사용자 편의를 위한 북 마크 에이전트

강상구[0], 정현섭, 최중민
한양대학교 컴퓨터공학과
{sgkang, hsjung, jmchoi}@cse.hanyang.ac.kr

A User-oriented Bookmark Agent

Sanggu Kang[0], Hyunsup Jung, Joongmin Choi
Dept. of Computer Science and Engineering, Hanyang University

요 약

본 논문에서는 사용자가 관심 있는 문서를 카테고리별로 직접 분류해서 추가하던 작업을 자동으로 분류하고 추가 할 수 있는 북 마크 에이전트를 제안한다. 북 마크 에이전트는 사용자가 브라우징 시 사용자 성향을 분석하여 관심 있는 문서를 얻을 수 있다. 문서 내에서 특징을 찾기 위해 TF·IDF를 사용 하였으며 또한 단어의 가중치 부여와 유사도를 계산하기 위해 벡터 공간 모델을 사용 하였다. 이 작업을 통해 부정적인 문서의 URL이 추가될 수 있으며 이러한 문제를 해결하기 위해서 사용자의 피드백을 이용하여 제거할 수 있도록 하였다.

1. 서론

최근 들어, 인터넷의 발전으로 엄청나게 늘어나고 있는 정보의 양은 사용자들에게 많은 지식과 다양한 서비스를 제공하고 있다. 사용자는 검색엔진이나 브라우징을 통해 관심 있는 정보를 수집하며 이 정보를 재사용하기 위해서 북 마크에 카테고리(category)별로 분류(classification)하고 있다. 여기서 항상 사용자가 수동으로 정보를 수집하고 분류해야 한다는 불편한 점이 있으며 이 문제를 해결하기 위해 북 마크 에이전트를 제안한다[1, 2].

북 마크 에이전트는 사용자의 브라우징 행위를 통해 관심 있는 문서를 얻을 수 있고 그 문서의 URL을 카테고리별로 분류해서 자동으로 추가한다. 단어의 가중치(weight) 부여와 수집한 문서의 특징(feature) 벡터를 계산하기 위해서 정보 검색에서 이용하는 벡터 공간 모델(vector space model)을 사용 하였으며 복잡한 계산량을 줄이기 위해 상위 랭크 된 10개와 20개의 특징을 이용하여 각각의 문서를 비교하였다[3, 4, 5].

본 논문의 구성은 다음과 같다. 2장에서는 관련연구를 설명한다. 3장에서는 북 마크 에이전트의 전체적인 시스템 구성과 각 모듈에 대한 기능을 설명한다. 4장에서는 10개의 특징과 20개의 특징을 사용하여 관련 있는 URL을 얼마만큼 정확하게 분류하는 지 실험 결과를 기술한다. 마지막으로 5장에서는 연구 내용을 요약하고 앞으로의 향후 과제와 함께 결론을 내린다.

2. 관련연구

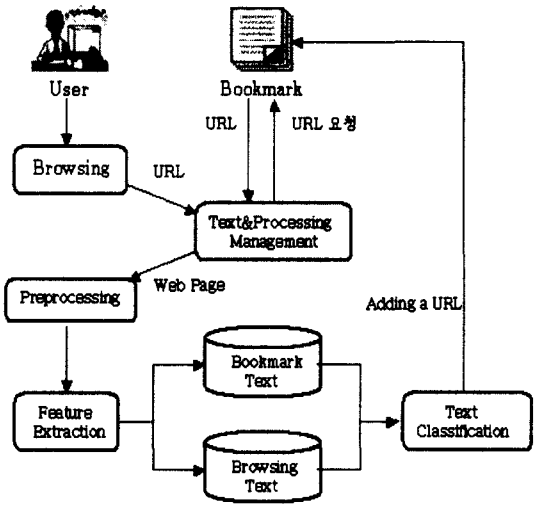
기존의 웹 개인화(web personalization)는 크게 다음과 같은 두 가지의 형태로 분리할 수 있다. 첫 번째, 수동 결정 규칙(manual decision rule)[6]은 시스템 관리자가 사용자의 개인 신상 정보나 세션 히스토리를 이용하여 직접 개인화 시키는 형태이다. 이 경우에 생성되는 규칙은 사용자 개인을 위한 것이다. 두 번째, 협동 필터링(collaborative filtering)[7]은 사용자의 평가나 성향을 직접 입력 받거나 상관관계를 판단하는 엔진을 통해서 사용자의 성향을 예측하는 경우이다.

웹 디렉토리의 가장 방대한 예는 Yahoo이며 분류된 페이지들만 해도 100만에 가깝다. 오늘날의 웹 정보는 계속해서 증가하므로 문서를 분류하기가 어렵다. 따라서 야후를 비롯한 대부분의 시스템들은 사람이 직접 수동으로 문서를 분류 하고 있다. 여기서 사용자마다 카테고리의 다른 의미를 가지는 것은 문제가 되고 있다.

WebGlimpse는 브라우징과 탐색을 결합한 도구이다. 이것은 작은 탐색 상자를 모든 HTML 페이지의 하단에 두어 브라우징을 중단하지 않고 해당 페이지의 이웃 페이들이나 전체 사이트에서 탐색이 가능하도록 한다. 이는 이웃 탐색을 통해 끊임없이 구축된 하이퍼링크들을 따라가는 것과 마찬가지이다. 따라서 개인 웹 페이지와 자주 찾는 URL 목록에 대한 색인을 구축하는 데 유용하다[8, 9].

3. 시스템 구현

북 마크 에이전트 구조는 그림 1과 같은 구조를 가지며 전체적인 시스템 구조 설명은 다음과 같다. 사용자 브라우징을 통해 성향을 분석하여 관심 있는 문서를 얻은 후, 문서가 북 마크에 없으면 북 마크에 분류된 문서와 브라우징을 통해 얻은 문서를 전처리 단계와 특징 추출 단계를 거쳐서 각각 저장한다. 두 개의 저장된 문서의 유사도(similarity)를 비교 하기 위해 벡터 공간 모델을 사용하고 있다.



[그림 1] 시스템 구조

3.1 사용자 브라우징

사용자는 브라우징을 통해 긍정적인(positive) 문서와 부정적인(negative) 문서를 구분하여 얻을 수 있지만 북 마크 에이전트는 사용자가 브라우징 하는 모든 문서를 처리해야 하는 문제점을 가지고 있다. 이 문제점을 해결하기 위해 에이전트는 사용자가 브라우징 하는 동안 개인의 웹 성향을 분석해서 긍정적인 문서를 추론하여 얻을 수 있다.

3.2 전처리 단계

북 마크는 여러 개의 카테고리로 구성이 되어 있으며, 각 카테고리 안에는 여러 개의 URL이 있고 이들 URL의 문서를 각각의 문서로 간주한다. 먼저 카테고리화 되어 있는 문서에서 HTML 태그와 불용어(stopword)를 제거한다. 그리고 브라우징을 통해 가져온 웹 문서에 대해서도 동일한 처리를 한다.

3.3 특징 추출 단계

본 논문에서는 분류된 문서들과 브라우징을 통해 얻은 문서 사이의 특징을 추출하기 위해 정보 검색에서 사용되는 단어의 빈도수와 역 빈도수(TF·IDF:Term Frequency·Inverse Document Frequency)에 기반해서 웹 문서를 특징화 한다.

북 마크의 분류된 문서들의 특징을 추출하기 위해 *tf·idf*를 사용하였으며 브라우징 시 얻은 문서에서는 *tf*만을 사용한다. w_{ij}는 북 마크에 카테고리별로 분류되어 있는 각 카테고리의 가중치이다.

$$w_{ij} = f_{ij} \times idf_i$$

f_{ij}는 분류된 문서 d_j에 있어서 나타나는 단어 t_i의 빈도수이고, idf_i는 분류된 문서 집합에 단어 t_i가 나타나는 문서의 수이다. 일반적으로 사용된 역 빈도수의 척도는 다음과 같이 주어진다.

$$idf_i = \log \frac{N}{n_i} + 1$$

여기서, N은 북 마크의 분류된 문서의 총수이고, n_i는 단어 i가 나타나는 문서의 수이다.

3.4 문서 분류 단계

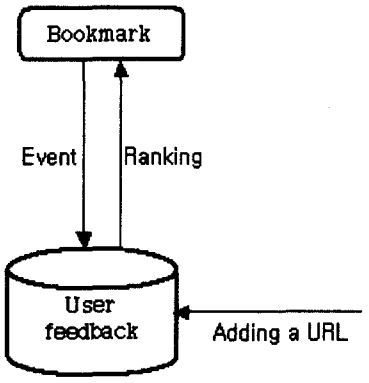
본 논문에서는 분류된 문서와 브라우징을 통해 얻은 문서 사이의 유사도 계산과 순위(rank)를 구하기 위해 벡터 공간 모델을 사용한다. 그리고 이 모델을 사용하므로 생기는 복잡한 계산량을 줄이기 위해 상위에 랭크 된 특징만을 고려한다.

$$sim(d_i, b) = \frac{\sum_{i=1}^{t} w_{ij} \times w_{ib}}{\sqrt{\sum_{i=1}^{t} w_{ij}^2} \times \sqrt{\sum_{i=1}^{t} w_{ib}^2}}$$

여기서, w_{ib}는 브라우징을 통해 얻은 문서의 가중치이다.

3.5 사용자 피드백(feedback)

사용자가 명시적으로 피드백을 주는 것이 아니고, 시스템이 사용자의 행동을 관찰, 모니터링하여 관심 있는 정보나 습성을 파악하는 것으로, 사용자의 부담 없이 정보가 입수가 가능하다는 것이 장점이다.



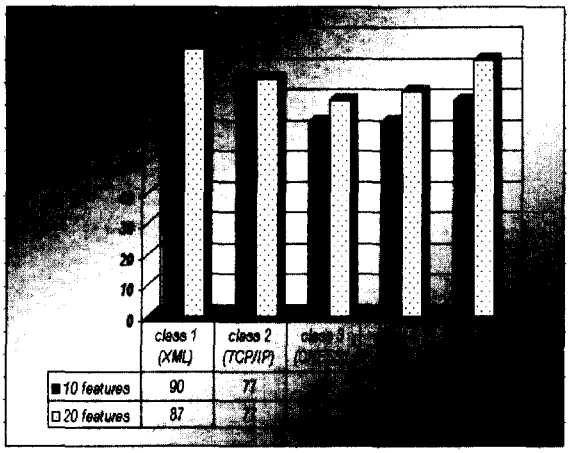
[그림 2] 사용자 피드백(feedback)구조

본 논문에서는 사용자가 분류된 문서들의 URL을 사용하면 그 이벤트에 의해 분류된 문서의 URL은 카운트 되며 그림 2처럼 사용자 피드백으로 모든 카테고리들의 내용은 보내어지고, 증가된 횟수에 의해 높은 수로부터 순위를 주어 다시 북 마크로 보내어진다. 그리고 에이전트에 의해 자동으로 추가 할 때는 URL을 저장소로 가져와 관련 있는 카테고리에 추가를 한다. 이때 그 카테고리속에 너무 많아지는 URL을 막기 위해 한정된 수로 제한하며, 피드백에 의해 마지막으로 랭킹 된 URL을 삭제하고 거기에 저장한다. 이렇게 함으로써 에이전트에 의해 부정적인 URL이 추가 된 것을 사용자의 피드백에 의해 제거 할 수 있다.

4. 실험 및 분석

실험은 수동으로 문서를 분류하는 야후 검색 엔진에서 5개의 카테고리를 선정하여 사용하였으며 각 카테고리 속에는 관련된 5개의 URL을 먼저 추가 하였다. 표 1에서 카테고리를 클래스(class)로 분류하고, 상위 랭크 된 10개와 20개의 특징을 이용하여 브라우징 시 얻은 문서가 각 클래스에 정확하게 분류 되는지 비교 하였다.

[표 1] 특징 10개와 20개 분류 비교



실험 문서는 각 클래스 당 30개의 문서로 실험 하였으며 표 1의 정확도(accuracy)는 각 클래스에 해당하는 전체 문서 중에서 몇 개의 문서가 정확하게 분류되는 지에 대한 비율 값이다. 클래스 1에서는 10개의 특징을 이용할 때 더 높은 정확도를 가지는 데 이는 분류에 크게 영향을 주는 특징이 문서마다 3-4개 정도 나타나기 때문이다. 그리고 분류에 영향을 주지 않는 특징 수가 많아지면 정확도는 낮아진다. 반면 클래스 3에서 클래스 5번까지는 분류에 영향을 주는 특징이 문서마다 하나 정도이기 때문에 크게 영향을 미치지는 못한다. 그러므로 많은 특징을 가진 20개의 특징을 이용하면 더 정확한 값을 얻을 수 있다. 특별(specific)한 클래스와 일반적(general)인 클래스에 따라 특징 수를 다르게 두는 것이 시간적인 면과 정확도 면에서 더 효과적이라는 것을 실험을 통해 알 수 있다.

5. 결론 및 향후과제

본 논문에서는 사용자 브라우징 시 성향을 분석해 관심 있는 웹 문서를 가져와 감독 학습 방식에 의해 분류된 문서들 사이에서 단어들 가중치와 이들 사이의 벡터를 계산하여 높은 값을 가지는 문서가 더욱 관련이 있다고 고려를 하고 랭킹을 주어서 URL을 추가 하도록 하였다. 계속해서 추가 되는 URL이 정말로 사용자와 관련이 있는지, 없는지를 체크해서 관련 있는 URL은 상위로 위치하고 그렇지 않은 URL은 하위로 위치를 한다. 다시 말하면, 북 마크에 추가된 URL을 사용자가 자주 사용하면 긍정적인 값을 가지고, 부정적인 값을 가지면 다음에 추가될 URL에 의해서 삭제가 된다. 이렇게 함으로 써 늘어나는 URL을 막을 수 있다.

향후 연구 과제로는 어떤 사이트에 대해서 그 사이트의 첫 페이지를 가지고 올 것인지, 아니면 관심 있는 페이지만 가져올 것인지, 아니면 그 사이트 상위 페이지부터 하위 페이지까지 모두 다 가져올 것인지에 대한 연구가 필요하다.

참고 문헌

[1] Hao Chen, Susan Dumais, Bringing Order to the Web: Automatically Categorizing Search Results, In *Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems*, pp. 145-152, 2000.

[2] Susan Dumais, Hao Chen, Hierarchical Classification of Web Content, In *Proceedings of SIGIR -00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pp. 256-263, 2000.

[3] H. Drucker, D. Wu, and V. Vapnik, Support Vector Machines for Spam Categorization, *IEEE transactions on Neural Networks*, pp. 1048-1055, 1999.

[4] 김태훈, 최중민, 사용자 편의의 인터넷 정보검색을 위한 지능형 웹 브라우징 에이전트, *정보과학회 논문지 (B)*, 25권7호, pp. 1064-1078, 1998.

[5] Xiaobin Fu, Jay Budzik, Kristian J. Hammond Mining Navigation History for Recommendation, In *Proceedings of Intelligent User Interfaces 2000*, pp. 106-112, 2000.

[6] Broadvision, http://www.broadvision.com.

[7] Net Perceptions, http:// www.netperceptions.com.

[8] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, *Modern Information Retrieval*, ACM Press, 1999.

[9] U. Manber, M. Smith, and B. Gopal, WebGlimpse: Combining Browsing and Searching, In *Proceedings of USENIX Technical Conference*, pp. 195-206, 1997.