추가 수집 웹문서를 위한 페이지랭크 할당 모델

오은정[O] 강인호 김길창
한국과학기술원 전자전산학과 전산학전공
{ejoh,ihkang,gckim}@csone.kaist.ac.kr

PageRanking of Newly Crawled Web Documents

Eun-jung Oh[O] In-ho Kang Gil Chang Kim
Division of Computer Science, Dept. of EECS, KAIST

요 약

사용자가 얻고자 하는 정보를 인터넷에서 빠르고 정확하게 검색하는 것은 중요하다. 웹문서들 간의 상대적인 중요성을 나타내는 페이지랭크는 검색의 질을 높일 수 있어, 정보검색에 많이 이용된다. 인터넷상의 웹문서는 짧은 시간에 빠르게 증가하므로 새로운 문서들이 생성될 때마다 전체 문서의 페이지랭크를 계산하는 것은 많은 시간과 비용이 소모된다. 기존 웹문서의 페이지랭크는 변경하지 않고 추가된 웹문서들만으로 페이지랭크를 계산할 수 있다면 시간과 비용면에서 효율을 높일 수 있다. 본 논문에서는 추가되는 문서는 이전 문서의 페이지랭크에 많은 영향을 미치지 않는다는 점을 이용하여 추가되는 문서를 위한 페이지랭크를 할당 모델을 제시하고 평가한다.

1. 서론

'정보의 바다'라 일컬어지는 인터넷은 다양하고 많은 양의 정보가 존재하고, 수량 또한 빠르게 증가한다. 방대한 규모의 웹문서들 사이에서 얻고자 하는 정보를 정확하게 검색하는 것은 더욱 중요하다.

초기 검색 시스템들은 단순히 웹문서에 검색하는 검색어가 포함된 문서들을 사용자에게 보여주는 방식으로, 재현율(recall)을 높이기 위해 관련문서를 모두 찾는 것에 중점을 두었다[7]. 이 방식은 단순히 검색어가 포함된 모든 문서를 검색 결과로 제시하므로 사용자가 필요로 하지 않는 문서들을 많이 포함한다. 최근에는 내용 기반 검색과 웹 문서간의 링크 구조를 이용하여 검색하는 방식이 제안되어 사용되고 있다[5]. 이 방식은 정확률(precision)을 높이기 위해 관련성이 높고 권위있는 페이지를 먼저 제시하는 것에 중점을 두었다[4]. 웹문서 사이의 링크구조를 이용하여 문서들에 대한 값을 수치로 표현한 것이 페이지랭크(PageRank)이다. 구글[1] 검색기에서는 페이지랭크를 사용하여 많은 사람들이 추천하는 권위있는 문서를 상위 결과로 제시하여 좋은 성능을 보였다[5].

웹문서들은 일반적으로 짧은 시간에 생성, 변경되는 특징을 가진다. 이에 따라 최상의 결과를 제시하기 위해서는 새로운 문서의 수집과 수집된 문서에 대한 색인(indexing) 작업이 필요하다. 추가된 문서들이 기존 문서 집합에 미치는 영향을 최소화하여 독립적이고 개별적으로 색인 작업을 하면 전체적인 정보 변경을 최소화하여 비용을 절감할 수 있다. 문서들의 페이지랭크를 할당하는 것도 색인 작업 중 하나이다. 추가된 웹문서들의 페이지랭크를 계산하려면 전체 문서들 사이의 링크 구조가 필요하다. 즉, 페이지랭크는 모든 문서가 있어야 계산 가능하다. 그러나 웹문서의 방대한 양 때문에 모든 문서들의 링크 구조 수정 및 페이지랭크를 재계산하는 것은 많은 시간과 비용이 소모된다. 따라서 추가되는 문서 집합의 링크 구조만을 이용하여 페이지랭크를 할당하면 색인 작업의 효율을 높

[1] http://www.google.com

일 수 있다.

본 논문에서는 추가되는 문서에 대해 페이지랭크를 구하는 모델을 제시하고, 이에 대한 평가 방법을 제안한다. 페이지랭크는 문서간의 상대적인 중요성을 나타내므로 페이지랭크 자체보다는 문서들간의 순위가 중요하다. 본 논문에서는 웹문서에 적당한 페이지랭크를 할당하여 문서들간의 적절한 순위를 나타내는 것을 목적으로 한다.

본 논문에서 제시하는 모델에 대한 실험 및 평가는 다음과 같은 가정 하에 이루어진다.

가정1. 기존 문서의 페이지랭크는 미리 계산되어 저장된 상태이다.

가정2. 기존 문서 집합의 링크 구조를 알 수 없다.

가정3. 추가되는 문서의 양이 적을 경우, 기존 문서들의 페이지랭크에 큰 영향을 미치지 않아 기존 문서의 페이지랭크는 변함이 없다.

본 논문의 구조는 다음과 같다. 2장에서는 페이지랭크에 관한 관련 연구를 보이고, 3장에서는 점진적인 페이지랭크 할당 모델을 제시한다. 4장에서는 이에 대한 평가 방법을 제안하고 실험 결과에 대해 기술한다. 5장에서는 실험 결과를 논의하고, 6장에서 결론을 맺는다.

2. 관련연구

웹문서들은 서로 링크로 연결되어 있다는 점에서 일반적인 문서와는 다른 구조적 특징을 가진다. 이 장에서는 웹 문서의 구조적 특징을 이용한 관련연구를 보인다.

2.1. 권위자(Authority)와 허브(Hub)

웹문서의 구조적 특징인 링크 구조를 통해 어떤 문서가 권위적인(authoritative) 문서인지를 찾거나 유사한 문서들을 모으는 등의 정보를 알아낼 수 있다. 많은 허브가 가리키는 문서를 권위자(authority)라 하고, 많은 권위자 문서를 링크로 갖는 문서를 허브(hub)라 한다. 다음은 권위자와 허브 값을 구하는 식이다[1,4].

$$authority(p) = \sum_{\forall q, q \text{ points } p} hub(q) \quad \cdots\cdots\cdots \quad (1)$$

$$hub(p) = \sum_{\forall q, p \text{ points } q} authority(q) \quad \cdots\cdots\cdots \quad (2)$$

authority(p): 웹문서 *p*의 권위자 값
hub(p): 웹문서 *p*의 허브 값

권위자 값과 허브 값은 검색 결과 100-200개의 웹문서를 이용하여 계산되고, 질의어에 대한 인기도를 나타낸다[4]. 그러나 이 값은 검색할 때마다 계산해야 하므로 실질적으로 사용하기에는 많은 제약이 따른다.

2.2. 페이지랭크

웹문서들의 상대적인 중요성을 측정하기 위해 제안된 것이 페이지랭크이다. 권위자와 허브 값은 검색시 매번 계산하는 반면 페이지랭크는 색인 작업시 미리 계산하여 빠른 검색이 가능하다. 따라서 페이지랭크는 상업적인 정보 검색기에 더 많이 사용된다. 페이지랭크는 링크를 통해 연결되어 있는 웹문서들을 그래프 구조로 생각하고 웹문서들의 순위를 계산한 것이다. 많은 문서들이 가리키고 있는 문서가 더 중요하다고 생각하고 그것에 대한 값을 수치로 표현한다[5]. 페이지랭크를 계산하는 수식은 식 (3)과 같다[4].

$$PR(A) = (1-d) + d(PR(T_1)/C(T_1) + \ldots + PR(T_n)/C(T_n)) \quad \cdots\cdots\cdots \quad (3)$$

*PR(A)*는 문서 *A*의 페이지랭크, $PR(T_i)$는 문서 *A*를 가리키는 문서 T_i의 페이지랭크, $C(T_i)$는 문서 T_i가 가리키는 문서 개수, *d*는 사용자가 특정 문서에서 만족하지 못하고 다른 문서로 이동할 확률을 나타낸다.

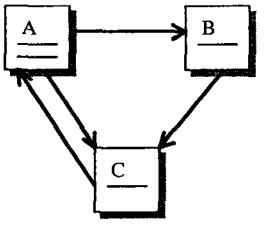


그림1. 웹 문서들의 링크 구조

그림 1과 같은 문서들 사이의 링크 구조에서 각 문서들의 페이지랭크는 *d*를 0.75라 가정하고 다음과 같은 방정식을 유도한다.

$$PR(A) = 0.25 + 0.75 \times PR(C)$$
$$PR(B) = 0.25 + 0.75 \times (PR(A)/2)$$
$$PR(C) = 0.25 + 0.75 \times (PR(A)/2 + PR(B))$$

전체 웹문서들의 페이지랭크를 구하기 위해 초기에 시작 값을 설정하고 반복적으로 페이지랭크

를 계산하는 방식을 적용한다. 웹문서의 페이지랭크는 반복 계산 과정을 통해 초기값에서 원래 페이지랭크의 근사치로 수렴한다[5].

3. 점진적 페이지랭크 할당

문서들의 상대적 중요성을 나타내는 페이지랭크는 검색에서 중요한 요소로 쓰인다. 문서들이 추가될 때마다 모든 문서들의 페이지랭크를 계산하면 많은 시간이 소비된다. 따라서 추가되는 문서들이 이전 문서들의 페이지랭크에 대해서 많은 영향을 미치지 않는다고 가정하고 추가되는 문서에 대해서만 페이지랭크를 계산하여 비용을 절감한다. 추가되는 문서의 양이 이전의 문서의 양에 비해 매우 적을 경우, 기존 문서의 페이지랭크의 변동이 작다고 가정하고 점진적인 페이지랭크 할당 (incremental PageRanking)이 가능하다.

3.1. 웹 문서들의 링크 구조

전체 웹문서는 이전에 이미 존재하던 문서(D_{old})와 새로 추가되는 문서(D_{new})로 나뉜다. 본 논문에서는 이전 문서들은 페이지랭크만 알 수 있고 문서들간의 링크 구조는 알 수 없다고 가정한다. 따라서 추가되는 문서 집합의 문서들 간의 링크 구조 정보만 가진다. 웹문서들의 구조는 그림 2와 같다. 추가되는 문서들 중에는 이전 문서들에 존재하던 문서가 있을 수 있다. 공통 부분의 문서는 추가된 문서가 이전에 존재하던 웹문서를 가리키거나, 기존의 문서 집합에서 링크만 가지고 있던 웹문서가 추가되면 생긴다. 이런 공통 부분의 문서들은 추가되는 문서의 페이지랭크를 계산하는데 이용된다.

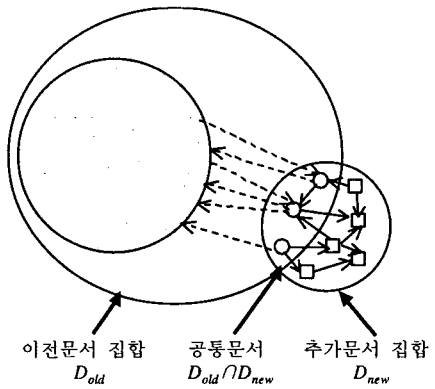


그림2. 웹 페이지의 구조

웹문서의 페이지랭크를 계산하기 위해서는 문서마다 그 문서가 가리키는 문서 개수를 알아야 한다. 그러나 이전 문서들은 가정에 의해 단순히 페이지랭크만 계산되어 있어 링크 구조를 알 수 없다. 공통 부분 웹문서($D_{old} \cap D_{new}$)가 가리키는 문서 개수 즉, 아웃링크(out-link) 개수는 이전 문서들과 새로 추가된 문서들의 크기에 비례한다고 가정하고 식 (4)와 같은 방식으로 계산한다. 그림 3은 공통 부분의 아웃링크 개수를 계산하는 구조를 나타낸다. 계산된 아웃링크 개수는 페이지랭크를 구하는데 사용한다.

$$Outlinks(p) \approx \left(\frac{n(D_{old})}{n(D_{new})} + 1\right) \times Newoutlinks(p) \quad \text{..... (4)}$$

$p : p \in (D_{old} \cap D_{new})$
$Outlinks(p)$: 문서 p가 가리키는 전체 문서 개수
$Newoutlinks(p)$: 문서 p가 가리키는 추가 문서 개수
$n(D_{old})$: 이전 웹문서 개수
$n(D_{new})$: 새로 추가된 웹문서 개수

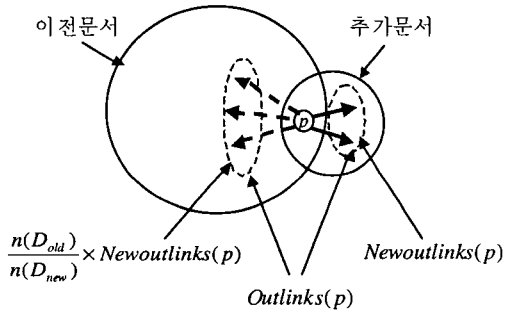


그림3. 공통 부분 문서 p의 $outlinks$개수

3.2. 점진적인 페이지랭크 계산

초기값은 추가되는 문서가 공통 부분의 문서인지에 따라 다르게 설정한다. 추가된 문서들 중 이전 문서들과 공통된 문서들에 대해서는 이전 문서의 페이지랭크를 이용하고 그 외의 문서들에 대해서는 동일한 값을 할당하여 전체 합이 1이 되도록 한다. 페이지랭크는 초기값을 어떤 값으로 설정하느냐에 상관없이 하나의 페이지랭크에 수렴한다. 수렴하기까지의 반복 횟수에 차이가 있을 뿐이다[7]. 보다 빠른 수렴을 위해 위와 같이 초기값을 설정한다.

본 논문에서는 공통된 부분의 문서들은 추가된 문서들의 페이지랭크를 계산하는데 영향을 주지만

그 값의 변화는 없다고 가정한다. 따라서 공통 부분의 문서들은 추가된 문서들의 페이지랭크를 구하는 과정을 통해 변경되지만 문서가 추가될 때 반영하지 않는다. 설정된 초기값과 아웃링크 개수로 페이지랭크를 구하는 식을 적용하며, 페이지랭크가 수렴할 때까지 반복하여 계산한다. 그림 4는 점진적인 페이지랭크 할당 알고리즘을 나타낸다. PR_{prev}는 반복 과정에서 문서의 이전 단계의 페이지랭크를 나타내고 c는 상수이다.

1. 문서 p의 초기값설정
   if($p \in D_{old} \cap D_{new}$)
   $PR(p)$ = 이전 페이지랭크
   else
   $PR(p) = 1/\alpha$ ($\Sigma PR(p)=1$ 성립하도록 α 설정)

2. 문서 p의 아웃링크 개수 계산
   if($p \in D_{old} \cap D_{new}$)
   $Outlinks(p)$ = 추정한 아웃링크 개수 (식4)
   else
   $Outlinks(p)$ = 문서의 아웃링크 개수

3. 추가 문서 집합의 페이지랭크 계산(식3 : $d=0.75$)
   if($p \in D_{old} \cap D_{new}$)
   변경된 문서 p의 값은 이전 페이지랭크로 설정

4. if($|\sum PR(p) - \sum PR_{prev}(p)| >= c$)
   go to 3.
   else
   exit

그림4. 점진적인 페이지랭크 할당 알고리즘

3.3. 페이지랭크의 정규화

이전의 모든 웹문서들의 페이지랭크의 합을 1으로 정규화한다. 그러나 문서들이 추가됨에 따라 계산상에서 이전 문서의 페이지랭크의 합은 작아진다. 따라서 이전 문서들의 페이지랭크의 합을 1이 아닌 $\lambda(0<\lambda<1)$로 하고, 추가된 문서들의 페이지랭크의 합은 $1-\lambda$로 본다. 전체 문서들을 한꺼번에 페이지랭크를 구했을 때와 같이 모든 문서들의 페이지랭크의 합은 1이 된다. 기존 문서 집합의 페이지랭크 합이 1이므로 λ에 대한 보정이 필요하다. 그러나 공통 부분 문서들의 페이지랭크는 그대로 사용하였기 때문에 λ값에 대한 추가 문서 집합의 보정은 무시한다. 그림 5는 점진적인 페이지랭크 할당 모델에서 문서들의 페이지랭크 합을 나타낸다. 그러나 이전 문서들의 페이지랭크는 사실상 변화가 없다고 가정하였기 때문에 모든 문서의

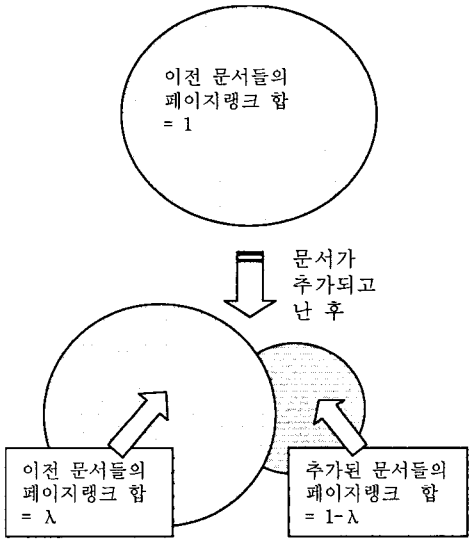


그림5. 점진적인 페이지랭크 할당 방법에서 문서들의 페이지랭크 합

페이지랭크를 더하면 $2-\lambda$가 된다.

가정에서도 언급했듯이 추가되는 문서들만 페이지랭크를 계산하고 이전 문서들은 페이지랭크를 변경하지 않는다. 즉, 점진적인 페이지랭크 할당 모델은 페이지랭크를 구하는 비용을 절감하고, 문서들의 페이지랭크의 순서를 유지하는 것이 목적이다. 그러나 구해진 페이지랭크는 정확한 값이 아니므로 정기적으로 모든 문서들에 대해 정확한 페이지랭크를 계산하는 것이 필요하다. 따라서 이와 같이 페이지랭크의 합이 1이 되지 않는 것은 무시할 수 있다.

4. 실험 및 평가

4.1. 데이터

실험 데이터로 TREC-9 2001WebTrack의 WT10g을 사용하였다[2,3]. WT10g는 전체 1,692,096개의 문서를 가진다. 이 데이터는 웹 스파이더(Web Spider)에 의해 추가된 문서의 유형별로 5 가지 경우에 대해 실험하였다. 전체 문서의 5%, 2%, 1% 에 해당하는 문서들이 추가된다 설정하고, 적절한 λ를 찾기 위해 λ를 변화시켜 실험하였다.

4.2. 추가되는 문서 유형

웹 스파이더는 웹문서들의 링크를 따라가면서

문서들을 수집한다. 웹 스파이더가 문서들을 가져오는 전략(Strategy)에 따라 위와 같이 다섯 가지로 구분한다.

(1) Random : 임의적으로 선택한 경우
(2) PathDepth : URL의 사선(/) 개수에 따라 선택한 경우
(3) Inlink :인링크(in-link)[2] 개수에 따라 선택한 경우
(4) Outlink : 아웃링크 개수에 따라 선택한 경우
(5) PageRank : 페이지랭크에 따라 선택한 경우

4.3. 평가 방법

추가되는 문서의 페이지랭크는 정확한 문서의 페이지랭크를 구하기보다는 순서를 유지하는 것을 목적으로 한다. 웹문서들의 상대적인 중요성은 페이지랭크 자체보다는 그것이 나열된 순위가 중요하기 때문이다. 따라서 점진적인 페이지랭크 할당으로 구해진 페이지랭크가 적절하게 구해졌는가를 평가하기 위한 새로운 평가 방법을 제안한다.

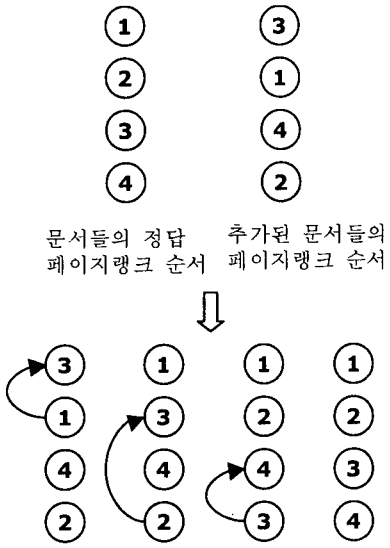
이 평가 방법은 문서들의 정답 페이지랭크의 순위와 추가된 문서들의 페이지랭크의 순위를 비교하여 원래 정답 페이지랭크 순위가 되는데 필요한 교환 연산의 거리(순위차)를 더한다. 모든 거리의 합을 전체 문서들의 수로 나누어 평균 교환 연산 거리를 평가하는 값으로 사용한다. 이 값을 문서당 재정렬 비용(reordering cost)라고 정의한다.

문서당 재정렬 비용이 작을수록 더 좋은 결과이다. 그림 6은 이런 방식으로 계산되는 평가 예를 보인다. 위 예에서는 교환 연산의 거리의 합은 4이고 전체 문서들의 숫자가 4이므로 평균 거리는 1이다.

전체 문서의 정답 페이지랭크는 전체 WT10g의 데이터를 기존의 페이지랭크 계산하는 방법으로 계산한 것이다. 본 논문에서는 이 방법을 독립적(stand-alone) 방식이라고 정의한다. 이전 문서의 페이지랭크 역시 독립적 방식으로 계산한다. 추가된 문서의 페이지랭크를 점진적인 페이지랭크 할당 모델로 계산하여 이전 문서들의 결과에 추가한 것이 최종 결과이다. 추가되는 문서는 이전에 나타나지 않았던 문서들의 페이지랭크만 추가되는 것으로 한다. 이 두 결과를 제시한 평가 방법을 통해 문서당 재정렬 비용을 계산한다.

4.4. 실험 결과

실험은 추가되는 문서 유형에 따라 실시하였다. 추가되는 문서의 양을 다르게 하여 실험하였고, 페이지랭크를 구하는데 영향 요소가 될 수 있는 λ 값을 변화시키면서 실험하였다. 실험 결과는 추가되는 문서들만 독립적 방식으로 구한 결과와 비교한다.



재정렬 비용 : (1+2+1)/4 = 1

그림6. 평가 결과 계산 과정

표 2는 추가되는 웹문서들의 유형별 평가 결과를 나타낸다. 점진적인 페이지랭크 할당 방식이 독립적 방식보다 추가되는 문서의 유형에 상관없이 더 좋은 결과를 보인다.

표2. 추가된 문서 유형별 문서당 재정렬 비용

페이지 유형		문서당 재정렬 비용	stand-alone 문서당 재정렬비용
Random	-	85,741.006	777,847.532
Path-Depth	큰것	28,716.760	763,697.770
	작은것	91,608.734	757,849.132
In-link	큰것	394,774.654	646,182.888
	작은것	15,584.012	683,667.826
Out-link	큰것	297,536.797	706,393.980
	작은것	137,262.597	776,968.115
Page-Rank	큰것	458,068.599	595,824.626
	작은것	15,518.205	699,751.220

(추가되는 문서 = 5%)

[2] 웹문서 p가 있을 때, p를 가리키는 웹문서의 개수

결과를 통해 페이지랭크가 작은 것이 추가되는 것이 가장 이상적인 경우임을 알 수 있다. 이 결과는 인링크 개수가 작은 것이 추가되는 것과 비슷한 결과를 보인다. 이것은 평균적으로 인링크 개수가 많을수록 페이지랭크가 커지는 경향이 있기 때문이다. URL의 사선(/) 개수의 경우 사선 개수가 적은 것이 추가되었을 경우가 큰 것이 추가되었을 경우보다 더 좋은 결과를 보인다.

표 3에서는 추가되는 웹문서의 양에 따른 실험 결과를 보인다. 추가되는 웹문서의 수가 적을수록 더 좋은 결과를 보인다. 그러나 임의 선택하여 추가하였을 경우 반대의 경우를 보였다.

표 4는 페이지랭크에 영향을 미치는 추가적인 요인인 λ값에 따른 문서당 재정렬 비용을 나타낸다. 이 표를 통해 (1-λ)값이 추가되는 문서의 정답 페이지랭크의 합과 비슷할수록 더 좋은 평가 결과를 냄을 알 수 있다.

표3. 추가된 문서 양별 문서당 재정렬 비용

페이지 유형	페이지 양	문서당 재정렬 비용
Random	5%	85,714.006
	2%	176,055.373
	1%	201,297.438
PageRank 큰 것	5%	461,315.905
	2%	333,280.281
	1%	254,806.996
PageRank 작은 것	5%	15,577.484
	2%	4,637.027
	1%	2,242.264

(λ =0.998)

5. 토론

본 논문에서는 공통 부분 문서의 페이지랭크가 변함이 없다고 가정하였다. 페이지랭크 순서대로 95, 98, 99%를 기존 문서로 사용할 경우 공통 문서의 평균 페이지랭크 변화율은 그림 7과 같다. 추가되는 문서 집합의 크기에 따라서 페이지랭크의 변화율이 감소하는 것을 알 수 있다. 99%를 기존 문서 집합으로 하였을 경우 3%의 변화율을 가졌다. 따라서 실험 데이터의 크기가 클 경우, 본 논문에서의 가정은 크게 위배되지 않는다고 볼 수 있다.

추가된 문서의 유형에는 다섯 가지 경우를 생각한다. 표 2에서 나타난 문서당 재정렬 비용은 임의로 선택한 문서들이 추가된 경우에 대해

표4. λ값에 따른 문서당 재정렬 비용

페이지 유형	λ값	문서당 재정렬 비용	정답 페이지랭크 합
Random	0.990	223,677.615	0.0021686
	0.995	223,508.852	
	0.998	176,055.373	
Page-Rank 큰것	0.990	335,419.361	0.0063911
	0.995	4,637.027	
	0.998	4,870.534	
Page-Rank 작은것	0.990	228,509.463	0.0064769
	0.995	10,199.025	
	0.998	13,800.163	

(추가되는 문서 =2%)

비교하였을 때 URL의 사선(/) 개수가 큰 문서, 인링크 개수가 작은 문서, 페이지랭크가 작은 문서들을 추가하면 더 좋은 결과를 보임을 나타낸다. 추가되는 문서들이 이전 문서들의 페이지랭크에 많은 영향을 미치지 않는 것이 더 좋은 결과를 나타내었다. 이전의 문서들보다 상대적으로 덜 중요한 문서들이 추가된다는 것은 페이지랭크가 작은 것이 추가되는 것과 같다. 아웃링크개수는 그 문서의 상대적 중요성에 많은 영향을 미치지 못한다.

표 3에서 확인할 수 있듯이 일반적인 경우 추가되는 문서의 개수가 적을수록 문서당 재정렬 비용이 더 적게 나온다. 이것은 추가되는 문서의 양에 따라 문서당 재정렬 비용을 예측하는데 도움이 됨을 의미한다. 그러나 임의적으로 추가되는 문서를 선택하였을 경우 추가되는 문서의 양이 줄어듦에도 불구하고 문서당 재정렬 비용은 커진다. 이것은 추가되는 문서들의 페이지랭크의 편차가 크기 때문이다.

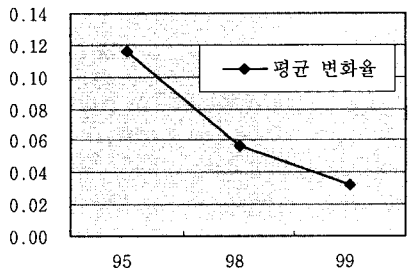


그림 7. 공통 문서 페이지랭크 평균 변화율

표 5는 페이지랭크가 작은 문서들이 추가되는 경우와 임의 선택한 문서들이 추가되는 경우의 정답 페이지랭크의 평균과 표준편차를 나타낸다. 페이지랭크가 작은 문서들의 평균과 표준편차가 임의 선택한 문서들의 평균과 표준편차보다 작다. 즉, 임의 선택한 문서들의 페이지랭크가 일반적으로 크고, 페이지랭크의 분포도 더 넓어 페이지랭크가 큰 문서가 포함될 가능성이 크다. 이런 이유로 임의로 선택한 문서들을 추가하였을 경우에 문서당 재정렬 비용이 더 크게 나타났다.

표5. 정답 페이지랭크의 평균과 표준편차

페이지 유형	페이지 양	평 균 ($x 10^{-7}$)	표준편차 ($x 10^{-7}$)
Random	5%	2.3732	20.5553
	2%	2.2079	2.3902
	1%	2.4278	19.6156
PageRank 작은 것	5%	1.7403	0.3516
	2%	1.7329	0.2149
	1%	1.7307	0.2072

표 4에서 알 수 있듯이 λ값은 추가되는 문서들의 합을 결정하는 요인이다. 따라서 이 값이 어떻게 설정되느냐에 따라 페이지랭크에 영향을 줄 수 있다. λ값을 너무 작게 설정하면 추가되는 문서들의 합은 커진다. 만약 추가되는 문서들의 페이지랭크가 작은 것들이 대부분이라면 추가되는 페이지랭크가 (1- λ)값으로 정규화되어 원래 문서의 페이지랭크보다 과대 평가 된다. 이 경우 원래 문서의 순위보다 순위가 많이 올라간다. 반대의 경우도 마찬가지이다. 따라서 λ값을 적절하게 설정해 주어야 추가된 문서들의 페이지랭크의 순위 변동에 많은 영향을 미치지 않는다.

결론적으로 웹 스파이더가 페이지랭크 값이 작은 문서들을 추가할 수 있으면 점진적인 페이지랭크 할당 모델이 좋은 결과를 나타낼 것이다. 따라서 웹 스파이더가 이전의 웹문서들의 페이지랭크를 알고 있다는 것을 이용하여 문서들을 페이지랭크가 큰 순서대로 가져오는 방법을 적용할 수 있다. 그런 방식으로 추가된 문서에 대해 점진적인 페이지랭크 할당 모델을 적용하여 페이지랭크를 계산할 수 있다.

6. 결론

본 논문에서는 검색기에 추가되는 웹문서들의 페이지랭크 할당 모델을 제시하였다. 페이지랭크를 추가되는 문서가 있을 때마다 새로 구하는 것은 많은 비용이 소모된다. 그러므로 새로 추가되는 문서들의 페이지랭크만 계산하여 웹문서들의 결과에 더해주면 많은 비용을 절감할 수 있다. 따라서 페이지랭크가 존재하는 이전 문서들과 공통 부분이 되는 문서들을 이용하여 새로 추가되는 문서들의 페이지랭크를 구하는 방식을 제안하고, 점진적인 페이지랭크 할당이라 정의한다.

점진적인 페이지랭크 할당 방식을 적용하여 추가되는 문서들의 페이지랭크를 계산한 값은 추가되는 부분만 독립적 방식으로 계산하여 추가하는 것보다 더 좋은 결과를 보인다. 또한 페이지랭크가 작은 것이 추가되었을 경우 이 문서들은 이전 문서의 페이지랭크에 많은 영향을 미치지 않아 더 좋은 결과를 나타냄을 확인하였다.

향후 연구로 웹 스파이더가 문서를 가져오는데 가져오는 문서의 페이지랭크가 작은 문서를 가져오기 위해 필요한 요소에 대한 연구가 필요하다.

7. 참고문헌

[1] Chris Ding, Hongyuan Zha, Ziaofeng He, Perry Husbands, Horst Simon, "*Analysis of Hubs and Authorities on the Web*" Lawrence Berkeley National Laboratory report LBNL-47847, 2001.

[2] D.Hawking, N. Craswell, CSIRO, "*Overview of the TREC-2001 Web Track*" NIST special publication 500-250 TREC 2001: 61-67, 2001.

[3] Peter Bailey, Nick Craswell, David Hawking, "Engineering a multi-purpose test collection for Web retrieval experiments", Information Processing and Management, 2001.

[4] Jon M. Kleinberg , "*Authoritative Source in a Hyperlinked Environment*" In Proceedings of the Nineth Annual ACM-SIAM Symposium on Discrete Algorithms: 25-27, 1998.

[5] Larry Page, Sergey Brin, R. Motwani, T. Winograd, "*The PageRank Citation Ranking : Bringing Order to the Web*" Stanford Digital Library Technologies Project, 1998

[6] Sergey Brin, Lawrence Page, " *The Anatomy of a Large-Scale Hypertextual Web Search Engine*" Computer Networks and ISDN Systems, 30(1-7):107-117, 1998.

[7] G.Salton and M.McGill, "*Introduction to Modern Information Retrieval*", McGraw-Hill,New York, NY, 1983.