단어기반 웹문서 검색을 위한 효과적인 단어 가중치의 계산

권 순 만° 박 병 준
광운대학교 컴퓨터과학과
kw43hitel@kw.ac.kr°, bjpark@cs.kw.ac.kr

Efficient Term Weighting For Term-based Web Document Search

Soon Man Kwon° Byung Joon Park
Department of Computer science, Kwangwoon University.

요 약

웹(WWW)은 방대한 양의 정보들과 함께 그에 따른 웹의 환경과 그에 따른 정보도 증가하게 되었다. 그에 따라 사용자가 찾고자 하는 정보가 잘 표현된 웹 문서를 효과적으로 찾는 것은 중요한 일이 되었다. 단어기반의 검색에서는 사용자가 찾고자 하는 단어가 나타난 문서들을 사용자에게 보여주게 된다. 검색 단어를 가지고 문서에 대한 가중치를 계산하게 되는데, 본 논문에서는 이러한 단어기반의 검색에서 단어에 대한 가중치를 효과적으로 계산하는 방법을 제시한다. 기존의 방식은 단어가 나타난 빈도수에 한정되어진 계산을 하게 되는 반면, 수정된 방식은 태그별로 분류를 통한 차별화된 가중치를 부여하여 계산된다. 기존의 방식과 비교한 결과 본 논문에서 제시한 수정된 방식이 더 높은 정확도를 나타냈다.

1. 서론

시간이 지남에 웹의 환경과 웹 문서들은 매우 빠르게 증가하고 있다. 이에 따라 방대한 규모의 웹 문서에서 사용자가 원하는 정보를 효과적으로 찾는 것은 매우 중요한 일이 되었다. 초기의 검색 시스템들은 단순히 웹 문서를 검색하는 검색어가 포함된 문서들을 사용자에게 보여주는 방식으로, 재현율(recall)을 높이기 위해 관련 문서를 모두 찾는 것에 중점을 두었다. 이 방식은 단순히 검색어가 포함된 모든 문서를 검색 결과로 제시하므로 사용자가 필요로 하지 않는 문서들을 많이 포함하게 된다[1].

본 논문에서는 정보검색시스템에서 단어기반의 검색을 보다 효율적으로 수행하기 위해서, html문서를 태그별로 분석하여 차별화된 가중치를 부여하여 정확도(precision)를 높이는 방법에 대한 연구를 하였다.

본 논문에서는 2장에서 단어기반의 검색에 대한 소개를 하고 3장에서는 태그별로 차별화된 가중치를 부여하는 방법에 대해서 논할 것이고, 4장에서는 실험과 평가를 5장에서는 본 논문에 대한 결론을 내린다.

2. 단어기반의 검색(Term-based search)

단어기반 검색은 웹 검색에서 가장 일반적인 방식으로, 사용자가 찾고자하는 검색어를 포함하는 웹 문서들을 검색하여 사용자에게 보여주는 방식이다.

2.1 TF-Idf

단어 가중치(Term weight)는 시스템에 저장된 웹 페이지들과 사용자의 질의어(query)와의 유사성 정도를 계산하는데 사용된다[2]. 단어(Term)의 가중치 계산은 TF-idf 방식을 사용하여 계산되어 질 수 있다[3].

단어와 문서의 관련성은 단어빈도수(**tf**, term frequency)와 문서빈도수(**df**, document frequency)로 표현된다. **tf**는 특정 단어의 출현 빈도를 나타내고, 이는 특정 단어가 얼마나 문서의 내용을 잘 표현하고 있는지를 나타낸다. **df**는 특정 단어가 출현한 문서의 수를 나타내고, 이는 특정단어가 가지는 변별성 정도를 나타낸다. **df**가 큰 단어는 관련 문서와 비관련 문서를 구분하는데 별로 유용하지 못한 반면, **df**가 작은 단어는 구분하는데 용이함을 뜻하게 된다[1]. N을 시스템에 있는 전체 문서의 수로 정의하고, n_i를 단어 k_i로 검색하여 나타난 문서의 수로 정의하자. $freq_{i,j}$를 문서 d_j에서 단어 k_i가 출연한 수로 정의하자. 아래와 같이 문서에 대한 가중치를 계산할 수 있다[4].

$$w_{i,j} = freq_{i,j} \times \log \frac{N}{n_i}$$

2.2 재현율(Recall)과 정확도(Precision)

정보검색 시스템(IR)에서 상위 N개(top-N)의 추천을 평가하기 위해서 가장 공통적으로 널리 사용되는 방법은 recall과 precision이다[5,6].

- 재현율(Recall)

사용자가 찾고자하는 관련 문서들을 test집합으로 가정하고, hit집합을 검색되어져 나온 관련 문서들로 정의하자. Recall은 hit 집합을 test 집합으로 나누어 계산한다. 이를 수식으로 나타내면 아래와 같다.

$$recall = \frac{size\ of\ hit\ set}{size\ of\ test\ set} = \frac{|test \cap top\text{-}N|}{|test|}$$

- 정확도(Precision)

Precision은 hit 집합을 상위 N개의 집합(Top-N)으로 나누어 준다. 이를 수식으로 나타내면 아래와 같다.

$$precision = \frac{size\ of\ hit\ set}{size\ of\ top\text{-}N\ set} = \frac{|test \cap top\text{-}N|}{|N|}$$

이들 두 개의 측정값들은 거의 사실상 상반된다. N이 커질수록, recall은 증가하게 되고, 동시에 precision은 감소한다. 그러나, 두 개의 측정값들은 상위 N개의 추천

들을 발생시키는 시스템들의 질(quality)을 측정하는데 중요하다[7,8].

3. 차별화된 가중치의 선정과 계산

3.1 가중치의 선정의 배경

본 논문에서는 단어 기반의 검색을 기초로 하고 있다. 웹 문서가 **html**태그로 이루어졌다는 것을 강안하여서[9], 사용자가 찾고자 하는 단어가 **html**태그의 어느 부분에 나타나는지에 대하여 서로 다른 가중치를 주었다.

html태그별로 가중치를 고려하지 않은 문서와 고려한 문서와의 차이는 그림1과 그림2를 비교한 표1을 통해 알 수 있다.

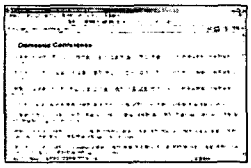

그림 1 A.html

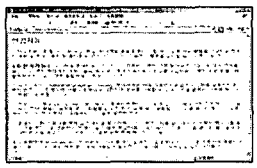

그림 2 B.html

표 1 A.html과 B.html의 비교

구분	A.html	B.html
단어빈도수(tf)	6	5
TF-idf	20.3	18.2
수정된 TF-idf	30.4	60.9

표 1은 "인공지능"이란 단어로 검색한 결과로 나온 두 개의 문서를 비교한 것이다. A.html의 단어빈도수(tf)는 6이고, B.html의 단어빈도수(tf)는 5로 나왔다. 기존의 TF-idf 방식에 따른다면, A.html문서가 B.html문서의 가중치 보다 높기 때문에, B.html문서를 추천하기 보다는 A.html문서를 우선적으로 추천하게 된다. 그러나, A.html문서는 "인공지능"에 대한 내용을 표현한 문서가 아니고, "인공지능"이란 단어가 많이 들어있는 문서이다. 오히려 B.html의 내용이 "인공지능"에 대한 내용을 더 잘 표현한 문서이다. 따라서 기존의 방식에서는 이러한 문제점이 있다고 하겠다. 이런 문제를 해결하기 위해, 수정된 TF-idf방식을 적용시키게 되면, A.html문서의 가중치보다 B.html문서의 가중치가 더 커지게 되므로, B.html문서를 추천하게 된다.

3.2 추출할 html 태그의 선정 및 추출

인터넷을 이용하는 사용자는 웹 브라우저를 사용한다. 이때 보여지는 웹문서는 **html**문서 형태로 보여지게 된다. 실험에서는 **html**태그의 대표적인 ***<title>***태그와 ***<body>***태그에 들어있는 ***<font size>***태그를 추출하였다. ***<meta>***태그와 ***<B>***, ***</>***와 같은 태그들은 제외를 시켰다.

3.3 차별화된 가중치의 부여

추출된 태그에 대한 가중치를 다음과 같이 부여하였다.

표 2 **html**태그별 가중치 우선순위

우선순위	**html** 태그
1	***<title>***
2	***<font size="7">***
3	***<font size="6">, <h1>***
4	***<font size="5">, <h2>***
5	***<font size="4">, <h3>***
6	***<font size="3">, <h4>***
7	***<font size="2">, <h5>***
8	***<font size="1">, <h6>***

3.4. 문서 가중치의 계산

문서의 가중치 계산은 아래에 있는 식으로 계산한다. 사용자가 검색한 단어가 어느 html 태그 중 어느 부분에 나타나는 지를 분석하여, 그 태그마다 차별화된 가중치를 부여하게 된다.

$$\text{modified\_TF-idf} = TF \times Tag\_weight \times \log\left(\frac{N}{n_i}\right)$$

4. 실험 및 평가

4.1 시스템의 구조

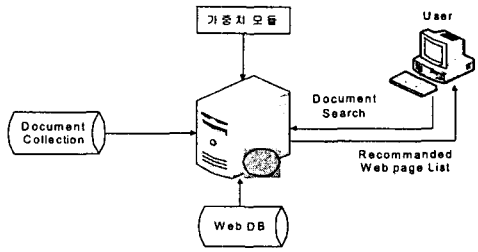

그림 3 시스템의 구조

그림3은 본 논문에서 사용한 시스템의 구조를 보여준다. 사용자는 웹서버에 접속하여 "document collection"이라는 곳에 저장된 **html** 문서들을 검색하게 된다. 이미 검색된 내용에 대한 추천리스트와 가중치들은 Web DB에 저장이 되어있어서, 다시 검색을 하지 않아도 된다. 만약 새로운 검색인 경우에는 가중치 모듈에 있는 가중치를 가져와서 **html**문서를 태그별로 분석하여 가중치를 계산하게 되고, 가중치가 높은 문서 순으로 사용자에게 **html**문서로 보여주게 된다.

4.2 샘플로 사용한 웹 페이지들

샘플로 사용한 웹 페이지들은 yahoo 사이트에서 컴퓨터와 관련된 카테고리로 검색한 문서들을 사용하였으며, 한 단어당 70~80여개의 문서들과 20여개 이상의 단어를 대상으로 실험을 했다.

4.3 실험 과정

실험에서 사용한 가중치는 표3과 같다. 기존의 TF-idf방식은 가중치를 1로 고정시켰으며, 나머지 (1)부터 (11)까지의 차별화된 가중치를 적용하였다. (여기서 font2는 font

size=2를 말한다.)

표 3 실험에서 사용한 가중치

	font2	font3	font4	font5	font6	font7	Title
TF-idf	1	1	1	1	1	1	1
(1) mod_TF-idf	1	2	2.2	2.4	2.6	2.8	10
(2) mod_TF-idf	1	2	2.5	3	3.3	3.8	10
(3) mod_TF-idf	1	2	3	4	4.4	4.7	10
(4) mod_TF-idf	1	2	3	4	5	6	10
(5) mod_TF-idf	1	3	3.5	4	4.5	5	10
(6) mod_TF-idf	1	3	4	4.5	5	6	10
(7) mod_TF-idf	1	3	4	5	6	7	10
(8) mod_TF-idf	1	4	4.5	5	5.5	6	10
(9) mod_TF-idf	1	4	5	6	7	8	10
(10) mod_TF-idf	1	5	6	7	8	9	10
(11) mod_TF-idf	2	3	4	5	6	7	10

기존의 TF-idf방식으로 계산된 가중치와 수정된 TF-idf 방식으로 계산된 가중치를 비교하기 위해 recall과 precision을 사용 하였으며, precision 수치는 상위 랭크 **N** 개의 문서들을 기준으로 측정하였고, 20여개 단어에 대한 평균 precision으로 계산되었다. 평균 precision에 대한 계산식은 아래와 같다.

$$avr\_prc=\frac{1}{m}\sum d_i\_prc$$

여기서, **N**은 10, 20, 30, 40을 사용하였고, 평균 precision을 ***avr_prc***로 정의했다. d_i***_prc***는 단어로 검색하여 추천된 페이지들에 대한 precision을 의미한다. ***m***은 검색한 단어의 수를 말한다.

4.4 실험결과

- Recall 측정결과

기존의 TF-idf방식과 수정된 TF-idf방식은 둘 다 단어기반의 검색을 원칙으로 하기 때문에, recall의 수치는 0.67정도로 같게 나왔다.

- Precision 측정결과

표 3은 상위 N개의 페이지들을 기준으로 기존의 TF-idf 방식과 수정된 TF-idf 방식에 의해 계산된 가중치를 보여준다. 기존의 TF-idf 방식에서의 precision은 0.5~0.6의 수치를 보였으나, 수정된 TF-idf방식을 사용했을 때의 precision은 0.7~0.95 정도로 보다 높은 수치가 나왔다. 따라서, 실험에서는 그림 4와 같이 (11) mod_TF-idf의 가중치를 적용한 것이 가장 좋은 결과를 나타냈다.

표 4 Top-N에 대한 평균 precision의 비교

	Top-10	Top-20	Top-30	Top-40
TF-idf	0.6	0.525	0.6	0.55
(1) mod_TF-idf	0.85	0.675	0.667	0.65
(2) mod_TF-idf	0.85	0.675	0.667	0.65
(3) mod_TF-idf	0.85	0.675	0.667	0.675
(4) mod_TF-idf	0.85	0.675	0.667	0.675
(5) mod_TF-idf	0.85	0.7	0.7	0.675
(6) mod_TF-idf	0.85	0.7	0.7	0.675
(7) mod_TF-idf	0.85	0.7	0.733	0.7
(8) mod_TF-idf	0.9	0.725	0.733	0.7
(9) mod_TF-idf	0.9	0.725	0.733	0.675
(10) mod_TF-idf	0.9	0.75	0.733	0.675
(11) mod_TF-idf	0.95	0.75	0.733	0.725

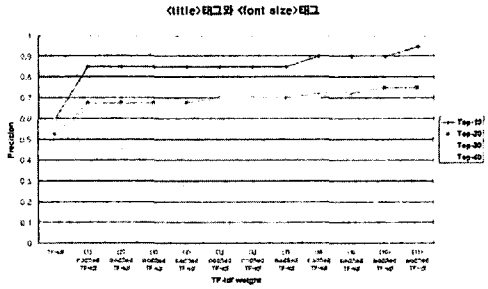

그림 4 Top-N에 대한 평균 precision의 비교

5. 결론

인터넷의 발달로 웹에서 얻을 수 있는 정보가 급진적으로 증가하고 있다. 사용자의 요구 또한 다양해지고 있다. 그에 따라 사용자가 원하는 웹문서를 찾아주는 검색분야는 더욱 중요해졌다.

본 논문에서는 수정된 TF-idf방식으로 단어기반의 검색에서 인덱스 가중치를 효과적으로 계산하는 방법을 제시하였다. 이 방식은 recall이란 측정은 기존의 방식과 같게 나올지라도 기존의 방식보다 precision을 높일 수 있는 방법이라고 할 수 있겠다.

참고문헌

[1] Salton, G. & McGill, M., Introduction to modern Information Retrieval, McGraw-Hill Press, New York, 1983.

[2] Ricardo Baeza-Yates, Berthier Ribeiro-Neto, Modern information retrieval, ACM press, 1999

[3] Mingjun Lan, Shui Yu, Ruth Backer, Walei Zhou., "A Co-Recommendation Algorithm for Web Searching.", Fifth International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'02), IEEE International Conference, 2002

[4] Nathaniel Good, J. Ben Schafer A. Konstan, Al Borchers., "Combining Collaborative Filtering with Personal Agents for Better Recommendations.". In Proceedings of AAAI, pp.439-446, AAAI Press, 1999.

[5] Dik L. Lee., "Document Ranking and the Vector-Space Model.", IEEE software, 14(2), 1997.

[6] Sergey Brin and Lawrence Page, "The Anatomy Of a Large-Scale Hypertextual Web Search Engine", International World Wide Web Conference Proceedings , pp.107-117, 1998

[7] Emmanouil Vozalis, Konstantinos G. Margarits., "Analysis of Recommender Systems' Algorithms.", presented at HERCMA, Athens, Greece, 2003

[8] Venkat N. Gudivada, Vijay V. Raghavan, William I. Grosky, Rajesh Kasanagottu, "Information Retrieval On The World Wide Web.", IEEE Internet Computing, pp.58-68, 1997

[9] Budi Yuwono, Dik L. Lee., "Search and Ranking Algorithms for Locating Resources on the World wide Web.", 12th International Conference on Data Engineering, pp.164-171, 1996