

복잡한 서비스 자동생성을 위한 행동 네트워크기반 웹서비스 결합

정명철, 김경중, 조성배
연세대학교 컴퓨터과학과
mcjung@sclab.yonsei.ac.kr, kjkim@cs.yonsei.ac.kr, sbcho@cs.yonsei.ac.kr

Web Service Composition based on Behavior Network for Automatic Construction of Complex Service

Myung-Chul Jung, Kyung-Joong Kim, Sung-Bae Cho
Dept. of Computer Science, Yonsei University

요 약

방대한 양의 웹 정보를 적절히 가공하여 사용자가 원하는 서비스를 생성하기란 쉽지 않다. 웹서비스는 사용자가 원하는 적절한 정보를 표준화된 인터페이스를 통해 처리할 수 있는 기능을 제공하고 결합을 통해 보다 복잡한 서비스를 생성할 수 있도록 해준다. 오프라인에서 물품을 사기 위해 상점에 들어갈 때 점원의 도움을 받는 것처럼 웹서비스도 상황에 맞춰 유연하게 제공되어야 한다. 그러나 일반 웹서비스 생성은 사용자의 정보를 능동적으로 판단하지 않고 정적인 규칙에 의존한 결합이 주를 이루고 있다. 본 논문의 서비스 에이전트는 행동 네트워크를 사용하여 사용자에게 적절한 서비스를 제공한다. 행동 네트워크는 외부환경과 내부목표를 바탕으로 각각의 행동들이 서로 연결 관계를 가지고 있다. 각각의 행동의 선택은 다른 행동과 외부환경에 영향을 주기 때문에 동적으로 사용자의 환경과 입력이 변하여도 적절히 대처하며 서비스 선택을 할 수 있다. 행동 네트워크를 바탕으로 동적 서비스 생성 시스템을 만들었고, 이를 이용하여 Amazon 도메인에서 능동적인 서비스 제공이 가능함을 보였다.

1. 서 론

인터넷상의 정보가 폭발적으로 증가하면서 사용자가 자신이 필요로 하는 적절한 정보나 서비스를 찾아서 사용하기란 쉬운 일이 아니다. 이를 해결하기 위해 능동적인 서비스 생성에 관한 연구가 진행되고 있다[1]. 본 논문에서는 웹서비스를 통해 제공되는 다양한 정보를 결합하고 재가공하여 사용자에게 적절한 서비스를 제공해주는 행동 네트워크[2]를 이용한 웹서비스 결합을 제안한다.

웹상의 정보를 자유롭게 이용하기 위해 기존의 웹문서를 파싱하기 보다는 다양한 정보를 제공 받을 수 있는 웹서비스를 선택하였다. HTML 문서는 문법적으로 명료하지 않고 정보의 위치(URL)가 자주 바뀌는 등 문제가 많다. 웹서비스는 XML에 기반하고 있고 WSDL(Web Service Description Language)을 통해 정보를 주고받기 때문에 보다 체계적이다. 웹서비스의 재활용성을 높이고 보다 복잡한 서비스 생성을 위해 웹서비스 결합이 많이 연구되고 있다. 웹서비스 결합 시스템은 다양한 정보를 가져와 통합, 재가공하여 사용자에게 동적으로 서비스를 만들어낸다.

매순간 사용자의 환경과 목표가 변하는 상황에서 웹서비스 결합 시스템이 적절한 정보와 서비스를 생성하기 위해서 규칙 기반 선택 방법보다 동적으로 환경과 행동 노드의 상태 값을 계산해 적절한 서비스를 택하는 행동네트워크를 사용하였다. 사용자의 행동 상황과 프로파일을 분석해 행동들을 선택함으로써 다양한 서비스를 만들어낸다.

그림 1은 웹서비스를 결합을 통한 서비스 생성을 도식화하였다. 사용자 스케줄이나 위치 정보부터 사용자에게 관련된 정보들을 모은 사용자 환경 온톨로지(Ontology)를 구성한다. 온톨로지에 기반을 둔 연속적인 서비스 노드의 동적 선택이 다

양한 서비스 생성을 가능케 한다.

행동 네트워크를 바탕으로 한 동적인 웹서비스 결합의 유용성을 보이기 위해 아마존 상품 정보를 동적으로 제공하는 시스템을 아마존 웹서비스 툴킷을 이용하여 개발하였다[3]. 데모 시스템을 통해 행동네트워크 모델이 서비스 추천에 적절한지를 실험해 보았다.



그림 1. 웹서비스 자동생성

2. 서비스 추천 방법

동적인 웹서비스 선택을 위해 여러 개의 서비스 목록을 만들어 놓고 이를 사용자 상황에 맞게 적절하게 결합을 한다. 이때 다양한 행동들의 목표를 상호 조정해주기 위한 행동 선택 방법이 필요하다. 이러한 행동 선택 방법 중의 하나인 행

동 네트워크는 행동들의 선택 순서를 추상적인 연결 관계를 이용하여 표현하는 방법이다.

2.1 웹서비스

웹서비스는 다양한 인터넷 환경 내에서 정보의 통합과 서비스의 상호 연계라는 필요성 때문에 대두되었다. 기존의 사이트에 의존한 서비스 제공에 한계를 넘어 XML을 이용해 웹 시스템을 통합하는 플랫폼을 제공하였다. 웹서비스의 특징은 다음과 같이 요약될 수 있다.

- 모듈형: 서비스 컴포넌트는 자기 스스로도 유용하며 재사용 가능하고 더 큰 컴포넌트를 만들 수 있다.
- 가용성: 서비스는 그것을 사용하려는 시스템에 이용가능하다. 서비스는 그것이 사용 가능한 시스템 또는 특정한 패러다임 외부에 노출되어야 한다.
- 서술성: 서비스는 서비스의 인터페이스를 명시하는데 사용되는 기계가 읽을 수 있는 정보를 가지고 있어야 한다.
- 구현독립성: 서비스 인터페이스는 궁극적인 구현에 독립적인 방법으로 제공되어야 한다.
- 검색성: 서비스는 사용자가 검색할 수 있는 저장소에 접근 가능한 형태로 담겨 있어야 한다.

2.2 행동 네트워크

행동 네트워크는 행동, 외부목표, 내부목표가 링크를 통해 연결되어져 있다. 그림 2는 행동 네트워크의 한 예를 보여준다. 행동은 선행조건, 추가조건, 삭제조건, 활성화도, 실행코드로 이루어져 있다. 선행조건은 행동이 실행되기 위해서 참이어야 하는 조건들의 집합이다. 삭제조건은 행동이 실행되었을 때 거짓이 되기 쉬운 조건들의 집합이다. 활성화도는 행동이 활성화된 정도를 나타낸다. 실행코드는 행동이 선택되었을 때 수행되는 내용을 담는다. 그림 3은 행동의 구성요소를 보여준다.

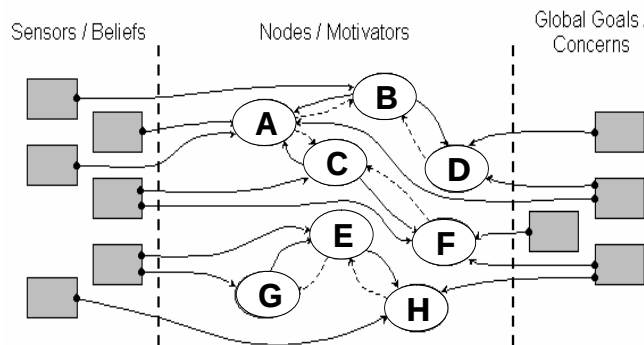


그림 2. 행동 네트워크의 예

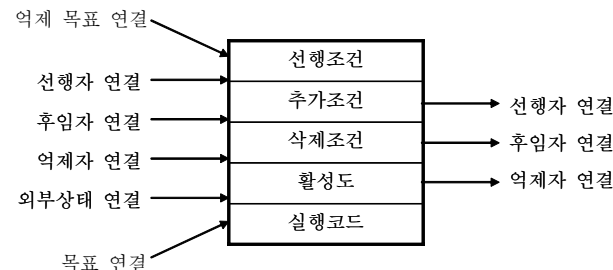


그림 3. 행동의 구성요소와 연결관계

외부상태가 행동의 선행조건일 때 외부상태연결이 설정된다. 목표에 도움이 되는 행동일 경우 목표와 행동사이에는 목표연결이 설정된다. 목표에 방해가 되는 행동일 경우 억제 목표연결이 설정된다. 행동들 사이에는 선행자연결, 후임자연결, 억제자연결이 존재한다.

2.3 행동선택 과정

그림 4는 여러 행동 중에서 현재 상황에 가장 적합한 하나의 행동을 선택하기 위한 과정을 보여준다.

```

/* 후보 중에서 하나의 서비스를 선택*/
WHILE (1) {
    initialization(); // 후보 리스트를 초기화
    spreading activation(); // 활성화도 값을 갱신
    normalization(); // 서비스의 활성화도 값을 정규화 함이  $\pi$ 가 되도록
    FOR all services {
        IF( all preconditions are true
            && activation(service) >  $\theta$  ) {
            candidate(service); // 후보 리스트에 추가 }
    }
    /* 가장 높은 활성화도 값을 가지는 후보 서비스를 하나 선택 */
    IF( candidate() == NULL ) { /* 후보 리스트에 서비스가 하나도 없다면 */
         $\theta = 0.9 * \theta$  /*  $\theta$ 를 줄임 */ }
    ELSE { elect(); break; }
}

```

그림 4. 행동 선택 알고리즘

3. 실험 및 결과

제안하는 모델의 유용성을 보이기 위해 Amazon Web Service를 이용한 가상의 구매 시나리오를 작성하였다. 한달에 평균 책을 5권정도 구입하는 20대 학생인 사용자가 3권의 책을 구매하는 동안 서비스 에이전트의 행동 선택을 기록하고 이를 분석한다.

그림 5는 웹서비스 결합 시스템을 보여주고 있다. 상단에 주소창과 사용자 피드백을 받을 수 있는 슬라이드가 있다. 가운데 창은 사용자가 웹 정보를 보는 곳이다. 하단은 웹서비스를 통한 서비스를 제공하는 곳이다. SOAP(Simple Object Access Protocol)을 통해 아마존 웹서비스와 정보를 주고받았다.



그림 5. 웹서비스 결합 시스템 인터페이스

3.1 행동 네트워크 설계

행동 네트워크는 그림 6과 같이 정의되고, 위쪽에 위치한 상자는 외부상태를 나타내며, 아래쪽에 위치한 상자는 서비스

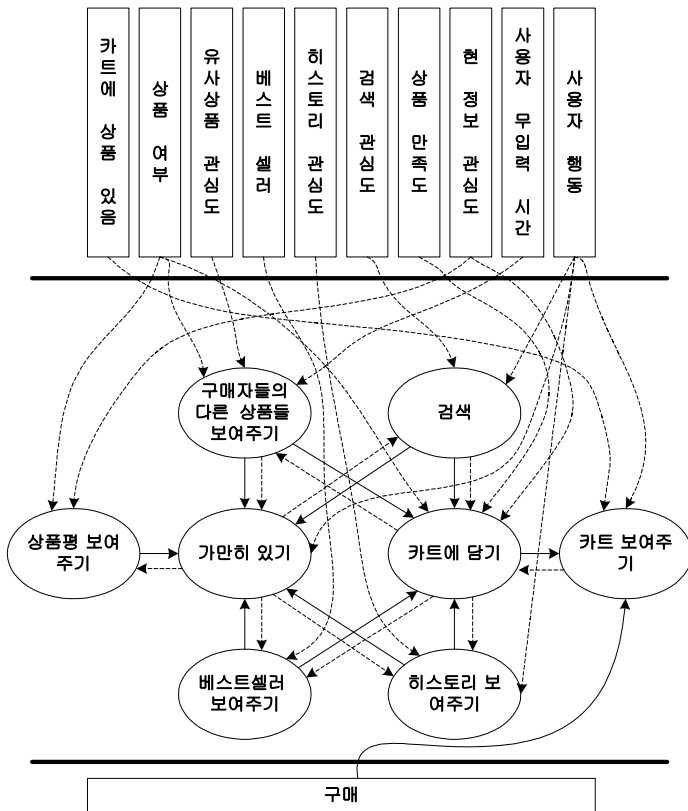


그림 6. 행동 네트워크 (실선은 후임자연결 또는 외부상태연결을 나타내고, 점선은 선행자연결 또는 목표상태연결을 나타냄)

에이전트의 내부목표를 나타낸다. 중앙의 원은 에이전트가 선택할 수 있는 기본 행동(표1)을 나타낸다. 외부상태, 내부목표, 행동이 여러 가지 연결을 통해 네트워크 형태로 엮여져서 행동 선택을 수행한다.

주어진 환경에서 얻을 수 있는 외부상태는 사용자가 행동중인지 아닌지, 사용자 무입력 시간이 김, 상품 만족도, 추천 만족도, 검색 관심도, 히스토리 관심도, 베스트셀러 관심도, 카트에 상품 있음, 현 정보 관심도 높음과 낮음 등이다. 내부목표는 사용자가 상품을 구매하도록 하는 것이다.

표 1. 서비스 목록

서비스	서비스 내용
검색	사용자가 필요한 정보를 검색할 수 있도록 검색창을 띄워줌.
히스토리 보여주기	사용자가 검색했던 상품 정보들을 요약해서 보여줌.
베스트셀러 보여주기	가장 많이 팔린 상품들을 요약해서 보여줌.
구매자들의 다른 상품들 보여주기	현재 상품을 구입한 사람들이 많이 구입한 상품 순으로 보여줌.
상품평 보여주기	현재 상품에 대해 다른 사용자나 전문가가 쓴 상품평을 보여줌.
가만히 있기	서비스 제공을 하지 않고 가만히 있음.
카트에 담기	현재 보고 있는 상품의 구입을 유도하기 위해 카트에 담을지를 묻는다.
카트 보여주기	카트에 담긴 상품을 보여줌.

3.2 결과 및 분석

그림 6은 사용자가 책 3권의 구매를 마칠 때까지의 서비스 선택 결과를 나타낸다. 서비스 에이전트는 총 134번의 서비스 선택을 했다. 10번까지는 사용자가 쇼핑을 시작하는 단계이므로 검색하기를 많이 추천했고 사용자가 상품에 대해 구체적으로 살펴보고 있는 10번에서 40번까지는 상품평 보여주기, 구매자들이 구매한 다른 상품 보여주기과 베스트셀러 보여주기를 고르게 추천했다. 사용자가 구입할 책을 결정한 시점인 40번에서 50번사이에서 카트 담기를 추천해 사용자가 책 한권을 카트에 담았고 이후 서비스 에이전트는 카트에 담긴 상품 보여주기를 실행하면서 사용자가 결제를 하도록 유도하였다. 60번이후로는 사용자가 더 이상 새로운 상품 정보보다는 기존의 검색했던 상품들 중에서 구매를 결정하려는 상황이다. 이에 서비스 에이전트는 히스토리 보여주기 행동을 많이 추천하여 사용자 구매를 유도하는 적절한 서비스 선택을 하였다. 100번 이후는 카트 담기, 카트 보여주기과 히스토리 보여주기를 적절히 추천하여 사용자의 상품 구매라는 목표에 부합하는 행동을 하였다.

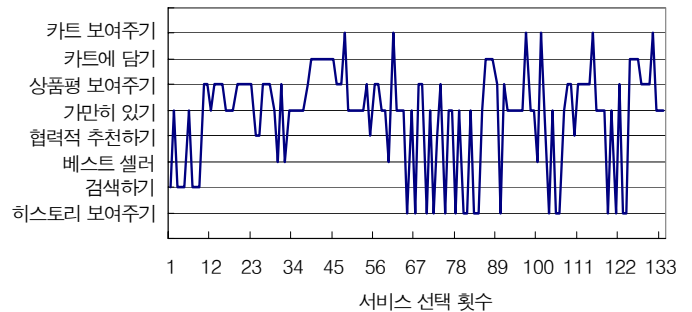


그림 7. 서비스 선택 순서

4. 결론 및 향후연구

본 논문에서는 생태학에 기반을 둔 행동 네트워크를 이용하여 웹서비스를 결합하는 시스템을 연구하였다. 행동 네트워크의 사용이 웹서비스 결합에 유용함을 보이기 위해 웹서비스 결합 시스템을 만들고 실험결과를 분석해보았다. 시스템은 사용자 환경과 웹 환경을 자동적으로 분석하고 필요한 행동 선택을 하였고 목표에 부합하는 웹서비스 결합을 보여주었다. 생태학에 기반을 둔 행동선택 방법이 기존의 규칙 기반의 행동 방식과 달리 예측이 불가능한 상황에서도 동적으로 행동 선택 기능을 수행한다.

향후연구는 UDDI(Universal Description, Discovery and Integration)를 이용해 특정 도메인에 국한되지 않은 웹서비스들을 통합하는 것과 유비쿼터스 환경에서 서비스 결합 모델을 연구하는 것이다.

감사의 글

본 연구는 프론티어 연구사업의 지원에 의한 것임.

참고문헌

- [1] J. J. Bryson, D. L. Martin, S. A. McIlraith and L. A. Stein, "Toward behavioral intelligence in the semantic web," *IEEE Computer*, vol. 35, no. 11, pp. 48-54, 2002.
- [2] P. Maes, "How to do the right thing," *Connection Science Journal*, vol. 1, no. 3, pp. 291-323, 1989.
- [3] Amazon Web Services, <http://www.amazon.com/webservices>.