FastMap을 이용한
웹 문서 시각화 시스템의 설계 및 구현

문진석*, 손기락, 김차성
한국외국어대학교 컴퓨터공학과

Design and Implementation of Web Document Visualization System using FastMap
Jinsuk Mun, Kirack Sohn, Chasung Kim
Dept. of Computer Science & Engineering, Hankuk University of Foreign Studies

요 약
인터넷의 발달과 더불어 매일매일이 증가되는 문서는 정보로부터 자신에게 필요한 정보만을 추출하는 데 많은 시간과 노력이 소요된다. 이러한 정보수집의 어려움으로 정보를 쉽게 얻는 것이 중요하며, 웹 문서 시각화 시스템은 사용자가 정확히 정보를 과거에 접해봤던 웹 문서를 다시 방문하는 경험에서 딴하게 한다. 이를 위해 인터넷 익스플로라를 통해서 방문한 웹 문서의 URL, 키워드, 문서간의 유사성을 추출하여 시각화 한다. 시각화 알고리즘으로 Fastmap을 사용하였다. 본 논문에서 Fastmap은 웹 문서간의 유사성을, 즉 상대적인 거리의 결국 형태를 2차원 공간으로 표현하는 알고리즘을 사용한다. 2차원 공간으로 매핑된 주변에 있는 웹 문서 간 거리를 확대하면 방문한 웹 문서와 유사성이 있는 문서를 쉽게 찾을 수 있다.

1. 서론

온라인에서 인터넷을 정보의 바다라고 말하며 많은 양의 정보를 얻는 것이 점점 더 수월해지고 있다. 그러나, 방대한 정보의 양과 많다 많은 정보를 습득하고 이용할 수 있는 방법을 찾고 있지만 그 안에는. 오히려, 정보의 양에 따른 정보처리량은 자신이 필요로 하는 정보를 추출하는데 많은 시간과 노력을 요구한다. 이러한 환경에서 사용자가 방문한 웹 문서는 과거에 방문한 웹 문서를 다시 방문하는 경험에서 웹 히스토리 정보를 보다 체계적, 시각적으로 분류하는 도구가 필요하다[1].

본 논문에서 제시하는 웹 문서 시각화 시스템은 사용자가 방문한 웹 히스토리를 보다 체계적으로 관리하기 위해 웹을 방문할 때마다 문서의 유사성을 추출하여 유사성이 있는 웹 문서까지 시각화(Visualization) 하였다. 시각화를 위한 알고리즘으로 Fastmap을 사용하여 웹 문서간의 유사성을 보다 2차원 공간으로 매핑한다. Fastmap은 웹 검색이나 클러스터링에서 문서간 유사성을 k-차원(k=1,2,3)으로 매핑하는 문서나 n-차원(n≥3)을 k-차원으로 감소시키는 문제에서 적용되고 있다.

본 논문의 구성은 2절에서는 관련 연구에 대한 공간, MDS(Multi-Dimensional Scaling)를 기술하고, 3절에서 시스템 구성, 웹 문서 추출, 웹 문서 분석, 데이터베이스 설계, 시각화, 사용자 인터페이스를 다루었으며, 마지막 4절에서는 결론 및 향후 연구방향을 다룬다.

2. 관련 연구

2.1 벡터 공간 모델
정보검색 분야에서 문서간의 유사한 정도를 기술하기 위해 여러 모델을 개발하였다. 그 중에서 가장 대표적인 모델인 벡터 공간 모델은 문서간의 벡터를 n차원 벡터공간으로 생각하여 벡터 사이의 cosine 값을 측정하여 문서간에 유사한 정도를 계산하는 방법이다. 이 모델은 문서간의 유사성 계산 방법에서 대부분 기존모델로 사용되었으며, 특히 SMART 시스템 을에서 사용한 모델이다. 이 시스템은 각 문서의 주제제목, 크리에이터 기법, 이원문서 기법과 같은 정보검색의 여러 분야에서 시범되었으며, 이 실험을 통해 문서내 주요어, 주요 단어의 빈도수와 문서의 주요 방향에 의해 주요한 특성 및 문서의 주요간단부를 발견하고, 그 결과를 문서 내 주요어로 가공하여 보여주었다[2].

2.2 MDS

MDS(Multi-Dimensional Scaling)는 Fastmap과 같은 문제, 즉 문서간 유사성을 k-차원으로 매핑하는 문제나 n-차원을 k-차원으로 감소시키는 문제와 관련된 다양한 분야(정, 사회학, 심리학, 시각 분석, 물리학)에서 적용되고 있다. 또한, MDS는 데이터를 두 간의 유사성으로부터 데이터 집합의 구조를 파악하는데 유용하다. 반면에 애플리케이션을 이용한 Fastmap은 비교적 두 가지 단점이 있다. 첫째, 데이터의 수많은 변량, 복잡도 O(N^3)이다. 이는 큰 데이터 집합, 즉, 활용되는 데이터베이스에 적절히 비성 용이성이 있다. 둘째, 빠른 데이터 경계가 불가능하다. 데이터 베이스에서 항목의 추가, 검색 등에 가장 적합한 수행 복잡도는 O(N^2)이나 그러나, MDS는 O(N^3)의 복잡도로 데이터를 효율적으로 검색할 수 없다[3].

본 논문에서 이용한 Fastmap은 수행 복잡도가 O(N^2)이다.
3. 웹 문서 시각화 시스템

본 논문의 웹 문서 시각화 시스템의 전체적인 시스템 구성은 크게 두 부분으로 구성되어 있다. 첫째, 웹 클라이언트의 인터넷 익스플로러와 웹 문서 시각화 시스템을 통합하여 웹 문서 추출, 웹 문서 분석 그리고 데이터베이스에 저장하는 전처리 부분이 있다. 둘째, 전처리 부분에서 얻어진 정보를 이용하여 시각화, 사용자 인터페이스 부분이다.

웹 문서 시각화 시스템의 전체적인 구조는 [그림 1]과 같다.

3.1 웹 문서 추출

사용자가 웹 문서 시각화 시스템을 쉽게 사용할 수 있도록 하기 위해서 COM을 사용하여 인터넷 익스플로러 4.0에 병합하였다. 웹 문서 추출은 인터넷 익스플로러 4.0에서 현재 보고 있는 웹 문서의 다중모드가 활성화되고, 파일이 완료될 때는 DocumentComplete 이벤트가 발생한 후 웹 문서 추출모듈을 호출한다. 웹 문서 인터페이스를 얻기 위해서 MSHTML.DLL의 IHTMLDocument2 인터페이스를 사용한다. 이 인터페이스는 평가 튜플로 구성된 객체이다. 즉 파싱된 HTML의 객체들에 구조적으로 관리하는 역할을 한다. 따라서 IHTMLDocument2 인터페이스를 얻으면 파싱된 HTML에 자바스크립트 접근할 수 있다. IHTMLDocument2 인터페이스를 얻는 방법은 웹 사이트 작동에 따라 다르다. 파싱 방법은 Visual Basic과 같은 환경에서 쓰이며, 수행 속도면에서 직접 인터페이스 함수를 호출하는 방법이 가장 효과적이다. 실행 방법은 Visual C++ 환경에서 수행속도나 메모리에 있어서의 방법보다 좋다. 본 논문에서는 수행 속도와의 혜택을 위해서 인터페이스 함수를 호출하는 방법을 사용한다.

또한 IHTMLDocument2는 매우 많은 프로파티와 메소드가 가지고 있으며 이중에 get_all() 프로파티를 사용하여 HTML의 모든 요소들의 전체 객체를 추출한다.

3.2 웹 문서 분석

웹 문서 분석은 특정 추출(Feature Selection)과 문서간의 유사성을 추출하는 작업으로 구성된다.

특정 추출은 의미의 문서에서 문서를 대표 할 수 있는 중요한 키워드를 추출하여 문서의 특징이 되는 키워드 벡터를 생성하는 방법이다. 문서간의 유사성은 문서간의 키워드 벡터 사이의 코사인값을 측정하여 문서간에 유사한 정도로 계산하는 방법이다.

3.2.1 특정 추출

먼저 문서에 나오는 모든 문단들을 단어 또는 토כי 열로 변환하는 어휘 분석 과정을 수행한다. 그 결과 문서는 키워드와 슬라이드로 이루어진 벡터로 표현되며, 이렇게 표현된 벡터는 특정 추출을 거쳐, 문서의 특정이 되는 중요한 키워드를 표현하는 벡터만을 추출한다. 그리고 키워드의 벡터들에 따라 가장치 벡터를 적용한다.

본 시스템에서 사용하는 특정 추출하는 과정은 크게 두가지 과정을 거친다. 첫째, 각 문서의 키워드 중에는 a, the, and 등 (430여 단어) 정도 발생 반도가 높지만 문서의 특성에서는 가치가 없는 용어와를 제거한다. 둘째, 스테이지를 통한 방법으로 각 키워드의 어휘론적인 변형을 적용한 방법을 적용하는 것이다. 위의 두가지 방법을 이용하면 키워드 벡터의 크기를 줄여, 셰익에 요구되는 메모리 용량을 줄일 수 있고, 전처리 효율을 높일 수 있다[2][3][6].

3.2.2 웹 문서 유사성

본 논문에서는 문서사이의 유사도 측정방법으로서 다음과 같은 표[식 1]에 표현된 코사인 측정법을 사용한다. 코사인 측정법은 비교 대상이 벡터화가 되었을 경우 계산량을 줄일 수 있는 방법이 있다.

\[
similarity(d_1,d_2) = \frac{d_1 \cdot d_2}{\|d_1\| \cdot \|d_2\|}
\]

노트는 두 벡터의 내적이고, \(\|\cdot\|\)는 벡터의 Euclidean norm이다[3].

3.3 데이터베이스 설계

[그림 3]은 웹 문서 시각화 시스템의 시각화를 위해 설계한 데이터베이스의 ER 다이어그램이다. 이는 시각화(Visualization)를 위한 기초 자료를 가린다. 데이터베이스는 문서 데이터, 키워드 데이터, Cosine Similarity 데이터, Frequency 데이터로 구성되어 있으며 문서 데이터는 문서번호, 문서의 헤더를 포함하는 URL, 문서의 키워드, 사용자에게 문서의 중요성을 나타내는 웹 문서의 방문 횟수, 2차원 좌표 정보로 구성된다. 좌표 점은 문서간의
유사성을 FastMap 알고리즘에 적용하여 추출한 2차원 좌표를 가지고 새로운 문서로 부가할 때마다 구한다. 이를 데이터베이스 에 저장하여 웹 문서 시각화 시스템의 수행 시간을 단축한다.

사용자 인터페이스(User Interface)는 사용자의 상호작용으로 생성된 웹 문서에 대한 정보를 보여준다.

[그림 4] 웹 문서간의 유사성을 2차원 공간으로 배경화 사용자 인터페이스

4. 결론 및 향후 연구방향

본 논문은 웹 문서 기반 흐름정보를 효과적으로 관리하여 관계에 방문한 웹 문서를 쉽게 찾을 수 있는 웹 문서 시각화 시스템을 구현하였다. 웹 문서 시각화 시스템은 사용자를 대상으로 웹 문서를 분석하고, 사용자가 현재 방문 중인 웹 문서가 2차원 공간 어느 부분에 위치인지 확인할 수 있다. 이로 인해 사용자는 흐름정보에 의한 웹 문서의 자료를 얻기 위해 URL을 입력하여 검색하는 동일 시간과 노력을 줄일 수 있게 된다.

향후 과제로서 시각화 구현에 있어 2차원 공간으로 배정하면서 생성되는 오차, 즉 stress를 최소화시키는 방법의 연구와 사용자가 실용적으로 사용할 수 있는 사용자 인터페이스 구현에 대한 연구가 필요하다.

참고문헌


