제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003. 5)

위치 정보를 이용한 확장 벡터 모델의 문서 길의 정규화에 관한 연구

김광영*, 서정현*, 이민호*, 주원균*, 정창후*, 류범종*
*한국과학기술정보 연구원
e-mail: kykim, jerryseo, cokelee, joo, chjeong, ybj@kisti.re.kr

A Study on the Document Length Normalization of Extended Vector Model Using the Information of Location

KwangYoung Kim*, Jerry Seo*, MinHo Lee*, WonKyun Joo*, ChangHoo Jeong*, BeomJong You*
*Group for Intelligent Information System, Korea Institute of Science and Technology Information

요 약

인터넷의 발달과 인터넷 이용자수의 급격한 증가로 정보 검색 시스템의 필요성이 커지고 있다. 또한 대용량의 문서에서 사용자가 원하는 정보를 정확하게 찾기가 점점 어려워지고 있다. 현재 대부분의 검색 시스템들은 문서 길이에 대한 정규화를 처리하고 있다. 현재 문서 길이 정보도 검색 시스템의 검색 성능에 기여를 하고 있다. 일반적으로 TREC이나 HANTEC2.0을 이용한 검색 성능 평가를 했을 때 문서 길의 정규화를 하지 않는 것보다 한 것이 우수한 성능을 보여 주고 있다. 본 논문에서는 KISTAL2000을 이용하여 위치 정보를 사용하여 문서 길의 정규화 방법에 제시하고 이에 대한 실험하였다.

1. 서론

오늘날의 인터넷과 네트워크는 거대한 정보의 집합체로 바뀌고 있다. 널리 확산된 각 종 IT 인프라를 통해 웹, DB, 비정형문서 등 매년 새로이 생성되는 데이터는 전 세계적으로 1~2 exabyte (1exabyte = 10^{18})에 이르고, 인터넷 정보가 급증하고 있다, 만약 효율적인 Access 방법을 제공해 주는 적절한 검색 엔진이 없다면 정보의 생산, 유통, 소비에 이르는 정보 사이클 자체가 불가능 하다. 또한 인터넷 사용자들은 원하는 정보를 정확하게 찾기가 점점 어려워질 것이다. 또한 검색 대상 문서의 수가 급격히 증가함에 따라 검색 결과 또한 상한단 양으로 사용자가 원하는 정보인지를 쉽게 판단하고 확인하기가 어렵다.

현재 대부분의 검색 시스템에서는 문서 길이에 대한 정규화를 처리한다. 문서 길이에 대한 정규화를 처리함으로써 정보 검색 시스템의 성능 향상에 많은 영향을 주고 있다. 일반적인 검색 시스템에서 사용되는 Vector모델에서는 문서 길이에 대한 정규화는 없지만 포아송 모델, 피벗 단일 정규화 모델, 추론 네트워크 모델 등에서 이미 문서 길이에 대한 정보를 이용하여 문서의 가중치 계산의 한 요소로서 사용되고 이다. 문서 길이 정보도 검색 성능 평가에 중요한 요소로 작용하고 있음을 알 수 있다.

KRISTAL2000 IRMS 검색 시스템은 SumTF을 이용하여 문서 길이에 대한 정규화를 처리하고

있다. 하지만 문서 길이에 대한 정규화를 처리하기 위해서 먼저 모든 문서에 대한 가중치를 계산 후 문서 길이에 대한 정규화를 처리하는 방식으로 사용함으로써 검색 속도는 문서 길이에 대한 정규화를 하지 않은 것이 더 빠르다. 일반적으로 정보 검색 시스템들은 검색 성능과 검색 속도를 동시에 고려하여야 한다.

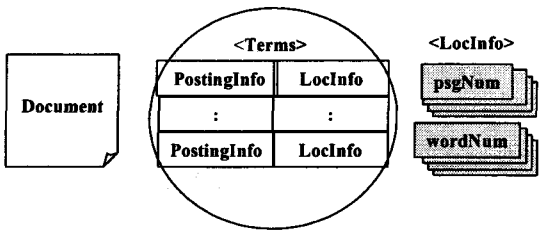
또한 KRISTAL2000은 검색 성능 향상을 위해서 사용자가 입력하는 어절에 대해서 이웃한 어절을 검색 처리를 중점으로 두고 검색을 처리하고 있다. 일반적인 벡터 가중치와 이웃한 어절의 가중치를 추가함으로써 벡터 가중치만을 사용하는 것보다 성능이 향상됨을 알 수 있었다.[1] KRISTAL2000 시스템에서는 한국어에 대해서는 근접도 연산자 Within[1]을 사용하며 영어권 언어에 대해서는 Near[2] 연산자를 사용한다.

KRISTAL2000에서는 어절의 위치 정보를 이용한 문서 길이에 대한 정규화 처리함으로서 검색 성능과 속도를 고찰하였다.

2. 이웃한 어절의 위치 Posting File 구조

문서 내에서의 색인어의 위치는 색인어의 출현 빈도와 함께 검색의 정확도를 높이는 중요한 요소로 작용할 수 있다.[1] 특히, 구절 검색과 같이 정확한 어구(Phrase)를 검색하고자 할 때는 색인어의 위치 정보가 빈도보다 더 중요하다. KRISTAL2000에서는 색인어의 문서 내 위치 정보를 포스팅 파일로 구성하여 색인어의 위치 정보를 검색에 이용하고자 한다. [1] 또한 위치 정보를 이용하여 문서 길이에 대한 정규화를 동시에 처리함으로써 검색 속도를 개선을 고려하였다.

[그림 1]에서와 같이 문서에서 색인 처리하여 Term마다 포스팅 정보와 LocInfo[3]인 위치 정보를 가지고 있다. 이 위치 정보(LocInfo)를 이용하여 Term과 Term사이의 위치 정보를 계산을 한다. 계산을 하면서 가장 긴 위치 정보(WordNum)[4]를 계속 유지를 한다. 이 정보를 이용하여 문서 길의 정규화를 처리한다.

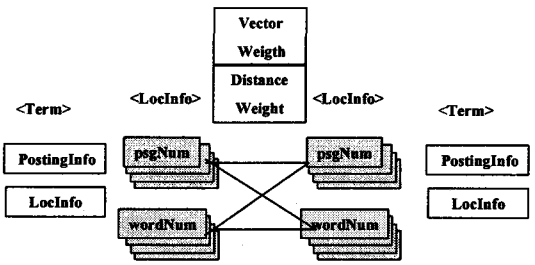


[그림 1] 포스팅 정보와 위치 정보

본 논문에서는 [그림 2]와 같이 포스팅 파일 위치 정보를 이용하여 색인어 간의 위치 정보를 계산하여 Vector모델로 계산된 가중치에 추가하는 방법을 사용하였다.

실제 (Vector Weight + Distance Weight) / 문서 길이에 대한 정규화 값으로 처리 한다

Vector Weight는 벡터 모델로 Ranking된 Similarity이고, Distance Weight는 어절간의 위치에 따른 Weight 값이다.



[그림 2] 근접도 연산처리

3. 위치 정보를 이용한 문서 길의 정규화 실험

본 논문에서는 KRISTAL2000에서 사용되는 문서 길이에 대한 정규화 방법을 [표-1]과 같은

1) Term들이 순서 되로 나열 된 것
2) Term들이 순서에 상관없이 나열 된 것
3) Term들의 위치 정보
4) 질의어 중에서 LocInfo 중에서 가장 큰 값

식을 사용하여 실험하였다.

[표 1] 문서 길의 정규화 계산식

식1. Weight Doc_j = $Weight_{Vector}(Doc_j)$

식2. Weight Doc_j = $Weight_{Vector}(Doc_j)$/ $\log(Max(WordNum_j)+1.0)$

식3. Weight Doc_j = $Weight_{Vector}(Doc_j)$/ $\log(SumTF+1.0)$

$MaxWordNum_j$ = $Max(WordNum_j)$

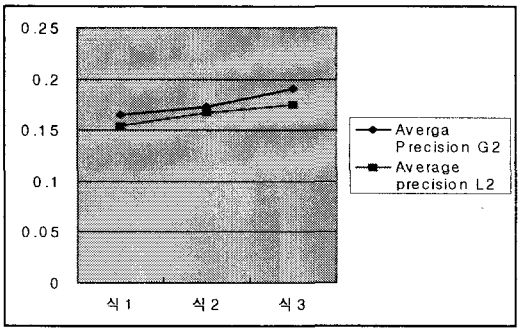
Weight Doc_j는 Term에 대한 가중치 계산인 Vector 모델로 가중치를 계산 처리한 것이다. 식3은 벡터 가중치에 문서 길이 SumTF를 이용하여 정규화를 처리한 것이다. 식2는 문서 길이 SumTF대신 Max(WordNum)을 이용하여 문서 길이에 대한 정규화를 처리하였다. 식1은 문서의 정규화를 처리하지 않았을 때 비교하기 위한 식이다. Max(WordNum)는 질의어들 중에서 가장 문서 끝에 있는 단어의 위치를 나타내는 것이다. 이것을 이용한 문서 길이 정규화를 했을 때 얼마의 성능이 향상될 것인가를 실험을 통하여 고찰하였다.

4. 실험

본 논문에서는 [표-1]을 이용하여 가중치를 실험을 HANTEC V2.0을 이용하여 실험을 하였다. 이 컬렉션은 120,000 건의 문서와 TREC 형식의 50개 질의, 그리고 각 질의에 대해 적합성 정도에 의한 8종류의 적합문서 집합으로 구성되어 있다. TREC 형식의 질의에서 <quer> 필드만을 사용하여 성능 평가 하였다. 평가는 G2.rel, L2.rel를 이용하여 50개의 질의에 대한 Average Precision값을 구하였다.

[표-2] G2, L2 Average Precision

<quer>	G2	L2	Retrieval time(sec)
식1	0.1652	0.1546	1:10
식2	0.173	0.1673	1:22
식3	0.1908	0.1749	3:33



[그림 3] G2, L2에 대한 식1,2,3 비교

실험 결과 [표-2]와 [그림 3]에서 가장 우수한 것은 SumTF를 이용하여 KRISAL2000의 문서 길이를 정규화한 식3이다. 그러나 KRISTAL2000 시스템에서는 문서 길이 정보 SumTF를 이용하지 않고 Max(WordNum)를 이용한 문서 길이 정규화를 처리해도 정규화를 하지 않은 것보다 성능이 향상됨을 알 수가 있었다. 실제 검색 시간은 Max(WordNum)가 SumTF보다 더 빠른 검색 성능을 보여 주었다. 시스템에 따라서 SumTF, AverageTF, uniqueTF를 어떻게 포스팅 파일을 구성하는지에 따라서 검색 시간은 상관성이 없을 수 있다. 하지만 현재 KRISTAL2000에서는 검색 성능과 검색 속도를 고려하면 Max(WordNum)를 이용하는 것이 검색 성능과 속도 측면에서 우수함을 볼 수 있었다.

5. 결론

본 논문에서는 문서 길이에 대한 정규화 방식을

적용하여 검색 성능을 향상시키기 위해서 다양한 방식 적용하여 실험을 하였다. 실험 결과에서 볼 수 있듯이 식3에 서 가장 높은 값을 보였다. 그러나 검색 시간을 고려한 성능 면에서는 Max(WordNum)를 사용하는 것이 문서 길이에 대한 정규화를 하지 않은 것 보다 더 향상됨을 볼 수가 있다.

참고문헌

[1] 김광영, 서정현, 최성필, "이웃한 어절의 위치 정보를 이용한 KRISTAL2000 검색 성능 향상", 정보과학회 2001년 10월 p121~123

[2] 이석훈, 맹성현, 김지영 "정보 검색 평가체제 구축을 위한 HANTEC 테스트 컬렉션의 패키징" KOSTI2000 p31~48

[3] 이준호, 최광남, 한현숙, "정보 검색 연구를 위한 KRIST 테스트 컬렉션의 개발" 정보관리학회지, 제2권 제2호 pp225-232, 1995

[4] Amit Singhal, Chris Buckley, Mandar Mitra "Pivoted Document Length Normalization"(1996)