

대화에서 응답 관계의 시각화

김경덕
위덕대학교 컴퓨터멀티미디어공학부
kdkim@mail.uiduk.ac.kr

Visualization of Relation among Turns on Conversation

Kyungdeok Kim
Division of Computer and Multimedia Engineering, Uiduk University

요 약

본 논문에서는 실시간 대화 행위에서 대화 메시지 사이의 응답 관계를 시각적으로 표현하는 방법을 제안한다. 제안한 방법은 기존 텍스트 기반 대화 방식과 트리 기반 대화 방식을 결합한 형태로서 대화 메시지의 일반적인 응답 관계뿐만 아니라, 기존 트리 기반 인터페이스에서 지원이 어려운 최근 수신 대화 메시지의 응답 관계를 시각화함으로써 대화자의 상호작용을 용이하게 한다. 이러한 방법은 기존 텍스트 방식의 테이블 구조에 트리 구조를 결합하여 대화에서 응답 관계를 명확히 구분한다. 제안한 방법의 구현은 XML과 DOM을 이용하여 대화 메시지와 대화 시스템을 구현하였으며, 응용 분야는 협업, 원격 교육, 온라인 게임 등이다.

1. 서론

컴퓨터를 이용한 실시간 대화 도구는 컴퓨터의 발전과 함께 꾸준히 사용되고 있으며, 인터넷의 발달로 그 활용도가 더욱 넓어지고 있다. 주로 교육과 상업적 용도로 이용되고 있으며, 인터넷과 이동 통신을 중심으로 다양한 관련 서비스 기술이 개발되고 있다. 이러한 관련 기술의 대부분이 비형식 대화를 위하여 개발되고 있으며, 사무 환경에서도 점차 그 영역을 확장하고 있는 중이다 [1, 2, 3].

현재 대화 시스템은 2D와 3D 인터페이스와 결합된 다양한 소프트웨어가 개발되고 있으나, 메시지의 교환 방식이나 표현 방식은 기존 텍스트 기반 대화의 표현 방식과 거의 차이가 없다. 이러한 기존의 대화 방식은 대화자의 메시지 도착 순서에 따라 대화 인터페이스에 나열됨으로써 대화에서 응답 관계의 파악하기 어렵다 [4, 5, 6]. 그러므로, 대화자는 메시지의 응답 관계를 직접 해석함으로써, 그 모호성에 따라 잘못된 의미의 해석과 수정을 위한 추가적인 메시지 전달이 요구된다[5]. 또한, 기존 대화 시스템에서는 대화 끼여들기, 응답 관계의 조직화, 의미 전달, 발언권 제어 등의 관리가 어렵다 [4]. 이러한 대화 시스템의 문제로 인하여 사무 환경에서 의사 결정 도구로 사용하기는 아직 어려운 상태이다. 그러므로, 대화자들간의 대화 메시지가 상호간의 응답에 의하여 이루어지는 상황을 시각적으로 나타내고, 동시에 다중 응답 메시지를 체계화하여 그 관계를 시각적으로 명확히 표현하는 방법이 필요하다.

대화에서 응답 관계의 시각화를 위한 관련 연구로는 Vronay의 Flow Chat[5]는 각 사용자별로 트랙을 지정하고 대화자가 작성한 메시지를 시간별로 인터페이스에 표

시한다. 각 대화자는 자신의 트랙에만 메시지를 작성하므로 다중 대화자간 메시지 전달 시에 다중 응답 관계를 구분하여 나타내기 어렵다. Smith의 Conversation Tree[4]는 트리 기반의 대화 인터페이스에서 각 노드를 이용하여 대화 메시지를 나타낸다. 트리 기반의 대화 인터페이스는 다중 대화자간 메시지들의 일반적인 응답 관계를 명확히 나타내지만, 어느 부분이 최근 대화 메시지의 응답 관계를 나타내는지 파악하기 어렵다. Viegas의 Visualizing Conversation[2]는 2D 인터페이스에서 서클을 대화자로 표시하고, 대화 메시지를 수신할 때 서클의 크기를 대화 메시지의 길이에 맞춰 크기를 조정하며, 시간의 경과에 따라 작아지게 표시한다. 그리고, 대화 메시지의 기록을 위하여 대화자별로 타임 라인을 이용하지만, 다중 질문에 대한 응답 관계를 나타내기는 어렵다. 그 외, 실제 사용중인 대화 시스템으로 세이클럽[8], 프리첼[9], 토마토넷[10], 가채탕[11], 매직챗[12] 등이 있으며, 대부분 텍스트 기반 대화 방식을 사용하고 사용자 캐릭터 이미지와 연관하여 대화 메시지를 나타낸다. 그리고, 2D 인터페이스를 이용한 매직챗[12] 등도 응답 관계를 명확히 나타내지 못한다.

아직 대화 시스템을 위한 대화 행위의 체계적인 표현 연구가 아직 미흡한 편이며, 2D 및 3D를 이용하는 대화 인터페이스는 사회 정보(social information: 대화 톤, 대화 시점, 메시지 스타일 등)를 부분적으로 지원하고 있지만, 대화의 체계적 표현은 아직 미흡한 편이다[2]. 그러므로, 본 논문에서는 대화의 메시지의 응답 관계를 시각화하기 위하여 텍스트 기반과 트리 기반 대화 인터페이스를 이용하여 대화 메시지의 일반적인 응답 관계와 최근 수신 대화 메시지의 응답 관계를 시각화하는 방법을 제

안한다.

본 논문의 구성은 다음과 같다. 제 2절에서는 대화 시스템의 구성을, 제 3절에서는 대화에서 응답 관계 시각화에 대하여 설명하며, 제 4절에서는 결론 및 향후 연구 방향을 기술한다.

2. 대화 시스템의 구성

대화 메시지의 응답 관계 시각화를 지원하는 대화 시스템의 구성은 다음 그림 1과 같다. 대화 시스템의 구성은 상호작용, 사용자 인터페이스, 정보전송 및 관리의 3계층으로 이루어지며, 각 계층의 구성 요소와 기능은 다음과 같다. 먼저, 상호작용 계층의 구성 요소는 대화 메시지를 입력하는 부분과 사용자의 이벤트를 수신하여 응답 관계를 표시하는 부분으로 구성된다. 인터페이스 부분은 텍스트 기반 및 트리 기반 대화 인터페이스로서 사용자에게 대화 메시지 응답 관계의 시각화를 지원한다. 정보 전송 및 관리 부분은 메시지 송수신 부분, 송수신된 메시지는 대화 메시지인지 또는 시스템 관리 메시지인지 분석하고 해당하는 기능(사용자 관리, 클라이언트 관리, 대화 메시지 관리)을 처리하는 부분, 클라이언트의 생성 및 관리 부분, 사용자 목록을 유지하고 관리하는 부분으로 이루어진다.

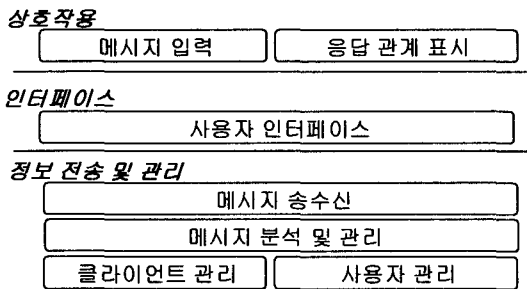


그림 1. 대화 시스템의 구성

3. 대화 응답 관계 시각화

기존 텍스트 기반 대화 인터페이스의 일반적인 구성은 대화자 목록, 대화 메시지 기록 및 표시 부분, 메시지 입력 부분으로 이루어진다[5]. 본 논문에서 제안한 메시지 응답 관계 시각화를 위한 대화 인터페이스는 텍스트 및 트리 기반의 대화 인터페이스를 사용한다. 텍스트 기반 대화 인터페이스에 기록되는 각 대화 메시지는 트리 기반 대화 인터페이스에 기록되는 각 노드의 대화 메시지와 일대일의 대응 관계를 유지한다. 사용자가 텍스트 기반의 대화 인터페이스에서 수신된 특정 대화 메시지를 선택하면 자동적으로 트리 기반 대화 인터페이스의 해당 노드의 대화 메시지 응답을 위한 노드를 추가하고 입력을 대기한다. 대화자가 어느 대화 인터페이스를 이용하든 입력된 대화 메시지는 두 대화 인터페이스에 동시에 추가된다. 이러한 인터페이스의 이용으로 기존 트리

기반 대화 인터페이스에서 문제점인 최근 수신 메시지의 응답 관계 시각화를 지원한다.

대화 메시지는 XML 기반 메시지를 이용한다. XML은 정보 표현을 위한 자연스런 형태를 제공한다. 대화에 사용되는 메시지의 DTD(Document Type Definition)는 다음과 같다.

```
<!ELEMENT MESSAGE (PID)>
<!ELEMENT PID EMPTY>
<!ATTLIST MESSAGE from CDATA #REQUIRED>
<!ATTLIST MESSAGE to CDATA #REQUIRED>
<!ATTLIST MESSAGE id CDATA #REQUIRED>
<!ATTLIST MESSAGE reply CDATA #REQUIRED>
<!ATTLIST PID pid0 CDATA #REQUIRED>
.....
<!ATTLIST PID pidn CDATA #REQUIRED>
```

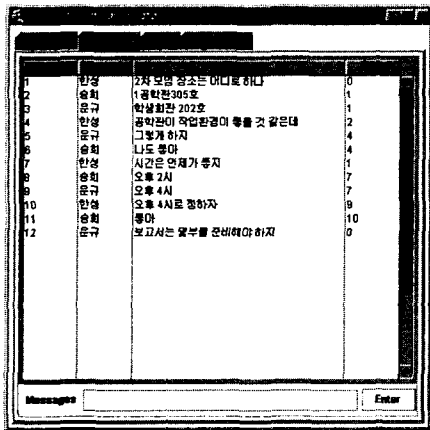
여기에서, 엘리먼트 노드 MESSAGE는 대화자가 입력하는 텍스트 노드와 엘리먼트 노드 PID로 구성되며, 엘리먼트 노드 MESSAGE는 속성으로 발송자(from), 수신자(to), 메시지번호(id), 응답번호(reply)를 가지며, 엘리먼트 노드 PID는 응답 관계를 지원하는 정보로서 현재 메시지의 상위 응답 메시지들의 엘리먼트 노드 MESSAGE의 속성 값 id로서 pid_n(n=0,..., n)을 가진다. 여기서, n은 상위 엘리먼트의 개수이다. 이 pid_n의 값은 메시지를 트리 기반 대화 인터페이스에서 적절한 위치를 지정하여 응답 관계를 유지하기 위하여 사용한다. 즉, 응답 메시지를 입력할 노드를 트리 기반 인터페이스에 추가하기 위하여 pid_n의 값을 사용하여 트리 기반 인터페이스의 트리 구조를 탐색하기 위한 정보로 이용한다. 탐색된 노드에 자식 노드를 추가하고 대화자의 입력을 대기한다. 다음 두

```
프로시저어 TextBasedConversation()와 TreeBased
Convertsaton()는 제안한 방법을 위한 주요 프로시저어
로서 텍스트 및 트리 기반 인터페이스에서의 대화 메시
지와 대화자 이벤트 입력 처리를 기술한다.
Procedure TextBasedConversation()
{
  if (User inputted a message in text field) {
    Set up attributes of the message.
    Attach the message to text-based interface.
    Attach the message as children of root node
    to tree-based interface.
  }
  if (User clicked an inputted message) {
    Decision a location for inserting a node in the
    tree-based interface as reply of a selected message
    using pid attribute's values of the selected message.
    Transfer to tree-based interface.
    Generation of a node for reply.
    Wait user's input.
  }
}
```

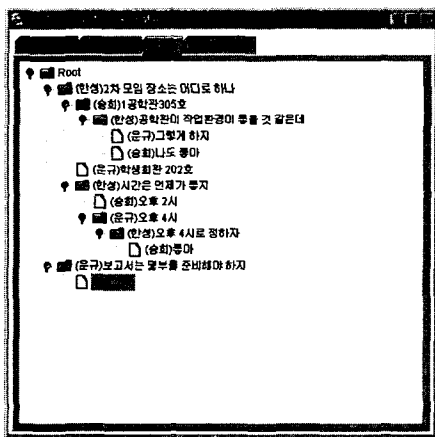
```

Procedure TreeBasedConversaton()
{
  if (User selected a node in the tree) {
    Generation an empty node under the selected node.
    Wait user's input.
  }
  if (User finished writing a message) {
    Set up attributes of the message.
    Attach the message beneath last message
    in the text-based interface.
  }
}
    
```

다음 그림 2는 구현한 대화 인터페이스로서 텍스트 및 트리 기반 인터페이스를 나타낸다. 구현 환경은 JDK 1.4 와 Kawa 5.0을 이용하였다.



(a) 텍스트 기반 인터페이스



(b) 트리 기반 인터페이스

그림 2. 대화 인터페이스

4. 결론

본 논문에서는 대화에서 응답 관계를 시각화하는 방법을 제안하였다. 제안한 방법은 응답 관계를 시각화하기 위하여 기존 텍스트 기반 인터페이스와 트리 기반 인터페이스를 결합하여 메시지의 일반적인 응답 관계와 기존 트리 기반 인터페이스에서 제공하기 어려운 최근 수신 메시지의 응답 관계의 시각화를 보였다. 이러한 시각화는 대화 흐름을 명확히 표현함으로써 대화자간에 의사소통을 더욱 명확히 지원한다. 이러한 대화를 지원하기 위하여 대화 메시지는 XML를 기반으로 작성하였으며, XML의 특성에 의하여 메시지에 다양한 특성의 추가와 메시지 수정 및 관리가 용이하다.

그러므로, 앞으로의 연구 방향은 대화 메시지에 사회 정보를 추가하여 사용자 대화 참여 상태의 시각화이다.

참고 문헌

- [1] C. Charlton, C. Little, R. Lloyd, S. Morris, and I. Neison, "Good Business Practice Needs Good Communications - New Generation Chat Software for Real-time Discussion," Proc. of the 10th Int. Workshop on Database & Expert systems Applications, 1999.
- [2] J. Donath, K. Karahalios, and F. Viegas, "Visualizing Conversation," Proc. of the 32nd Hawaii Int. Conf. on System Sciences, pp. 1-9, 1999.
- [3] S. Ortiz Jr., "Instant Messaging: No Longer Just Chat," IEEE Computer, Vol. 34, No. 3, pp. 12-15, 2001.
- [4] M. Smith, J. Cadiz, and B. Burkhalter, "Conversation Trees and Threaded Chats," Proc. of the ACM2000 Conf. on CSCW, pp. 97-105, 2000.
- [5] M. O. Thirunarayann and A. Perez-Prado, "Structured Chat," ACM Ubiquity, Vol. 1, Issue 38, 2000.
- [6] M. O. Thirunarayann, "Cutting Down on Chat Confusion," ACM Ubiquity, Vol. 1, Issue 45, 2001.
- [7] D. Vronay, M. Smith, and S. Drucker, "Alternative Interfaces for Chat," Proc. of the 12th Annual Symposium on User Interface Software and Technology, pp. 19-26, 1999.
- [8] 세이 클럽, <http://www.sayclub.com>
- [9] 프리챌, <http://www.freechal.com>
- [10] 토마토넷, <http://www.tomatonet.com/chat2>
- [11] 가채팅, <http://www.gachat.com>
- [12] 매직챗, <http://www.mhouse.net/mcworld>