답하는 대화형 도우미 에이전트를 제안하였다. 질문 문장이 한국어로 된 대화체 문장이라는 것을 감안하여 한글 전처리 과정에서, 형태소분석과 동의어 사전을 통해 질문을 개념화하였다. 지식 생성시에는 동일한 뜻을 가진 질문을 여러 형태로 입력할 수 있게 함으로써 융통성을 보였고, 또한 시스템 운영 중에 생성된 로그를

남김으로써 도우미에이전트 관리자가 자주 질문되는 내용에 관한 지식을 생성할 수 있다. 매칭스코어 계산시 입력문장의 개념단어들과 패턴의 개념단어들과의 순서를 고려하지 않음으로써, 한국어 어순의 자유로움을 고려하였다. 실험결과 특정 도메인에서 다양한 질문에 대해 제시한 예문과 비슷한 결과를 얻을 수 있었다.

본 논문에서 제안된 시스템은 질문의 정확한 의미분석을 하기보다는, 질문패턴을 여러 가지로 입력함으로써, 계산 부하를 줄이면서도 사용자가 원하는 답을 뽑아낼 수 있도록 하였다. 이는, 특정 도메인에서의 많은 양의 질문패턴 데이터가 있을 때 효과적으로 성능을 발휘할 수 있을 것이다. 도우미에이전트의 지식이 되는 스크립트 파일을 일일이 사람이 작성하기에는 한계가 있으므로, 향후에는 자동으로 도우미 에이전트의 지식을 축적할 수 있는 방향이 연구되어야 하겠다. 또한, 도우미 에이전트를 운용하면서 나온 질문 로그를 분석하여, 대답하지 못한 질문에 대한 다양한 질문패턴을 추천해 줄 수 있는 연구도 진행될 예정이다. 또한 사용자의 의도를 추론하여 적절한 대화를 수행하기 위한 베이지안 모델링 방법도 연구를 진행할 예정이다.

본 연구는 pattern matching을 이용한 대화형 에이전트를 설계하고 한국어에 관해 실용성을 보이는 것을 목표로 하였다. 최근, 정보의 양이 폭발적으로 증가하면서 효과적으로 정보를 찾는 방법이 중요해졌으나 현재의 정보검색 시스템은 몇몇을 제외하고 모두 키워드를 이용한 질의 방식으로 사용자의 요구사항을 정확하게 표현하기 어려운 문제를 가지고 있다. 이러한 이유로 대화를 이용한 질의는 가장 이상적인 정보시스템의 검색방법으로 생각된다. 자연어 처리 분야의 연구가 이러한 문제를 해결하기 위해 다양한 기여를 해왔지만 사람의 말이 가진 복잡성으로 인해 방법의 난해성과 낮은 성능이라는 문제를 지니고 있다. Pattern matching을 이용한 대화형 에이전트의 설계는 단순하지만 효과적으로 사용될 수 있는 범용적 시스템을 개발하는데 유용하게 이용될 수 있다.

본 연구의 우수성을 보이기 위해 보다 많은 실험과 정량적인 척도에 의한 평가가 요구된다. Precision과 recall은 정보검색 분야에서 가장 다양하게 사용되는 평가방법이다. P-R 척도이외에 정보검색 시스템의 성능을 하나의 기준으로 평가하는 복합 척도(composite measure)가 이용된다. Swets는 기존의 정보검색 방법에 문제를 느껴 통계적인 결정 이론을 이용한 평가방법을 제시하였다. 다양한 평가결과를 평균하기 위해 micro-average 방법을 쓰거나 macro-average 방법을 사용한다. 사용자의 정보탐

도우미 : 무엇을 도와드릴까요?

사용자 : 이 웹사이트는 무슨 사이트입니까?

도우미 : 연세대학교 소프트컴퓨팅 연구실의 홈페이지입니다.

사용자 : 어떤 분야를 연구하나요?

도우미 : 인공지능을 연구합니다. 특히 Genetic Algorithm, Fuzzy, Neural Network, A-Life 등을 중심으로 여러가지 연구를 하고 있습니다.

사용자 : 어떤 과목이 개설되어 있습니까?

도우미 : 컴퓨터입문, 오토마타등의 과목이 개설되어 있습니다.

질문 패턴

```
<doumi>
<qna>
<question>이 웹사이트는 무엇에 관한 사이트 입니까?</question>
<question>이 사이트는 무슨 사이트입니까? </question>
<question>이 사이트는 무엇에 관한겁니까? </question>
<question>무슨 사이트입니까?</question>
<response>
연세대학교 소프트컴퓨팅 연구실의 홈페이지입니다.
</response>
</qna>
<qna>
<question> 어떤 분야를 연구하나요? </question>
<question> 무슨 분야를 연구하나요? </question>
<question> 어떤것을 연구하나요? </question>
<response>
인공지능을 연구합니다. 특히 Genetic Algorithm, Fuzzy, Neural Network, A-Life 등을 중심으로 여러가지 연구를 하고 있습니다.
</response>
<qna>
<question>
어떤 과목이 개설되어 있나요?
</question>
<response>
컴퓨터입문, 오토마타등의 과목이 개설되어 있습니다.
</response>
</qna>
</doumi>
```

(그림 10) 대화의 예

색에 소요된 질의어의 길이도 평가를 위한 척도로 사용될 수 있다. 사용자의 주관적인 평가를 바탕으로 한 subjective test도 평가방법으로 사용된다. 서로 다른 상황에서 검색의 효과성의 차이를 검증하기 위해서 중요성 테스트(significance test)를 수행한다. 현재의 정보검색

시스템의 평가는 P-R기준을 중심으로 통계적인 방법, 주관적인 테스트가 연구되고 있다. 다양한 정보검색 시스템의 평가 기준을 이용하여 본 대화형 에이전트의 성능을 객관적으로 평가하는 것을 향후 연구로 진행할 예정이다.

감사의 글

이 논문은 한국 학술 진흥재단의 연구과제(KRF 2000-005-H20002)에 의해 지원되었음

참고문헌

- [1] 김수영, 조성배, "순차적 패턴매칭 기법을 이용한 대화 형 도우미 에이전트," *한국정보과학회 2000 추계학술발표회*, Vol 27, No.2, pp 24~26. 2000.
- [2] 노서영, 정천영, 서영훈 "핵심개념 기반의 강건한 한국어 대화체 파싱," *한국정보처리학회 논문지*, 제6권, 제8호, pp. 2113~2123, 1999.
- [3] 윤준태, 이충희, 김선호, 송만석 "연세대 형태소 분석기 morany: 말뭉치로부터 추출한 대량의 어휘 데이터베이스에 기반한 형태소 분석" *한글 및 한국어 정보처리 학술대회* 1999.
- [4] 정천영, 임희동, 서영훈 "대화체 기계번역을 위한 중심어 기반 한국어 분석", *충북대학교 산업과학기술연구소 논문지*, 제 13권 제1호, pp.47~56, 1999.
- [5] A.Pollok and A.Hockley, "What's wrong with Internet Searching," D-Lib Magazine. <http://www.dlib.org/dlib/march97/bt/03pollock.html>
- [6] E. Levin and R. Pieraccini, "Concept-based Spontaneous Speech Understanding System," *Eurospeech'95*, pp. 555~558, 1995.
- [7] E. Voorheesz, The TREC-8 Question Answering Track Report, National Institute of Standards and Technology, pp 77, 1999.
- [8] J. Cho, H. Garcia-Molina and L. Page, "Efficient crawling through URL ordering", The 7th Int. WWW Conf. (WWW 98). Brisbane, Australia, 14~18, April 1998.
- [9] J. Weizenbaum, "ELIZA-A computer program for the study of natural language communication between man and machine," *Communications of the ACM*, Vol. 9, No. 1, pp 36~45, 1965.
- [10] L.J. Mayfield, M. Gavald'a, Y-H. Seo, B. Suhm. W. Ward, A. Waibel "Parsing real input in JANUS : A concept based approach to spoken language translation," *Proceeding of TMI95*, pp 196, 1995.
- [11] Q. Yang, H.F. Wang, J.R. Wen, G. Zhang, Y. Lu, K.F. Lee and H.J. Zhang "Towards a next generation search engine," *Proc. of the Sixth Pacific Rim Artificial Intelligence Conference, Melborne, Australia*, August, pp 5~15, 2000.
- [12] Y.J. Yang, L.F. Chien, and L.S. Lee, "Speaker intention modeling for large vocabulary mandarin spoken dialogues," *Proc. of Fourth Int. Conf. on Spoken Language*, 1996.
- [13] <http://www.alicebot.org>