Journal of Digital Contents Society Vol. 11 No. 1 Mar. 2010(pp. 105-115)

폭소노미 기반 개인화 웹 검색 시스템

김동욱*, 강수용**, 김한준***, 이병정****

요 약

검색엔진들은 사용자로부터 질의어를 전송받아 질의어와 관련이 가장 높은 웹 문서들을 보여주게 된다. 하지만 검색엔진이 사용자의 질의어만 가지고 사용자의 의도를 파악하여 정확한 웹 문서를 제공하기는 어렵다. 따라서 검식 엔진 시스템은 다양한 개인화 방법을 사용하여 각 사용자가 원하는 검색 결과를 보여주기 위해 노력한다. 본 논문에서는 개인화 검색을 위해 '폭소노미'를 기반으로 사용자에게 적합한 질의어를 추천해 주는 방법을 제안한다. 또한 이러한 개인화된 검색 결과를 제공하는 시스템이 가질 수 있는 프라이버시 침해 위험성을 제거하면서도 검색 서비스 제공자 입장에서는 사용자 정보를 활용한 다양한 서비스(개인화 광고등) 제공이 가능하도록 하는 개인화 검색 서비스 구조를 제안한다.

Folksonomy-based Personalized Web Search System

Dongwook Kim*, Sooyong Kang**, Hanjoon Kim***, Byungjeong Lee****

Abstract

Search engines provide web documents that are related to user's query. However, using only the query terms that user provided, it is hard for search engines to know user's exact intention and provide the very matching web documents. To remedy this problem, search systems are needed to exploit personalized search technologies. In this paper, we propose not only a novel personalized query recommendation scheme based on folksonomy but also a new personalized search service architecture which reduces the risk of privacy violation while enabling search service providers to provide other various personalized services such as personalized advertisement.

Keywords : search engine, personalization, information retrieval, folksonomy

1. 서론

현재 우리는 다양한 검색엔진(e.g. Google, Yahoo, Bing)에 질의어를 전송하여 수많은 웹 문서 중에서 우리가 원하는 문서를 검색하게 된다. 이러한 검색엔진을 사용하는 사용자들은 원하는 정보가 포함된 문서를 얻기 위해서 키워드를 선택하게 된다. 이 때, 사용자들에 의해 선택되는 키워드들은 대부분 짧고 불분명한 의미가 많다[2, 6]. 또한 검색엔진 시스템은 불분명한 의미의 질의어에 대해 정확히 사용자가 원하는 정보가 포함된 문서만을 선택하여 보여주기 어렵다. 예를 들어, 생물학에 관심이 많은 사용자가 생물학적 바이러스의 의미를 가진 문서를 검색하기 위해 "Virus" 키워드를 질의어로 전송할 때, 검색 엔진 시스템은 해당 질의어가 생물학적 바이러스인지, 컴퓨터 바이러스인지 짧은 질의어만을 가지고 정확히 알기가 어렵다. 이러한 문제는 사용자가 직접 연관된 키워드를 추가적으로 선택하고 처음 선정했던 키워드를 포함하여 질의어를 재전송하는 방법과 검색엔진 시스템에서 자동으로 사용자 프로파일의 정보를 바탕으로 개인화된 검색 결과를 보여주는 방법으로 해결할 수

※ 제일저자(First Author) : 김동욱
접수일:2010년 03월 10일, 완료일:2010년 03월 31일
* 한양대학교 전자컴퓨터통신공학과
eliudkim@hanyang.ac.kr
** 한양대학교 컴퓨터공학부 (교신저자)
*** 서울시립대학교 전자전기컴퓨터공학부
**** 서울시립대학교 컴퓨터과학부
▣ 본 연구는 서울시 산학연협력사업 (과제번호: NT080624, 연구과제명: 집단지성 기반 사용자 인식 웹검색 시스템 개발)의 지원에 의해 수행되었음.

있다.

사용자가 수동적으로 질의어를 재전송하는 경우는 사용자의 의도를 정확히 반영할 수 있지만, 여전히 키워드 선택의 어려움이 존재하고 다시 전송해야 하는 불편함이 있다. 이러한 단점을 극복하기 위해 검색엔진 시스템은 사용자 질의어 기록(log)을 바탕으로 연관된 질의어를 추천해 준다[11, 12]. 연관 질의어 추천은 사용자가 원하는 정보를 쉽고 편리하게 찾는데 도움을 주지만, 해당 질의어의 모호성을 구분해서 질의어를 추천하지는 못한다. 시스템적으로 자동화되어 개인화된 결과를 보여주는 방법으로는 검색결과를 사용자 개개인에 맞춰 재순위화(Re-ranking)하여 보여주는 방법과 처음 입력한 질의어를 사용자의 프로파일에 맞추어 확장하는 방법이 있다. 두 방법 모두 한 번의 키워드 전송으로 사용자가 정확히 원하는 결과를 얻을 수 있어 정확도와 편리성을 증대시킬 수 있지만, 사용자에 맞춰 개인화된 결과가 오히려 사용자에게 혼란을 주게 될 수도 있다[1]. 예를 들어, 시스템은 특정 사용자의 선호도 정보가 생물학적 바이러스라서 “생물학적 바이러스”에 관한 문서의 결과를 상위에 보여주거나 생물학적 바이러스와 연관된 질의어로 확장을 했지만 사용자는 컴퓨터의 바이러스 감염으로 인해 “컴퓨터 바이러스”와 관련된 문서를 찾으려고 검색한 경우에는 오히려 혼란을 주게 된다.

이러한 개인화된 검색 결과를 보여주기 위해서는 검색엔진을 이용하는 각각의 사용자 개인정보가 서버에 저장되어야 한다. 하지만, 이러한 시스템은 개인의 프라이버시가 침해될 수 있는 위험성이 증가되게 된다[3]. 따라서 이러한 문제를 해결하기 위해 사용자 프로파일 정보를 클라이언트에 저장하여 제공하는 모델이 많이 제시되어 있다[4,10]. 이러한 시스템은 사용자 프로파일 정보가 클라이언트에서 저장되어 질의의 확장이나 재순위화(re-ranking)에 이용되기 때문에 프라이버시 침해의 가능성은 낮아진다. 하지만 검색엔진을 서비스하는 입장에서는 다른 개인화 서비스(예: 개인화 광고 등)를 위해 해당 정보를 활용할 수가 없게 된다.

본 논문은 개인화된 검색이 가능하고 프라이버시 문제를 해결하며, 검색엔진 서비스 업체에서는 사용자의 프로파일 정보를 활용 가능한 효율적인 시스템 아키텍처를 제안하며, 개인화된 검색을 위해 폭소노미 서비스를 제공하는 딜리셔스 사이트[13]의 데이터를 활용하여 질의어를 추천해주는 시스템을 구현하였다. 우리가 제안하는 시스템 아키텍처는 개인화된 검색을 위하여 재순위화나 질의어의 확장에도 적용할 수 있을 뿐 아니라 개인화된 태그를 추천해 주는 시스템에도 적용될 수 있을 것이다.

본 논문은 다음과 같이 구성된다. 2장에서는 관련 시스템 아키텍처와 배경 지식들에 대해 간략히 설명하며, 3장에서는 태그 정보를 이용하여 사용자의 프로파일을 생성하는 방법을 제시한다. 4장에서는 시스템 아키텍처를 제시하며, 마지막으로 5장에서 본 논문의 결론과 추후연구 방향을 제시한다.

2. 관련연구

2.1. 벡터 공간 모델

벡터 공간 모델은 텍스트 문서를 색인 단어(index term)들의 벡터로 나타내는 대수적(algebraic) 모델이다[8]. 벡터 공간 모델을 잘 활용하기 위하여 다양한 방법론들의 활용과 실제 적용 결과에 대한 연구들이 진행되어 왔으며[7,8,15], 다양한 방법들 중 본 연구에서도 활용한 방법은 다음과 같다.

두 문서간의 유사도를 판단하기 위한 방법으로 두 문서(d_1, d_2)내 색인 단어들의 벡터($\overrightarrow{d_1}$, $\overrightarrow{d_2}$) 사이의 코사인 각도를 이용하며 공식은 다음과 같다.

$$\mathrm{sim}(d_1, d_2) = \frac{\overrightarrow{d_1} \bullet \overrightarrow{d_2}}{|\overrightarrow{d_1}| \times |\overrightarrow{d_2}|} = \frac{\sum_{i=1}^{T} w_{i,d_1} \times w_{i,d_2}}{\sqrt{\sum_{i=1}^{T} w_{i,d_1}^2} \times \sqrt{\sum_{i=1}^{T} w_{i,d_2}^2}}$$

$\overrightarrow{d_1}$, $\overrightarrow{d_2}$: 두 문서의 색인 단어 벡터
$w_{i,d}$: 색인 단어 i의 가중치
T : 두 문서에 포함된 모든 색인 단어 수

여기서 가중치 $w_{i,d}$는 간단히 문서에 색인 단어가 포함되었는지의 유무로도 판단할 수 있지만, 좀 더 정확한 유사도 판단을 위해 문서에 포함된 어떤 단어가 문서 내에서 얼마나 중요한 정도를 나타내는 TF-IDF 가중치를 사용하였다.

$$w_{i,d} = tf_i * \log\left(\frac{N}{|\{i \in d\}|}\right)$$

tf_i : 문서 d에서 단어 i의 출현 빈도수
N : 문서 집합내의 모든 문서의 개수
$\{i \in d\}$: 단어 i가 포함된 문서의 개수

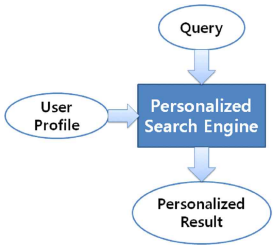
여기서 tf_i는 TF(Term Frequency)에 해당하며, 해당 문서에서 특정 단어의 빈도수를 뜻한다. TF의 의미는 해당 문서에 많이 나온 단어가 문서내에서 중요하다는 것을 나타낸다. $\log\left(\frac{N}{|\{i \in d\}|}\right)$는 IDF(Inverse Document Frequency)에 해당하며, 문서 집합내에서 해당 단어가 포함된 문서의 빈도수의 역수를 뜻한다. IDF의 의미는 문서와 연관된 단어일수록 많은 문서들에서 적게 사용되는 것을 나타낸다.

2.2. 폭소노미 (Folksonomy)

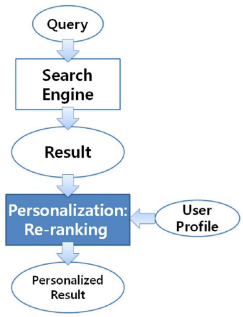
폭소노미란 전통적인 분류 기준인 디렉토리 대신 태그에 따라 나누는 새로운 분류 체계로서 "사람들에 의한 분류법"이란 의미이다. 폭소노미가 기존의 분류체계와 다른 점은 구성원들이 자발적으로 개별정보에 의미를 부여함으로써 단위정보를 체계화한다는 것이다. 이러한 폭소노미 서비스를 제공하는 가장 유명한 사이트로는 딜리셔스와 플리커[14] 사이트가 있다. 사이트의 사용자들은 자발적으로 자신이 관심을 가지는 URL이나 이미지 파일을 북마킹 혹은 저장을 할 때, 웹문서나 이미지와 연관된 태그를 작성함으로써 웹문서나 이미지 파일에 의미를 부여하게 된다. 이렇게 다수의 사용자에 의해 작성된 태그는 웹문서나 이미지들의 분류 기준으로 삼을 수 있다. 이러한 폭소노미의 기준이 되는 태그 데이터를 개인화 검색에 활용한 다양한 연구들이 진행되고 있다[6, 9].

2.3. 개인화 검색

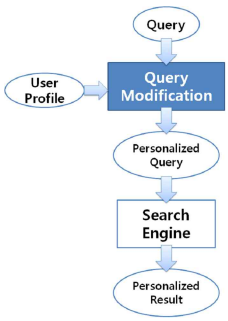
사용자가 방문하기 위하여 선택하는 페이지들과 질의어를 전송하는 것과 같은 사용자의 행동 정보를 통하여 사용자의 특징을 파악하는 것을 사용자 모델링 (혹은 프로파일링) 기술이라고 하며, 이로 인해 얻어지는 정보를 사용자 프로파일 혹은 사용자 모델이라고 한다. 그리고 검색 엔진 시스템에서 사용자 프로파일을 이용하여 검색 결과에 영향을 미치는 시스템 구성요소를 사용자 모델 컴포넌트 (User Model Component)라고 하며, 그림 1과 같이 세 가지 다른 측면에 영향을 끼친다[5].



(a) Part of Retrieval Process



(b) Re-ranking



(c) Query Modification

(그림 1) 사용자 모델 컴포넌트가 검색 시스템에 끼치는 영향

그림 1-(a)는 사용자가 전송하는 질의어에 대하여 사용자 프로파일을 활용하여 각각의 사용자에게 적합한 개인화된 검색 결과를 보여준다. 그림 1-(b)와 그림 1-(c) 보다는 빠른 응답성을 보이지만, 비개인화 정보 검색(Non-Personalized IR) 테크닉과 비교하면 오랜 시간을 소모하게 된다. 그림 1-(b)는 사용자의 프로파일에 따라 문서의 추천 순위를 재조정 하는 것으로, 사용자

에 맞추어 검색결과의 정확도(Precision)를 증가시킬 수 있게 된다. 대부분의 재순위화 시스템들은 프라이버시 문제 해결과 재순위화의 시간 소모를 줄이기 위하여 클라이언트 측에서 구현되었다. 그림 1-(c)는 사용자의 프로파일을 통해 질의어를 수정하는 시스템으로 개인화하기 위해 여러 문서를 다운로드하는 추가적인 오버헤드를 줄이기 위하여 대부분 클라이언트에서 질의어를 수정하여 전송하게 된다. 개인화를 위한 대부분의 시스템들은 각각의 사용자에 맞는 결과를 보여주기 위해 추가적인 오버헤드가 필요하게 되며 이러한 오버헤드를 줄이고, 사생활 침해를 막기 위해 클라이언트 측에서 구현되어 왔다. 하지만 이 시스템들은 검색 엔진을 서비스하는 업체가 사용자의 프로파일 정보를 이용한 다양한 개인화 서비스를 하지 못하는 단점을 가지게 된다.

우리는 본 논문을 통해 폭소노미를 이용하여 사용자 프로파일을 생성하는 방법과 사용자 프로파일을 통한 개인화 검색 시스템에서 발생하는 추가적인 오버헤드와 프라이버시의 위협을 줄이며 서비스 업체에서도 사용자의 프로파일 정보를 활용할 수 있는 시스템 아키텍처를 제안하고자 한다.

3. 사용자 프로파일 및 개인화 검색

폭소노미를 분석하여 사용자 프로파일을 생성할 수 있다면, 개인화 검색 시스템에서 사용자 모델 컴포넌트가 사용자 프로파일을 활용하여 개인화된 검색 결과를 보여 줄 수 있게 된다. 따라서 우리는 이 장을 통해 개인화 검색에 활용하기 위한 사용자 프로파일 생성 방법과 사용자 프로파일을 이용한 개인화 검색 방법을 제안한다.

3.1. 폭소노미 분석

다수의 사용자에 의해 만들어진 딜리셔스 사이트는 URL의 웹 문서 내용과 연관된 태그들을 분류할 수 있는 정보를 제공한다. 이러한 데이터를 사용자 프로파일로 활용하기 위해 다음과 같은 가정을 하였다.

> 사용자는 관심이 있는 분야와 관련된 웹 페이지들에 북마킹을 하며, 사용자가 북마킹한 URL의 태그들은 사용자가 관심 있는 분야의 단어들이다.

또한 사용자의 북마킹 데이터는 네 개의 원소를 가지고 있으며, 자세한 내용은 다음과 같다.

> < Date, User, URL, Tag >
>
> Date : 해당 URL을 북마킹한 시간
> User : 해당 URL을 북마킹한 사용자 ID
> URL : 웹 문서의 URL 주소
> Tag : 사용자가 북마킹할 때, 해당 URL의 문서를 분류하는 기준으로 삼은 단어

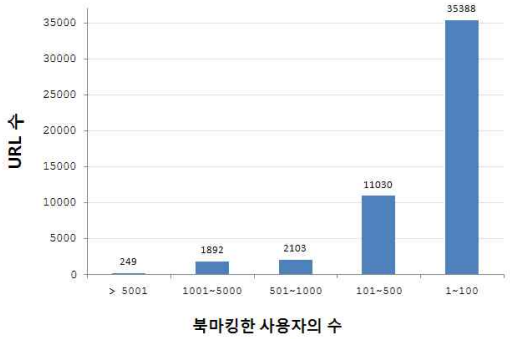
이러한 북마킹 데이터를 통해 많은 사용자들이 URL의 문서를 분류하는 기준으로 선택한 태그는 해당 URL과 연관성이 높으며, 이러한 URL과 태그의 연관성을 표현하기 위해 URL과 태그의 연관된 정도를 나타내는 값을 계산하여 m x n 크기의 행렬 M을 생성하였다. 여기서 m은 모든 태그의 수이며, n은 모든 URL의 수이다. URL과 태그의 연관된 정도를 나타내는 연관도 값(M_{ij})은 간단하게 URL j에 대해 태그 i로 북마킹한 사용자들의 수로도 표현할 수 있지만, 본 연구에서도 2.1절에서 살펴보았던 것처럼 URL과 태그의 연관도 값의 정확성을 높이기 위해 TF-IDF (Term Frequency - Inverse Document Frequency) 가중치[7,8,15]를 활용하여 다음과 같이 계산하였다.

$$M_{ij} = U_{ij} * \log(n / |URL(t_i)|)$$

U_{ij} : URL j를 태그 i로 북마킹한 사용자 수
n: 전체 URL 수
$URL(t_i)$: 태그 i로 북마킹된 URL 수

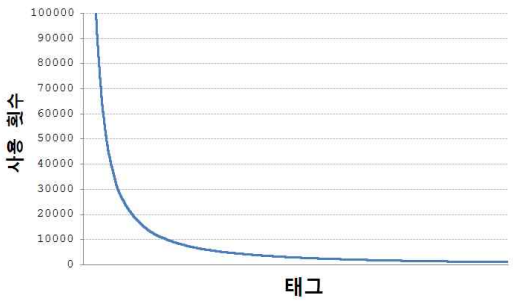
U_{ij} 는 TF-IDF에서의 TF(해당 문서에서 특정 단어의 빈도수)에 해당한다. 예를 들어, www.facebook.com 사이트에 대해 facebook으로 10619명이, social이란 태그로 10083명의 사용자가 북마킹을 하였다면, TF값은 각각 10619,

10083이 될 것이다. 하지만 이러한 값은 그림 2에서 볼 수 있는 실제 딜리셔스 사이트의 북마킹 횟수를 보면 문제가 된다. 임의의 5만개의 페이지 URL에 대해 100명 미만이 북마킹한 URL이 80%를 차지하였으며, 이러한 비율은 전체 URL에서도 비슷한 양상을 보인다.



(그림 2) 북마킹한 사용자의 수에 따른 URL 수

따라서 다수의 사용자가 북마킹한 URL과 태그들의 연관도 값(M_{ij})이 커지는 문제가 발생하게 되는데 이러한 문제를 해결하기 위해서 해당 URL을 북마킹한 모든 사용자의 수로 나누어 정규화하는 방법을 사용할 수 있다.($U_{ij}/B(u_j)$) 하지만, 이러한 변형은 다른 문제점을 야기한다.



(그림 3) 사용자의 태그 사용 횟수

그림 3은 딜리셔스 사이트 사용자의 태그 사용 횟수를 나타내는 그래프이다. X축은 사용자들이 사용한 태그들이며 Y축은 태그들의 사용 횟수를 나타낸다. 그림에서 볼 수 있듯이 사용자의 태그 활용 행태는 롱테일 (long-tail)의 모습을 띄고 있다. 1000명 이상이 태그로 사용한 단어의 개수는 28097개 이고, 1000명 미만이 태그로 사용한 단어의 수는 약 12만개였다. 롱테일의 꼬리 부분에 위치하는 태그에는 사용자의 id, 오탈자. 의미 없는 기호 등이 사용된 태그도 다수가 존재하였고, 이러한 태그들은 질의어 확장이나 추천, 재순위화에 이용될 경우 검색 시스템에 큰 오류를 야기 할 수 있다. 따라서 해당 URL과 크게 관계없는 태그가 북마킹한 사용자의 수가 작은 URL에서 높은 연관도를 가지는 경우를 해결을 위해 우리는 개인화된 연관 검색어 추천 과정에서 연관도 값(M_{ij})에 대한 정규화를 적용하였다. (3.3절 참조)

또한 IDF 값을 구하기 위해 태그가 포함된 URL의 수와 전체 URL의 수를 구해야 하는데, URL을 수집하는 방법과 수집한 전체 URL 데이터의 양에 따라 큰 차이를 보일 수 있으므로, 딜리셔스 사이트에서 정확히 제공하는 태그의 북마킹 횟수를 통하여 가중치를 계산할 수 있게 IDF 값을 전체 URL의 중에 해당 URL이 포함된 URL의 값으로 계산하지 않고, 모든 태그로 북마킹된 횟수와 특정 태그가 해당 URL을 북마킹하기 위해 사용된 횟수로 변형하여 다음과 같이 M_{ij}값을 계산하였다.

$$M_{ij} = U_{ij}/B(u_j) * \log_{10}(N/C(t_i))$$

U_{ij} : URL j를 태그 i로 북마킹한 사용자 수
$B(u_j)$: URL j를 북마킹한 모든 사용자 수
N : 모든 태그의 북마킹된 횟수 합
$C(t_i)$: t_i 태그로 북마킹된 횟수

따라서 M_{ij} 값은 문서와 가장 연관이 높은 단어를 구하기 위한 TF-IDF 가중치를 활용한 URL과 가장 연관이 높은 태그를 구분하기 위한 가중치 값이 된다. $U_{ij}/B(u_j)$값은 해당 URL을 북마킹한 횟수가 가장 높은 태그를 구하기 위한 TF값이 되며, $\log_{10}(N/C(t_i))$ 값은 전체 URL들을 북마킹하기 위하여 자주 사용되는 태그가 아닌 특정 URL에서만 자주 사용되는 태그를 구분하기 위한 IDF값이 된다.

3.2. 계층적 군집 클러스터링

딜리셔스 사이트의 태그와 URL의 관계를 살

펴보면, 유사한 개념(concept)을 가지는 URL들은 연관성이 높은 태그들로 이루어져 있으며 연관성이 높은 태그들로 이루어진 URL들은 유사한 개념의 URL들이었다. 또한 사용자들에 의해 해당 URL의 문서를 분류하는 기준으로 삼은 단어는 검색 엔진에서 해당 문서를 검색하기 위한 질의어로 사용되는 경우가 많았다. 따라서 해당 질의어와 연관성이 높은 URL의 태그들 중 앞 절에서 구한 연관도 값(M_{ij})이 높은 태그를 연관 질의어로 추천해준다면 효과적인 추천이 될 것이다. 이러한 연관 검색어를 추천해 주기 위해서 먼저 연관성이 높은 URL들을 구분하여 클러스터링하는 과정이 필요하다. 우리는 연관성이 높은 URL들을 클러스터링 하기위하여, 2.1절에서 보았던 코사인 유사도 공식[7,8,15]을 이용하여 URL간의 유사도를 계산하였다.

$$\mathrm{sim}(URL_i, URL_j) = \frac{\overrightarrow{URL_1} \bullet \overrightarrow{URL_2}}{|\overrightarrow{URL_1}| \times |\overrightarrow{URL_2}|} = \frac{\sum_{i=1}^{T} M_{i,URL_1} \times M_{i,URL_2}}{\sqrt{\sum_{i=1}^{T} M_{i,URL_1}^2} \times \sqrt{\sum_{i=1}^{T} M_{i,URL_2}^2}}$$

$\overrightarrow{URL_1}$, $\overrightarrow{URL_2}$: 두 URL의 태그 벡터
T : 두 URL에 포함된 모든 태그 수

이러한 유사도를 기반으로 계층적 군집 클러스터링(Hierarchical agglomerative clustering) 알고리즘[15]을 사용하여 URL들을 클러스터링 하였다. 그림 3에서 볼 수 있듯이, 일반적으로 다수의 사용자에 의해 사용된 중요한 태그의 수는 소수의 사용자가 사용한 덜 중요하거나 무의미한 태그의 수보다 훨씬 적다. 따라서, 코사인 유사도 계산을 위해 전체 태그 벡터를 이용한다면 실제로 유사한 URL이지만 유사도 값이 낮은 경우가 나타날 수 있다. 유사도의 정확도를 높이기 위해, 본 논문에서는 각 URL의 상위 20개의 태그만을 사용하였다. 또한, 대부분의 URL들은 북마킹 횟수가 적기 때문에 URL에 포함된 태그의 수도 적으므로(그림 2), 이러한 URL들의 유사도 보정을 위해서 각 URL에서 연관도가 높은 상위의 태그가 두 URL 모두에 3개 이상 존재하는 경우에는 두 URL간의 유사도 값을 증가시켜 주었으며 유사도가 낮은 경우에도 클러스터링 되는 것을 막기 위하여 한계점(threshold) 미만의 값은 클러스터링 되는 것을 제한하였다.

이를 기반으로 <표 1>, <표 2> 와 같이 URL들의 클러스터링 정보와 해당 클러스터에 포함된 태그 정보를 구성한 데이터를 생성하였다. 데이터의 저장은 검색 속도를 위해서 역파일(inverted file) 형식으로 저장하였으며, URL 번호는 클러스터에 포함된 URL 중 몇 번째 URL인지를 나타낸다.

<표 1> 클러스터링 정보

Cluster No.	URL No.	URL (http://www 생략)
#1	#31370	lifehacker.com
#1	#22570	lifehack.org
#2	#9970	kannel.org
#2	#5609	hiptools.net/sms/
...		

<표 2> 클러스터내의 태그 정보

Term	Cluster No.
fashion	#7206
style	#7206
blog	#7206
photos	#7206
...	...

3.3. 사용자 프로파일 생성 및 개인화된 검색어 추천

클러스터 데이터를 바탕으로 각 사용자의 북마킹 정보를 저장한 사용자 프로파일은 <표 3>과 같이 생성된다. 사용자 프로파일은 각 사용자가 북마킹한 URL과 그 URL이 속한 클러스터 정보를 나타낸다.

<표 3> 사용자 프로파일 정보

User ID	Cluster No.	URL No.
User 1	#8999	#5
User 1	#1958	#1
User 2	#200	#2
User 2	#200	#3
...		

<표 3>의 사용자 프로파일 정보를 이용하여 개인화된 검색어 추천을 위해 다음과 같은 간단한 방법을 사용할 수 있다.

1. 사용자가 전송한 질의어가 포함된 클러스터 번호를 확인
2. 해당 클러스터내의 태그 중 M_{ij} 값이 높은 태그 순으로 연관 질의어를 추천

그러나 이러한 방법을 그대로 사용할 경우 몇 가지 문제점이 있을 수 있다. 먼저 3.1절에서 기술했듯이, 극히 적은 수의 사용자가 사용한, 해당 URL과 크게 관계없는 태그가, 북마킹한 사용자의 수가 적은 URL에서 높은 가중치를 가지게 되어 그 태그가 추천 될 수 있다. 이 문제를 해결하기 위해 클러스터 내에서 모든 태그들의 M_{ij} 값이 URL의 북마킹 횟수에 관계없이 일정한 값을 가지도록 식(1)과 같이 평준화 하였다.

$$M_{ij} / \sqrt{\sum_{i=1}^{m} M_{ij}^{2}}\ ,\ \text{m: 클러스터내의 URL 수} \qquad (1)$$

그리고 해당 클러스터에서 가장 중요한 태그의 M_{ij} 값을 높여주기 위해 식(2)와 같이 클러스터 내에서 해당 태그를 포함하는 URL의 비율을 식(1)에 곱해주었다.

$$U(tag)/m,\ U(tag)\text{: 태그를 포함하는 URL수} \qquad (2)$$

이러한 방법을 통해 하나의 클러스터 내에서 중요도가 높은 질의어를 추천해 주는 문제는 해결되었지만, 실제 질의어는 여러 개의 클러스터 내에 포함될 수 있기 때문에 각 클러스터 내에서의 가중치 값이 달라야한다. 따라서 사용자가 북마킹한 URL이 많은 클러스터 순으로 추천해 주기 위해서 사용자가 클러스터 내에 북마킹한 횟수의 값을 식(2)에 곱해 주었다.

$$\left(M_{ij} / \sqrt{\sum_{i=1}^{m} M_{ij}^{2}} \right) * \left(U(tag)/m \right) * \log(1+C) \qquad (3)$$

C는 각 사용자가 해당 클러스터내의 URL을 북마킹한 횟수

사용자가 전송한 질의어가 포함된 클러스터의 태그들 중에서 식(3)의 값이 높은 태그들을 추천해 준다면, 사용자가 북마킹한 URL에 따라 개인화된 검색어 추천 결과를 보여줄 것이다. 본 연구에서는 사용자가 전송한 질의어와의 연관성이 식(3)에 따라 가장 높은 상위 5개의 태그를 추천해주었고, 한계점(threshold) 값을 정해 너무 낮은 값의 태그가 추천되는 것을 방지하였다.

지금까지의 방법을 통해 동일한 클러스터에 속하는 URL(http://maxthon.com/download.htm, http://icab.de)을 북마킹한 사용자가 'internet'이란 질의어를 전송했을 때, 추천해주는 연관 검색어의 결과는 <표 4>와 같다.

<표 4> 연관 검색어 추천 결과

연관 검색어 추천 결과	browser, software, web
사용자의 북마킹 정보	maxthon.com/download.htm
	http://icab.de

위의 결과는 식(3)에 따라 사용자가 관심을 가지는 URL이 포함된 클러스터를 대표하는 태그들을 추천해 주게 되었다. 하지만 이 방법은 각 사용자가 북마킹한 URL만을 대표하는 태그들이 질의어를 전송한 사용자에게 중요한 의미를 가질 수 있지만, 식(3)에 의해 값이 낮아지기 때문에 연관 검색어로 추천되지 못한다. 따라서 해당 클러스터 내에서 사용자가 북마킹한 URL들 중 M_{ij} 값이 높은 상위 10개의 태그 역시 연관 검색어로 추천해 주었다. 이러한 방법을 통해 클러스터를 대표하는 태그뿐 아니라 사용자가 북마킹한 URL을 대표하는 태그 역시 연관 검색어로 추천해 준 결과는 표 5와 같다.

<표 5>에 나타난 결과는 동일한 클러스터 내의 서로 다른 URL을 북마킹한 두 사용자가 동일한 질의어인 'browser'를 전송 했을 때, 두 사

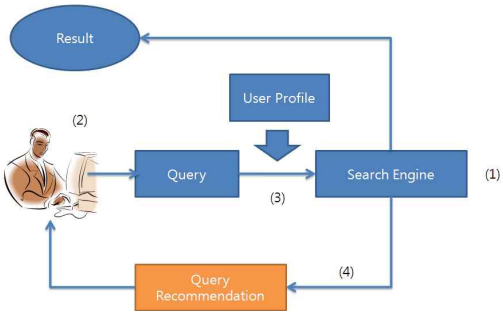
용자에게 연관 검색어를 추천해 준 결과를 보여준다. 같은 클러스터내의 URL을 북마킹했기 때문에 해당 클러스터를 대표하는 단어인 'software'와 'web'은 동일하게 추천되었으며, 추가적으로 두 사용자가 각각 북마킹한 URL을 대표하는 연관 검색어들이 추천되었다는 것을 알 수 있다.

<표 5> 연관 검색어 추천 결과

연관 검색어 추천 결과	(software, web), firefox, mac, internet, sleipnir, japan, tool, windows, osx, apple, freeware
사용자의 북마킹 정보	http://icab.de
	http://www.fenrir-inc.com/us/sleipnir/
연관 검색어 추천 결과	(software, web), firefox, statistics, internet, maxthon, tools, stats, market, download, free, marketshare
사용자의 북마킹 정보	http://maxthon.com/download.htm
	http://marketshare.hitslink.com/report.aspx?qprid=0

4. 시스템 구성 및 서비스 절차

3장에서 제안된 개인화 검색을 구현하기 위해서는 사용자 프로파일이 서버에 저장되어 서비스를 제공해 주어야 한다. 하지만 사용자 프로파일을 서버에 저장하는 것은 프라이버시의 위협요소가 된다[3]. 이러한 문제를 해결하기 위해서는 사용자 프로파일 정보가 클라이언트에 저장되어야 하지만, 클라이언트에서 서버 측의 모든 클러스터 정보를 저장하기 위한 저장 공간이 필요해 질 뿐 아니라 클라이언트 측에서만 개인화 검색 서비스를 제공해준다면 검색 엔진에서 해당 정보를 통하여 사용자에게 특화된 서비스를 제공하기 어려워진다. 이 장에서는 이러한 문제를 해결하기 위한 시스템 구성을 제안한다. (그림 4)는 본 논문에서 제안하는 개인화된 검색 서비스 절차를 나타낸다.



(그림 4) 개인화된 검색 서비스 절차

(1) 서버에서 인터넷상의 태그 데이터를 수집, 분석하여 클러스터 정보를 저장한다. (3.1-2장)

(2) 클라이언트에서 자신이 북마킹한 URL을 서버에 전송하면, 서버로부터 전송 받은 데이터를 가지고 클러스터내 태그 정보와 사용자 프로파일을 생성한다.

(3) 검색엔진에 해당 질의어를 전송할 때, 질의어에 대한 사용자 프로파일 정보도 같이 전송한다.

(4) 서버는 사용자 프로파일 정보를 분석해 사용자에게 알맞은 연관 검색어를 보여준다.

위의 절차에 따르면, 사용자 프로파일 정보를 질의어와 같이 전송해 주어야 하기 때문에 클라이언트에 사용자 프로파일을 저장해야 한다. 이를 위해 먼저 사용자 프로파일을 저장하기위한 필요 공간과 사용자 프로파일을 생성하는 방법에 대해 기술하고, 사용자 프로파일을 질의어와 같이 전송하는 방법과 전송으로 인한 오버헤드에 대해 기술한다.

3장에서 보았던 <표 1>, <표 2>, <표 3>의 데이터를 클라이언트 측에서 태그 정보를 수집하고 분석하여 저장하기에는 성능 문제와 저장 공간의 문제가 발생할 수 있다. 따라서 <표 1>의 데이터는 서버 측에서 수집, 분석하여 저장하고, <표 2>와 <표 3>의 데이터는 사용자가 북마킹한 정보만을 서버와 통신을 통해 클라이언트 측에 생성하면 성능 문제와 저장 공간의 문제를 해결할 수 있게 된다.

사용자가 북마킹한 URL이 포함된 클러스터내의 태그 정보(<표 2>)를 저장하기 위한 데이터의 크기는 다음과 같다.

((태그 크기 + 클러스터번호 크기) × 각 클러스터내 태그 수) × 북마킹한 URL이 포함된 클러스터 수

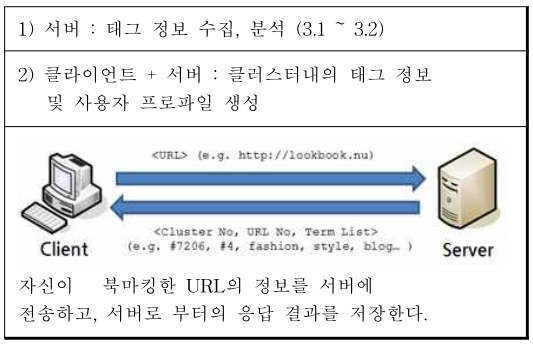
평균 단어의 크기를 10 byte, 클러스터 번호의 크기는 4 byte, 클러스터내의 태그 개수를 평균 100개 정도로 가정한다면, 하나의 클러스터 정보를 저장하는데, 약 1.4 KB 의 크기가 필요하다. 만약 사용자가 1만개의 클러스터에 관심이 있을 경우, 약 14 MB의 저장 공간을 사용하게 된다.

사용자 프로파일(<표 3>)을 클라이언트에 저장하기 위한 데이터에서는 사용자ID 필드를 제외한 정보가 저장되며, 크기는 다음과 같다.

(클러스터번호 크기 + URL번호 크기) × 북마킹한 URL 수

앞의 가정과 같이 사용자가 관심을 가지는 클러스터가 1만개이고 하나의 클러스터당 평균 10개의 URL을 북마킹했다면, 클러스터번호 크기가 4 byte, URL번호 크기를 4 byte로 가정해서 약 800 KB 정도의 저장 공간을 사용하게 된다. 따라서 전체 사용자 프로파일(10만개의 북마킹 URL의 정보와 약 백만 단어의 정보를 가정할 경우)을 저장하기 위해 약 15 MB의 용량이 필요하게 된다.

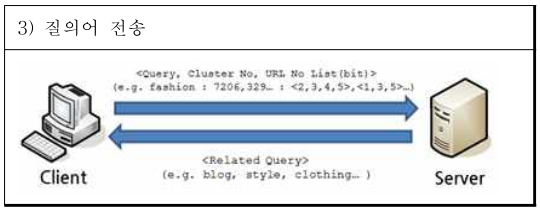
본 연구에서는 이 데이터들을 DB에 저장하였고, 데이터들을 생성하기 위한 진행 과정은 <그림 5>와 같다.



(그림 5) 태그 데이터 인덱싱 및 사용자 프로파일 생성 과정

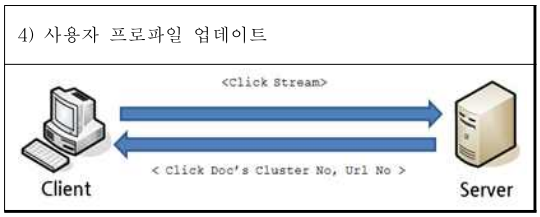
자신의 기본적인 북마킹 정보(딜리셔스 사이트의 혹은 로컬 컴퓨터에서 북마킹한 URL 주소)를 서버에 전송하면 해당 URL이 포함된 클러스터 번호, URL의 클러스터 내 번호, 해당 클러스터의 태그 리스트를 얻을 수 있게 된다. 이러한 과정은 개인화된 검색을 원하는 사용자의 선택에 의해 진행되며, 그 결과로 사용자 프로파일을 생성하게 된다. 한번 생성된 사용자 프로파일을 바탕으로 개인화된 검색에 지속적으로 이용하게 된다.

또, 앞에서 기술했듯이, 사용자 프로파일은 질의어와 같이 전송되기 때문에 전송 오버헤드가 생기게 된다. 질의어 전송과정은 그림 6과 같다.



(그림 6) 질의어 전송 과정

해당 질의어가 사용자가 관심을 가지는 클러스터 내에 포함된 단어인지를 검색하고, 포함된 단어라면 관련된 사용자 프로파일의 정보(클러스터 번호, 클러스터 내 북마킹한 URL 번호)를 전송해 주게 된다. 하나의 질의어가 4개의 클러스터에 포함되고 한 개의 클러스터 내의 북마킹한 URL이 평균 10개라 가정할 때, 질의어 전송시에 추가적으로 전송해야 하는 정보는 클러스터 번호(4 byte) × 4 와 URL 번호(4 byte) × 10으로 약 56 byte가 필요하게 되며, 북마킹한 URL이 많을수록 전송에 필요한 크기는 크게 증가한다. 따라서 이러한 오버헤드를 줄이기 위해 클러스터링 시 클러스터 당 최대 32개 정도의 URL을 포함하게 한다면, 4 byte로 32개의 URL 번호를 bit로 표현하여 전송이 가능하게 된다.



(그림 7) 프로파일 업데이트

서버에서는 클라이언트로부터 전송 받은 질의어와 사용자의 프로파일 정보를 가지고, 3장에서 보았던 개인화된 검색 추천 방법을 통하여 사용자에게 가장 알맞은 연관 검색어를 추천해 주게 된다. 하지만 한번 생성된 프로파일만을 개인화 검색에 활용하는 것은 시간이 지남에 따라 사용자의 선호도가 바뀌게 될 때, 적합한 개인화 검색 결과를 보여주지 못하게 된다. 따라서 서버 측에서는 (그림 7)과 같이 사용자가 선택한 문서에 대한 클러스터 번호와 URL 번호를 전송해 줌으로서 사용자가 원한다면 이러한 정보를 바탕으로 업데이트된 사용자 프로파일을 생성할 수 있게 된다.

5. 실험 결과

최종적으로 앞에서 제안한 방법들을 통하여, 패션 블로그와 음악 블로그에 관련된 10개의 페이지을 북마크한 가상의 사용자를 설정하여 'blog' 질의어에 대한 연관된 검색어 추천 결과를 구글과 야후의 연관 검색어와 비교해 보았다.

표 6과 표 7은 각각 구글과 야후에서의 연관 검색어 추천 결과를 나타낸다.

<표 6> 연관 검색어 추천 결과 (구글)

google blog, blogspot, blogger, fail blog, fashion blog, kanye west blog, blog search, free blog, blog 360

<표 7> 연관 검색어 추천 결과 (야후)

related concept	weblog, TypePad, Twitter, Google Blog search, nytimes, blogSpirit, blog spot, personal blog, WordPress, Beta, create blog, 06-05, ...
radio blog, yahoo blog, blog skins, free blog, blog search, blog templates, celebrity baby blog, pink is the new blog, anwar ibrahim blog, blog sites	

실제로 구글과 야후의 연관 검색어 추천 방식에 대해서는 정확하게 알 수 없다. 그러나, 표 6과 표 7에 나타난 결과를 바탕으로 유추하면, 사용자의 검색 기록(log)을 통하여 해당 질의어와 같이 다수의 사용자에 의해 많이 사용된 질의어를 추천해 주었을 것으로 추측된다. 야후의 경우 어떤 알고리즘을 통하여 연관된 개념의 단어를 추가적으로 추천해 준 것을 알 수 있다. 하지만 두 결과 모두 사용자가 선호하는 의미가 어떤 것인지를 모르기 때문에 개인화된 결과를 보여주지 못하고, 다양한 'blog'와 관련된 다양한 주제의 연관 검색어를 추천해 주었다.

표 8은 본 논문에서 제안한 개인화 검색 시스템의 연관 검색어 추천 결과이다. 임의의 사용자가 패션(상위의 5개 URL)과 음식(하위의 5개 URL) 관련 블로그를 북마킹했다는 가정 하에 연관 검색어를 추천해 주었다. 제안된 방법은 사용자 프로파일을 통해 사용자가 선호하는 정보를 알 수 있으므로 사용자에게 특화된 연관 검색어를 추천해 준다는 것을 표 8을 통해 확인할 수 있다.

<표 8> 제안 시스템의 연관 검색어 추천 결과

연관 검색어 추천 결과	recipe, cooking, food, fashion, style, clothing, moda, baking
	dessert, pizza, style, pie, menswear, men, nyc, trends, streetstyle, chaussures, fashionblog, tools, streetfashion, diy, international
사용자의 북마킹 정보	fashionindie.com/
	fashiontoast.com/
	permanentstyle.blogspot.com/
	lookbook.nu/
	stylesalvage.blogspot.com/
	foodbycountry.com/index.html
	bakerella.blogspot.com/2009/08/easy-as-pie.html
	seriouseats.com/2007/03/broiled-pizza.html
	copykat.com/
	tastyplanner.com/

6. 결론 및 향후연구

본 논문에서는 다수의 유저에 의해 만들어진 딜리셔스의 태그 데이터를 이용하여 사용자 프로파일을 생성하고, 생성된 사용자 프로파일 정보를 활용하여 개인화된 연관 검색어를 보여주는 기법을 제안하였다. 또한, 사용자 프로파일 정보가 서버나 클라이언트 어느 한 쪽에만 저장될 경우 발생되는 문제를 해결하기 위해, 본 연구에서는 사용자의 프라이버시 위협을 줄이면서도, 서버가 사용자의 프로파일 정보를 활용하여 개인화된 서비스를 제공할 수 있는 시스템 아키텍처를 제안하였다.

본 논문에서 제안된 연관 검색어 추천 방법은 검색 엔진에서 다수의 사용자에 의해 이슈가 되는 질의어를 모두 반영하지는 못한다. 따라서 전송되는 사용자 프로파일을 통해 질의어 기록(log)을 분석하여 사용자가 원하는 질의어만을 추가적으로 추천해 준다면 보다 좋은 연관 검색어 추천 결과를 보여주게 될 것이다.

본 논문에서는 간단하고 명확한 결과를 볼 수 있도록 연관 검색어 추천에 초점을 맞추어 개인화 시스템을 구현하였지만, 검색어 확장, 검색 결과의 재순위화, 개인화 맞춤 광고 등의 서비스에도 활용이 가능하며, 태그 추천 분야의 시스템 아키텍처로도 활용할 수 있을 것이다. 따라서 추후 연구를 통해 기존의 다양한 정보검색 분야의 기법들을 적용한 시스템 아키텍처를 구현하며, 개선점을 발견하여 보완하는 연구를 진행하고자 한다.

참 고 문 헌

[1] Dou, Z., Song, R., and Wen, J.R. (2007). A large-scale evaluation and analysis of personalized search strategies. In Proceedings of WWW ''07, 581-590.

[2] S. Wedig and O. Madani. A large-scale analysis of query logs for assessing personalization opportunities. In Proceedings of KDD ''06, pages 742 - -747, 2006.

[3] E. Volokh. Personalization and privacy. Communications of the ACM, 43(8):84 - -88, 2000.

[4] X. Shen, B. Tan, and C. Zhai. Implicit user modeling for personalized search. In Proceedings of CIKM '05, pages 824-831, 2005.

[5] Micarelli, A., Gasparetti, F., Sciarrone, F., and Gauch S.: Personalized Search on the World Wide Web. In Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.): The Adaptive Web: Methods and Strategies of Web Personalization, Lecture Notes in Computer Science, Vol. 4321. Springer-Verlag, Berlin Heidelberg New York (2007) this volume

[6] S. Xu, S. Bao, B. Fei, Z. Su, and Y. Yu. Exploring Folksonomy for Personalized Search. In Proc. of SIGIR' 08, 2008, 155-162.

[7] BAEZA-YATES, R. AND RIBIERO-NETO, B. 1999. Modern Information Retrieval. Addison-Wesley Longman, Boston, Mass.

[8] SALTON, G., WONG, A., AND YANG, C. 1975. A vector space model for automatic indexing. Commun. ACM 18, 11, 613 - -620.

[9] R. Li, S. Bao, B. Fei, Z. Su, and Y. Yu. Towards Effective Browsing of Large Scale Social Annotations. In WWW '07: Proceedings of the 16th international conference on World Wide Web, 2007. Track: Web Engineering, Session: End-User Perspectives and measurement in Web Engineering.

[10] P.A. Chirita, W. Nejdl, R. Paaju, and C. Kohlschutter. Personalized Query Expansion for the web. In Proceeding SIGIR '07

[11] R. Baeza-Yates, C. Hurtado, and M. Mendoza. Query recommendation using query logs in search engines. In International Workshop on Clustering Information over the Web (ClustWeb, in conjunction with EDBT), Creete, Greece, March, Springer, LNCS, 2004, 588-596.

[12] Z. Zhiyong, N. Olfa. Mining Search Engine Query Logs for Query Recommendation. In Proceeding WWW 2006.

[13] http://flickr.com/

[14] http://delicious.org/

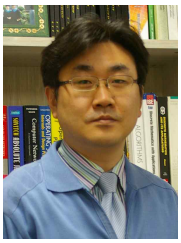
[15] Manning, C., Raghavan, P. and Schutze, H. (2008) Introduction to Information Retrieval, Cambridge University Press, Cambridge, MA.

김 동 욱

2009년 : 호서대학교 컴퓨터공학부 (학사)

현 재 : 한양대학교 전자컴퓨터통신공학과 (석사)

관심분야 : 정보 검색(Information Retrieval), 파일 시스템, 플래시메모리 기반 저장 시스템



강 수 용

1996년 : 서울대학교 수학과(학사)

1998년 : 서울대학교 전산과학과 (석사)

2002년 : 서울대학교 전기컴퓨터공학부 (박사)

2003년～현 재: 한양대학교 컴퓨터공학부 교수

관심분야 : 멀티미디어 시스템, 분산시스템, 플래시 메모리 기반 저장 시스템 등

김 한 준

1994년 : 서울대학교 계산통계학과 졸업 (이학사)

1996년 : 서울대학교 전산과학과 대학원 졸업 (이학석사)

2002년 : 서울대학교 컴퓨터공학부 대학원 졸업 (공학박사)

2002년～2002년 : 서울대학교 공과대학 박사후 연수과정

2002년～현 재 : 서울시립대학교 전자전기컴퓨터공학부 교수

관심분야 : 정보검색(Information Retrieval), 기계학습(Machine Learning), 데이터마이닝(Data Mining), 데이터베이스(Databases) 등

이 병 정

1990 : 서울대학교 계산통계학 (학사)

1998 : 서울대학교 전산과학(석사)

2002 : 서울대학교 컴퓨터공학 (박사)

1990년～1998년: 현대 전자 SW연구소

2002년～현 재: 서울시립대학교 컴퓨터과학부 교수

관심분야 : 소프트웨어 진화, 개발 방법론, 소프트웨어 품질 등