협동적 순위 평가와 워드넷을 이용한 검색엔진의 성능 향상

김형일[0] 김준태
동국대학교 컴퓨터공학과
{salmoner, jkim}@dgu.edu

Performance Inprovement of a Search Engine Using Collaborative Evaluation and The WordNet

Hyoung-Il Kim[0] Jun-Tae Kim
Dept. of Computer Engineering, Dongguk University

요 약

웹에서 사용자가 원하는 정보를 정확히 추출하기란 쉬운 문제가 아닐 것이다. 이러한 정보추출의 중요성의 문제는 방대해지는 정보의 양과 직결된다. 현재 웹의 정보는 사용자들이 원하는 모든 정보를 담고 있다고 이야기할 수 있을 만큼 많은 정보들이 내재되어 있다. 그러나 이러한 정보의 홍수 속에서 사용자들은 자신이 원하는 정보를 정확히 추출하기란 쉽지 않은 일이며, 정확히 추출이 되었어도 전통적 방식을 따르는 검색엔진은 내용기반 방식을 기초로 웹페이지의 순위를 결정함으로, 사용자에게 중요한 페이지를 상위에 위치시키기란 쉬운 일이 아니다. 본 논문에서는 이러한 전통적 방식의 검색엔진의 문제점을 해결하기 위하여 협동적 순위 평가 방법과 워드넷을 기반으로 검색엔진의 성능 향상 방법을 제안한다.

1. 서 론

웹의 급속한 성장은 시공간의 제약을 극복할 수 있는 가상공간이라는 점에서 가능할 수 있었으며, 이러한 정보의 집합장소인 웹은 정보이용이라는 장점을 내포한다. 그러나 방대한 정보는 역기능을 창출하기도 하였다. 이러한 정보의 역기능으로는 사용자에게 선택이라는 혼란을 안겨다 주기도하며, 방대한 정보에서는 원하는 정보를 획득하기 위해 많을 노력을 요구한다. 이러한 정보검색에서의 정보 추출의 문제점과 정보 선택의 문제점으로 인해, 현재는 정보검색의 편리성이 강조되어지고 있다. 이러한 정보추출의 편리성 문제로 인해서 현재는 검색엔진의 정보추출에 관한 다양한 알고리즘 연구가 활발히 진행되어 가고 있으며, 전통적 방식의 정보추출을 탈피한 검색엔진들이 상용화되고 있는 추세에 있다[4].

이러한 차세대 검색엔진의 연구에 힘입어 구글(Google), 다이렉트히트(DirectHit)와 같은 차세대 검색엔진이 등장하여 사용자들에게 많은 호응을 받고 있다. 차세대 검색엔진들은 전통적 방식을 따르는 일반 검색엔진들의 내용기반 방식에서 탈피하여 사용자의 질의에 대하여 보다 좋은 관련 웹 페이지들을 추출하기 위해 웹 페이지의 가중치를 내용기반 방식에 치중하지 않는 방식을 채택하고 있다. 그러나 이러한 시도에도 불구하고 검색 질의어의 다의성 문제로 인해 중요 페이지의 추출에는 여전히 어려움이 따르고 있다.

이러한 검색엔진에서의 문제점을 본 논문에서는 협동적 평가 방법과 워드넷을 이용하여 새로운 가중치 알고리즘을 고안하였다. 이러한 협동적 평가 방식을 이용하여 웹 페이지의 가중치를 결정하여 내용기반 방식을 탈피하였으며, 검색 질의어의 모호성 해결과 웹페이지의 가중치 저장을 위해 워드넷을 이용하였다. 이러한 결과 본 논문에서 제시한 검색엔진이 전통적 방식을 따르는 검색엔진과 차세대 검색엔진과의 비교에서 우수한 성능을 나타냄을 확인 할 수 있었다[2][3][5].

본 논문은 1999년 정보통신부 대학기초연구지원 사업의 연구결과임.

2. 관련연구

차세대 검색엔진 중 각광을 받고 있는 구글(Goole)은 스탠포드대학에서 개발이 진행되어 현재는 상용화된 검색엔진으로 국내에서도 많은 각광을 받고 있다. 구글에서 이용한 웹 페이지 가중치 방식은 웹 페이지들의 하이퍼링크(HyperLink)정보를 이용하여 결정하였다. 하이퍼링크정보는 특정 웹페이지 X가 웹페이지 Y를 하이퍼링크로 나타내었다면 X라는 웹페이지의 주제에 관해서 Y페이지는 중요도가 있다는 판단 하에 이용되어 졌을 것이다. 이러한 하이퍼링크 정보의 활용은 클라인베르그(Kleinberg)의 HITS 알고리즘에 잘 소개되어져 있다.

다이렉트히트(DirectHit)는 내용기반 방식의 웹페이지의 가중치 방식을 탈피하기 위하여 사용자 반응을 이용하여 검색엔진을 상용화하였다. 이때 사용한 사용자 반응이라 함은 특정 질의어에 대한 결과 웹 페이지가 사용자의 질의어와 부합할 경우에 사용자는 해당 웹 페이지를 검색하게 되고, 질의어에 합당한 정보들이 높은 가치를 소유하고 있을 경우에는 오랜 시간을 해당 웹 페이지에서 머무르게 된다. 이러한 사용자의 웹 페이지에 대한 반응을 다이렉트히트에서는 웹 페이지의 가중치로 이용하였다.

질의어의 모호성 해결과 웹페이지의 가중치 저장방식에서 사용한 워드넷(WordNet)은 프린스톤대학에서 개발한 온라인 사전이다[1]. 워드넷은 어휘의 표현을 의미를 이용하여 분류하여 놓았으며, 어휘의 의미관계를 동의, 반의, 상위, 하위 등으로 나타냄으로써 어휘의 의미 구조가 명확성을 갖는다[6][7]. 워드넷에서는 어휘의 의미 포함관계로 인해 계층적 구조를 갖는다. 이러한 워드넷의 명확한 어휘 분류로 인해 문서분류나 자연어처리 분야에서 많은 활용을 하고 있다[1][8][9]. 이와 같은 어휘의 계층구조를 문장 형태의 질의에 사용한다면 검색엔진의 성능 향상에 많은 기여를 할 것이다[9].

3. 실험용 검색엔진

3.1웹 페이지 가중치 알고리즘

본 논문에서는 내용기반 가중치 방식을 탈피하여 새로운 방법의 웹페이지 가중치 알고리즘을 제안하였다. 본 논문에서 사용한 가중치 알고리즘은 사용자들의 반응을 활용하여 협동적 방식의 가중치 방법을 사용하였다. 이때 협동적 방식의 가중치 방법은 사용자가 특정 질의어에 대해서 명시적 반응을 보인 웹페이지에 대해 가중치를 높임으로써 인기도(Popularity)를 반영하였다. 이때 사용한 가중치의 저장방식은 워드넷의 카테고리를 사용하여 가중치 데이터베이스에 저장함으로 질의어의 모호성 해결을 시도하였다. 본 논문에서 사용한 가중치 알고리즘은 [표 1]에 나타내었다.

```
Let Input_URL be an URL clicked by an user.
Let Weighting_DB be the database containing pairs of
an URL and it's weight.
Let i be a index of category in the WordNet for the
query.
Let CWi be a weight for the Category i.(i = 1,2,3,....,26)
Let TW be  Σ CW_i  , (i=1,2,3,..,26)
Initialize CWi & TW.

If ( Input_URL =  URL in Weighting_DB )
Then{
    CWi=CWi + Weighting_Value, ( i = 1,2,3,....,26 )
    TW = TW + Weighting_Value
    }
Else {
    Insert Categoryi &URL of Input_URL into Weighting_DB
    CWi = CWi + Weighting_Value, ( i = 1,2,3,....,26 )
    TW = TW + Weighting_Value
    }
```

[표 1] 웹페이지 가중치 알고리즘

3.2워드넷 기반의 가중치 데이터베이스

대다수의 검색엔진에서는 가중치 데이터베이스를 특정 질의어에 대한 가중치 값과 해당 URL의 형식을 기초로 구성한다. 그러나 이러한 내용기반 방식은 특정 질의어의 의미는 배제된 상태에서 가중치를 부여하여 저장함으로써 웹페이지의 가중치 저장 시 오류를 내포하고 있다. 또한 사용자 반응을 이용한 가중치 방식에서도 고유의 협동적 가중치 방식을 따름으로써 질의어의 다의성 해결에 대한 취약성으로 인해 웹페이지 결과 추출에 오류를 내포하고 있다. 이와 같은 결과 웹페이지의 추출에 대한 오류의 예제를 [표 2]에서 나타내었다.

검 색 어	URL_1의 가중치	URL_2의 가중치
Island (섬과 관련된 질의어)	50	10
Program Language (프로그램언어 관련 질의어)	10	10
Organization (단체에 관련한 질의어)	10	100
가중치의 총합	70	120

[표 2] 웹페이지 가중치 방식에 대한 카테고리 분류예제

[표 2]에서 URL_1은 섬 관련 가중치로 50, 프로그램언어 관련 가중치로 10, 단체 관련 가중치로 10을 가지고 있다. 이러한 URL의 가중치가 저장되어 있을 때, 검색질의어로 JAVA(섬)가 사용되어진다면 일반 검색엔진에서는 URL_2의 가중치 총합이 가장 높기 때문에 결과 웹페이지로 URL_2가 나타나게 될 것이다. 그러나 위 표를 보면 해당 질의어의 의미에서는 URL_1이 URL_2보다 중요도가 높은 웹페이지이다. 이러한 가중치 방식의 문제점 해결과 질의어의 모호성 해결을 위해 본 논문에서는 워드넷을 활용하여 해결할 수 있었다.

그리고 사용자의 의미 선택행위가 없이 JAVA(내포 의미 : 섬)로 질의어를 던졌을 경우에 본 논문에서 제안한 검색엔진은 JAVA의 가중치만을 고려하여 URL_1을 결과로 추출함으로써 일반 검색엔진에 비하여 우수한 성능을 보인다.

아래 [표 3]은 본 논문에서 제시한 워드넷 기반의 가중치 데이터베이스이다. 이때 가중치 데이터베이스에는 워드넷의 최상위 카테고리 26개를 이용하여 특정 웹 페이지에 대하여 각각 가중치를 기록하도록 하여 가중치 계산에서 문제점을 해결할 수 있었다. 이때 C1부터 C26은 워드넷의 최상위 카테고리이며 CW_1부터 CW_{26}은 해당 URL의 카테고리별 가중치이다.

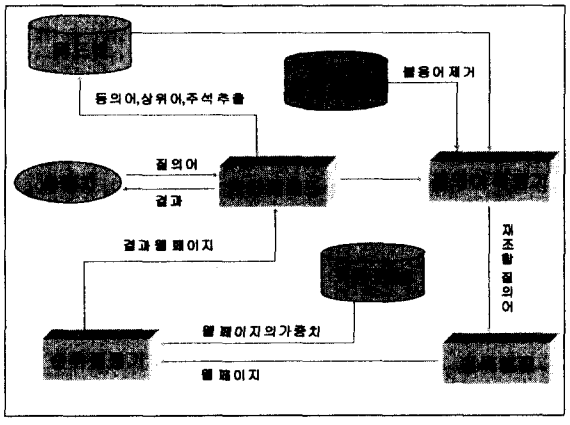
URL_i (i = 1, 2, 3, ... , n)					
Category	C1	C2	C3	·····	C26
Weighting	CW_1	CW_2	CW_3		CW_{26}

[표 3]가중치 데이터베이스의 형식

3.3시스템 구성도

사용자가 사용자 인터페이스를 통하여 질의를 할 경우, 사용자 인터페이스는 워드넷을 연동하여 사용자에게 특정 질의어의 의미를 나타내 줌으로써 질의어의 다의성을 해결하게 하였다. 이렇게 얻어진 질의어의 의미는 질의어 확장기를 통하여 재 조합된 질의어를 형성한다. 재 조합된 질의어는 웹페이지 수집기를 통하여 웹페이지들을 수집하고 순위결정기를 통해 웹페이지의 가중치가 결정되게 된다. 이렇게 가중치를 조정 받은 웹페이지들은 사용자 인터페이스를 통하여 사용자에게 보여진다.

사용자에게 결과 웹페이지로 보여줄 때는 웹페이지 순위결정기의 참여로 가중치가 높은 웹페이지부터 보여지게 된다. 결과 웹페이지에 대하여 사용자가 반응을 보인 웹페이지에 대해서는 협동적 가중치 모듈에 의해 가중치 값을 조정 받아 가중치 데이터베이스의 해당 웹페이지의 가중치 값을 갱신하게 된다. 이러한 협동적 방식의 사용으로 검색엔진 사용자들은 다른 사용자의 의견을 수렴함으로써 검색의 편리성도 얻어낼 수 있게된다. 실험 검색엔진의 시스템 구성도는 [그림 1]과 같다.



[그림 1]시스템 구성도

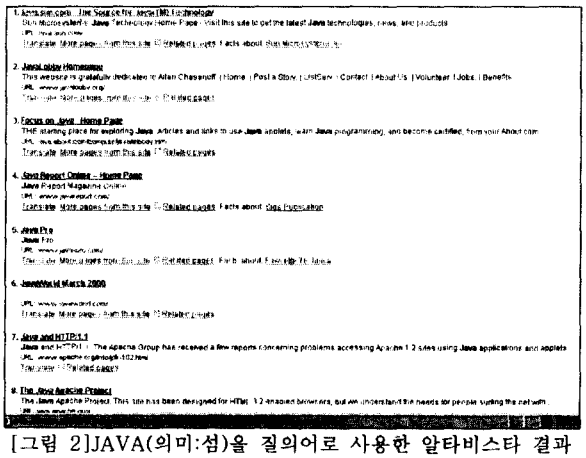
4. 실험 결과 및 분석

본 논문의 실험을 위하여 1,170개의 웹페이지를 실험 데이터를 선정하여 실험에 임하였다. 본 논문에서 제시한 협동적 방식의 가중치는 사용자 반응을 이용함으로써, 검색엔진 사용자들의 반응이 필요하게되어 추출한 웹페이지들에 대하여 동국대학교 컴퓨터공학과 학생들 150명을 선정하여 사용자 반응을 요구하였다. 본 실험용 검색엔진의 성능측정을 위하여 전통적 방식을 따르는 검색엔진인 알타비스타와 차세대 검색엔진으로 각광받고 있는 구글을 이용하여 비교실험에 임하였다. 비교 실험에서는 결과 웹페이지 중 상위 10개만을 추출하여 비교실험에 사용하였다. 실험에서 얻어진 결과는 [표 4]와 같다.

사용된 질의어			상용 검색엔진		실험용 검색엔진
형태	의미	질의어에 해당하는 카테고리	Altavista	Google	
Java	**커피**	**Food**	**0**	**0**	**4**
Java	**섬**	**Location**	**0**	**0**	**6**
Java	언어	Communication	10	10	10
Custom	**통관**	**Possession**	**0**	**0**	**8**
Custom	관습	Cognition	3	4	6
Horse	**마약**	**Artifact**	**0**	**0**	**4**
Horse	말	Animal	7	8	10
Bill	**지폐**	**Communication**	**0**	**1**	**3**
Bill	법안	Communication	2	3	8
Plant	**공장**	**Artifact**	**2**	**1**	**6**
Plant	식물	Plant	6	5	7
Sentence	**문장**	**Communication**	**0**	**0**	**9**
Sentence	판결	Action	5	6	7
관련문서의 개수			**35**	**38**	**88**
평 균 정 확 도			**26.9%**	**29.2%**	**67.7%**

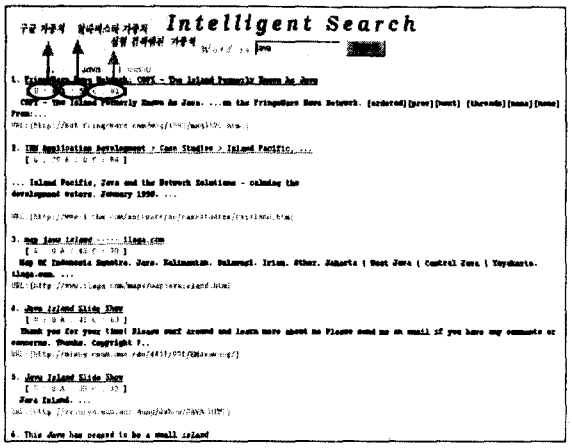
[표 4] 실험결과

다음 [표 4]을 보더라도 실험용 검색엔진이 기존의 검색엔진보다 관련도 높은 웹페이지를 상위에 위치시킴을 알 수 있다. 아래 [그림 2]와[그림 3]은 질의어를 JAVA(의미 : 섬)로 사용하여 알타비스타와 실험용 검색엔진에서 나온 결과이다. 이 결과에서도 알 수 있듯이 기존의 검색엔진은 질의어의 모호성에 전혀 대응할 수 없었으며, 상위에 적합한 웹 페이지도 위치시키지 못하였다. 본 논문에서 제안한 웹페이지의 가중치 알고리즘과 순위결정기는 웹페이지의 중요도를 판단하여 결과 출력시, 중요도 순서로 웹페이지를 위치시킴으로써 사용자에게 검색의 편리성도 제공하였다.



[그림 2]JAVA(의미:섬)을 질의어로 사용한 알타비스타 결과

[그림 3]은 본 논문에서 제안한 가중치 알고리즘을 활용한 실험용 검색엔진이다. 결과 웹 페이지에는 검색엔진들의 가중치를 표시하였다. G는 구글에서의 가중치 값, A는 알타비스타의 가중치 값, C는 실험용 검색엔진의 가중치 값을 나타낸다.



[그림 3]JAVA(의미:섬)을 질의어로 사용한 실험엔진 결과

5. 결론 및 향후 연구과제

본 논문에서는 사용자의 반응을 이용한 가중치 방법을 제안하여 검색엔진에 활용하여 성능향상을 이루었다. 그리고 검색엔진의 가중치 방식의 문제점을 해결하기 위하여 워드넷의 카테고리 구조를 데이터베이스에서 활용하여 질의어의 모호성 해결까지 이룰 수 있었다. 이러한 전반적인 연구과정을 통하여 실험용 검색엔진은 상용 검색엔진에 비하여 우수한 검색결과를 나타내 주었다. 향후 연구과제로는 질의어의 형태가 문장으로 이루어졌을 경우 사용자의 반응을 받지 않은 상태에서 질의어들 사이에서의 워드넷 카테고리 구조를 분석하여 검색 질의 문장을 확장하는 사용자 인터페이스를 개발하는 것이다.

6. 참고문헌

[1]C. Fellbaum, "WordNet : An Electronic Lexical Database", MIT Press,1998
[2]W. Frakes, and R.Baeza-Yates, "Information Retrieval : Data Structures & Algorithm", Prentice-Hall,1992
[3]R. Hoch, "Using IR Techniques for text classification in document analysis", SIGIR'94, 1994
[4]Hyoungil Kim, Kyeonah Yu, Juntae Kim, "Resolving Ambiguity in Search Query by Using the WordNet", International Conference on Artificial Intelligence and Soft Computing, 2001
[5]Xiaobin Li, Stan Szpakowicz and Stan Matwin, "A WordNet-based Algorithm for Word Sense Disambiguation", IJCAI-95, 1995
[6]Miller, "WordNet : An On-Line Lexical Database", International Journal of Lexicography, 1990
[7]S. Scott, and S. Matwin, "Text Classification Using WordNet Hypernyms", Coling-ACL '98Workshop, 1998
[8]Eric Siegel, "Disambiguating Verbs with the WordNet Category of the Direct Object", Coling-ACL '98 workshop,1998
[9]Ellen M. Voohees, "Query Expansion Using Lexical-Semantic Relations", SIGIR'94, 1994
[10]WordNet, http://www.cogsci.princeton.edu/~wn/