가중치에 따른 질의확장의 검색효율성

Retrieval Effectiveness of Query Expansion depending on Term Weights

최성환, 전주기전여자대학

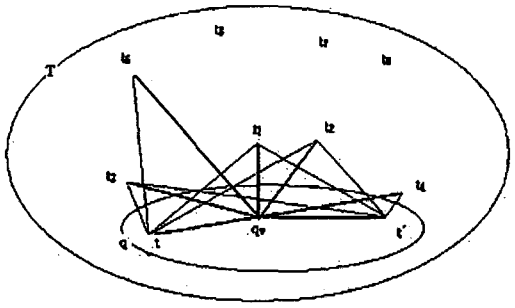
Choi Sung-Hwan, Jeonju Kijeon Women's College

기존의 질의확장 혹은 적합성 피드백 연구에서 코사인 정규화를 사용하여 검색성능을 향상시킨 연구들이 많다. 본 논문에서 실험한 결과를 근거로 하였을 때 이는 낮은 검색성능을 보였던 것이 검색공간의 확장으로 성능이 크게 향상되었을 가능성이 있다. 실험결과 가중치 유사도 모델간의 커다란 차이는 보이지 않고 코사인정규화 가중치 알고리즘에서 상당한 성능향상이 있었다. 그러나 기존의 코사인정규화 가중치 알고리즘을 이용한 전역적 질의확장의 경우 성능 향상률은 높으나 원질의어를 이용하여 가장 좋은 성능을 보였던 가중치 알고리즘들의 검색성능과 비교하면 오히려 낮은 성능을 보였다.

1. 서론

Qiu & Frei(1993)는 공기기반 유사도 시소러스를 이용하여 질의를 확장하는 연구에서 이전의 공기기반 질의확장의 실패요인은 1) 너무 높은 임계치 설정, 2) 전체 질의 개념과 확장 용어간의 유사도를 간과하고 각각의 질의 용어만을 고려했기 때문이라고 지적했다. 이러한 공기기반 유사도 측정은 두 개의 용어간의 유사도 모델로 Qiu & Frei(1993)가 지적한 두 번째 문제점을 해결하기 위해서는 여러개의 용어로 이루어진 질의어들과 확장하려는 한 개 용어간의 유사도 측정방법이 필요하다.

질의-용어간 유사도의 기본전제는 여러개의 용어로 구성된 질의어에서 단일어와 유사한 것보다는 질의 전체 즉, 질의개념과 유사한 용어를 고려하는 것이 정보요구에 보다 합리적이라는 것이다. <그림 1>의 문헌벡터공간에서 색인어집단을 T, 두 개의 용어 (t, t')를 포함하는 이용자 질의를 q라고 하고, 가상 용어 q_v는 전체 질의개념을 표현한다고 하자. 두 개의 용어간 거리가 가까우면 가까울수록 두 개의 용어는 더 유사한 관계를 가진다고 할 때, 용어 t_3은 다른 어떤 용어보다 t와 유사하고, t_4는 t'와 유사하다. 질의 q의 개념은 적절한 정보 구조를 이용하거나 질의 q의 센트로이드(centroid)를 계산함으로서 얻어질 수 있다. 만약에 확장할 용어수가 2개라고 할 때 우리는 문헌벡터공간에서 어떤 용어를 선택해야 하는가? 질의를 구성하는 각각의 용어 (t, t')를 고려했을때는 t_3과 t_4를 선택하는 것이 당연하다. 그러나 질의 개념 q_v는 질의를 구성하는 각각의 용어들이 표현하고자 하는 하나의 주제를 표현하기 때문에 질의확장시 q_v와 가까운 t_1과 t_2를 선택하는 것이 보다 타당한 논리적 근거를 얻을 것이다.



<그림 1> 문헌벡터공간에서 질의-용어간의 관계

본 논문에서는 Qiu & Frei(1993)가 제시한 질의전체 개념을 고려하여 질의확장시 가중치 특성에 따라 검색효율성에 어떠한 영향을 미치는지를 분석하였다.

2. 실험설계

2.1 실험개요

본 논문에서 사용한 실험시스템은 펜티엄 III에 리눅스 운영체제를 기반으로 한다. 데이터베이스로는 Mysql를 이용하고, 웹프로그래밍 언어인 PHP로 웹상에서 실시간 검색이 가능하도록 구현되었다. 구현된 실험시스템은 서지사항인 제목, 초록을 제시하고, 검색된 문헌의 제목을 클릭하면 직접 원문에 접근이 가능하다.

검색모형으로는 수식(1)과 같은 내적유사도를 기반으로 한 벡터공간모델을 채택하였고 질의확장시 가중치 기법에 따른 검색효율성을 평가하기 위해 다양한 가중치 알고리즘을 적용하였다.

$$Sim(d_j, q_k) = \sum_{i=1}^{n} (td_{ij} \times tq_{ik}) \qquad (1)$$

문헌 색인은 한글 실험집단의 경우 형태소 분석기 HAM으로 색인 하였고, 영문 실험집단은 HAM과 Porter 스테밍 알고리즘으로 색인 하였다. 질의어 처리도 문헌 색인 방식과 동일하게 자동으로 처리하였다. 벡터공간모델에서는 문헌과 질의가 모두 벡터 형태로 표현된다. 문헌 d_j의 용어 가중치 td_{ij}는 각 가중치 알고리즘 공식에 따라 계산되고 특정 문헌에 나타나지 않는 색인어들에 대한 가중치는 0이 할당된다. 본 논문에서 질의어 가중치 tq_{ik}는 $\langle \log tf + 1 \rangle$을 공통적으로 사용하였다.

2.2 실험집단

질의확장시 용어가중치 기법이 정보검색의 효율성에 미치는 영향을 평가하기 위해 다양한 실험집단을 사용하여 실험하였다. 본 논문에서 사용한 7개 실험집단은 한글 실험집단 2개(KT95, KRIST)와 5개(CACM, CISI, CRAN, LISA, MED)의 영문 실험집단을 사용하였다.

실험집단은 적합문헌이 미리 정의되어 있으며, 각 실험집단의 특징은 <표 1>과 같다.

<표 1> 실험집단 특징

	KT95	KRIST	CACM	CISI	CRAN	LISA	MED
주제	정보통신 신문기사	생명 기계공학	컴퓨터 과학	정보학	항공학	문헌정보학	의학
색인대상	제목 국문초록 저자 키워드	제목 국문초록 저자 키워드	제목 초록 저자 키워드	제목 초록 저자 키워드	제목 초록 저자	제목 초록 저자	초록
문헌수	4,414	13,515	3,204	1,460	1,400	6,004	1,033
색인어수	317,592	1,013,537	76,651	71,320	78,535	211,431	56,75
고유단어수	54,071	195,252	8,790	7,789	6,987	14,270	9,909
평균적합문헌수	28.98	11.21	15.31	40.97	8.16	10.73	23.2
질의수	50	28	52	76	225	33	30
질의당평균단어	3.48	9.23	10.46	19.08	8.48	18.58	10.43
평균바이트길이	1241	1960	682.81	1526	1174	612	1077
최대문헌빈도	2018	11877	1333	841	713	3546	440
평균 문헌길이	72.11	74.99	23.92	48.85	56.18	35.24	55

2.3 용어가중치 알고리즘

본 논문에서 사용한 용어가중치 알고리즘은 문헌 내 단어빈도와 역문헌빈도의 조합(*ntn, htn, atn, dtn, stn*) 수준과 코사인 정규화(*lnc, ntc, ltc, anc, atc*), 피벗 문헌길이 정규화(*dnb, dtu, Lnu, ltu*), *Okapi* 문헌길이 정규화(*otb, onb, otu*)에 따른 용어가중치 알고리즘으로 구분하였다(최성환. 2002). 이들 용어가중치 알고

리즘은 단어빈도, 역문헌빈도, 정규화의 세가지 요소들을 조합하여 만들어졌다.

질의어 가중치로는 〈$\log tf+1$〉을 공통적으로 사용하였다. 이 질의어 가중치의 특성은 단순출현빈도에 비해 가중치의 표준편차가 작으며 고빈도어의 지나친 영향을 줄인다. 실제로 이용자는 여러개의 동일한 단어를 반복해서 탐색어로 쓰는 일이 드물기 때문에 질의어 가중치는 거의 상수 역할을 하게 된다.

용어가중치 알고리즘 중에서 *atn*, *htn*, 그리고 *stn* 가중치는 문헌 내 단어빈도와 역문헌빈도의 조합이기는 하지만 실제로 정규화 기법을 사용하였다. *htn* 가중치 알고리즘은 고유단어수에 로그를 취하여 각 문헌마다 문헌길이를 반영하고, *atn*은 단어빈도를 문헌 내 최대단어빈도로 정규화시킨다(Harman 1992; Salton and Buckley 1988; Fox and Shaw 1994). 문헌내 단어들의 출현빈도 합으로 단어빈도를 정규화시키고 역문헌빈도를 최대값으로 정규화 시키는 *stn* 가중치는 정규화된 단어빈도와 역문헌빈도를 사용하였다. 즉, 용어가중치로 〈$\log tf+1$〉를 문헌 내 단어빈도의 합 〈*total tf*〉으로 정규화시키고 역문헌빈도를 최대역문헌빈도 〈max *idf*〉로 정규화시키는 가중치 알고리즘이다. *dtn* 가중치는 피벗 문헌길이 정규화요소를 사용하지 않고 단어출현빈도 요소와 역문헌빈도 요소만을 사용한 가중치 알고리즘이다(Singhal et al. 1996).

코사인 정규화 기법을 적용한 가중치 알고리즘으로, *lnc* 가중치는 출현빈도의 로그 값을 코사인 정규화함으로써 문헌벡터의 색인어들에 가중치를 부여하고, *ltc* 가중치는 단어의 출현빈도와 역문헌빈도를 곱한 값을 코사인 정규화함으로써 색인어들에 가중치를 부여한다(Buckley, Allan, and Salton 1995).

본 논문에서는 피벗 문헌길이 정규화를 적용한 가중치 알고리즘 중에서 *dnb* 가중치는 정규화 요소로써 바이트길이를 사용하였고, 나머지 *dtu, Lnu, ltu* 가중치들은 고유단어수 요소를 이용하였다. 피벗은 실험집단에서 평균 문헌길이(고유단어수)로 설정되었으며, 기울기 값은 상수 0.2로 설정되었다.

Okapi 문헌길이 정규화 기법을 적용한 가중치 알고리즘 *otu, otb,* 그리고 *onb* 가중치는 피벗 정규화 기법을 적용한 용어가중치 공식과 다른 형태의 공식처럼 보이지만 매우 유사한 형태의 특성을 지닌다. 단어출현빈도를 *Okapi* 바이트길이로 정규화 하는 *otb* 가중치는 *onb* 가중치 알고리즘에 정규화된 역문헌빈도의 공헌도를 가중한 공식이다.

2.4 질의확장

질의확장에는 여러 가지 문제들이 제기되어 왔다. 특히 확장용어의 선정 문제와, 확장용어의 가중치 할당, 확장용어의 크기 등이 문제가 된다.

본 논문에서는 확장할 용어를 선정하기 위해 동일문서를 공기단위로 하여 공기기반 유사도로 코사인 유사계수를 이용하였으며, 질의확장시 질의-용어간 유사도 $Sim(q,t)$를 아래 수식(2)와 같이 질의와 용어의 유사도값의 평균을 취하고 원질의어일 경우 1를 가중하였다. 최종 확장되는 질의어의 가중치는 수식(3)과 같이 질의-용어의 유사도값에 역문헌빈도를 곱하여 질의가중치로 사용하였다. 이는 확장용어에 대해서 고빈도어의 영향을 줄이기 위해서 사용된 방법이다.

$$Sim(q,t)=\frac{1}{n}\sum_{i=1}^{n}S(q_i,t) \qquad (2)$$

where
if exp*anding query* ≡ *original query term*
$sim(q,t)+1$
endif

$$w_{\exp}=Sim(q,t)*(\log\frac{N}{df}+1) \qquad (3)$$

질의-용어간 유사도를 구하면 경우에 따라 원질의어가 상위에 있지 않은 경우가 있다. 질의확장용어수가 작으면 이런 경우에는 원질의어가 제외될 것이다. 본 논문에서는 질의확장에서도 원질의어가 중요하다는 가정아래 질의-용어 유사도값에 1를 더하여 가중치로 사용하였다.

확장용어의 크기는 컬렉션마다 최적의 크기를 경험적으로 발견할 수 밖에 없다. 일반적으로 탐색결과기반 질의확장보다는 공기기반 질의확장이 확장할 용어가 더 많이 필요하다. 이는 탐색결과에 기반한 적합성 피드백은 검색된 문헌중에서 상위문헌만을 고려하기 때문으로 추론된다(Qiu & Frei 1993). 유사도 시소러스를 이용한 경우 MED(80개), CACM(100개), NPL(800개)에서 대략 확장할 용어수를 100개 정도로 하였을 때 안정적이고 좋은 성능향상을 보였다. 본 논문에서 확장용어의 크기는 높은 유사도값을 가지는 상위 용어수에 제한하지 않고 순위로 하였다. 이것은 경우에 따라 상위 용어들 중에서 동일한 유사도값을 가지고 있음에도 불구하고 용어수에 제한되어 확장에 포함되지 않는 경우가 발생할 수 있기 때문이다.

확장용어수(# *of* exp*anded terms*)
= 원질의어수(# *of original terms*) + 70

질의확장시 확장용어수(# *of* exp*anded terms*)는 원질의어수(# *of original terms*)에 순위 70위까지 질의 확장에 참여시킨다. 원질의어를 제외하고 순위 70위까지 확장시키는 것은 컬렉션 통계에서 알 수 있듯이 각 질의당 평균 단어수가 다르기 때문에 확장용어수에 원질의어의 수를 반영하는 것이 합리적이다. 본 논문에서 순위 70위까지 질의어를 확장시킨 것은 KT95에 대한 예비실험과 Qui & Frei(1993)의 연구 결과를 근거로 설정되었다.

3. 실험결과 및 분석

원질의어로 실험집단 7개를 대상으로 17개의 용어가중치 알고리즘을 적용하여 실험을 수행한 결과, 각 가중치 알고리즘은 실험집단에 크게 영향을 받지 않고 성능에 대한 상하위 그룹으로 크게 구분되었다. 11-포인트 평균정확률을 실험집단 전체에 대해 평균을 낸 결과 *htn* 가중치 알고리즘이 가장 좋은 평균 성능을 보였다. 특히 *htn*은 CACM과 CISI 실험집단에서 가장 높은 성능을 보였고 CRAN과 MED 실험집단에서도 2번째로 성능이 좋았다.

실험집단에 관계없이 문헌 내 단어빈도와 역문헌빈도의 조합에서 문헌길이를 반영하고 있는 *atn, stn,* 그리고 *htn* 가중치 알고리즘, 피벗 문헌길이 정규화 기법을 적용한 *dtu, ltu* 가중치 알고리즘, 그리고 *Okapi* 문헌길이 정규화 기법을 적용한 *otu, otb* 가중치 알고리즘들이 안정적이면서 좋은 성능을 보였다. 반면에 하위그룹이라 할 수 있는 용어가중치 알고리즘은 단어빈도와 역문헌빈도의 단순조합인 *ntn,* 그리고 실험집단에 따라 차이가 있기는 하지만 코사인 정규화 기법을 적용한 *anc, lnc,* 그리고 *ntc* 가중치 알고리즘으로서 전체적으로 낮은 성능을 보였다.

전체 실험집단에 대해 산출된 평균 성능을 기준으로 했을 때 4가지 범주 즉, 단어빈도와 역문헌빈도의 조합수준, 코사인 정규화, 피벗 문헌길이 정규화, *Okapi* 문헌길이 정규화 기법을 적용한 가중치 알고리즘에서 각 범주별로 최상위 평균 성능을 보이는 용어가중치 가운데 코사인 정규화 기법을 이용한 *ltc* 가중치 알고리즘이 가장 낮은 성능을 보였다. 다른 연구들(Buckley et al. 1995; Singhal, Buckley, and Mitra 1996; Savoy et al. 1997; Allan et al. 1998)에서도 이미 밝혀진 바와 같이 *htn, atn* 가중치를 비롯하여 피벗 코사인 정규화 기법을 적용한 *dtu, ltu* 가중치 알고리즘, 그리고 *Okapi* 문헌길이 정규화 기법을 적용한 *otb* 가중치가 단순 코사인 정규화 기법인 *ntc, ltc* 가중치 보다 성능이 우수한 것으로 나타났다.

<표 2> 질의확장에 따른 11-포인트 평균정확률과 순위
※[ ]는 질의확장전과 비교한 향상률

		low		← 평균질의길이 →				high	
	질의길이	3.48	8.48	9.23	10.43	10.46	18.58	19.08	
	실험집단 / 가중치	**kt95**	**cran**	**krist**	**med**	**cacm**	**lisa**	**cisi**	**평균**
high	**anc**	0.4483(5) [25.05]	0.4335(4) [16.19]	0.3416(5) [42.99]	0.5947(7) [21.02]	0.3313(12) [135.30]	0.2909(14) [24.00]	0.1839(11) [0.33]	0.3749(11) 37.84(1)
	lnc	0.4298(15) [19.12]	0.4519(1) [16.44]	0.3223(13) [50.89]	0.5997(5) [21.89]	0.3759(4) [79.34]	0.3339(8) [31.10]	0.201(3) [6.80]	0.3878(5) 32.23(2)
	onb	0.4405(10) [11.77]	0.4327(5) [8.91]	0.3571(1) [47.99]	0.5926(9) [18.19]	0.3318(11) [27.91]	0.3494(4) [14.90]	0.1756(15) [-4.31]	0.3828(8) 17.91(3)
	Lnu	0.4249(16) [12.77]	0.4166(12) [9.09]	0.3213(14) [45.58]	0.5861(12) [19.32]	0.3802(1) [26.99]	0.3482(5) [9.88]	0.19(9) [1.06]	0.3810(9) 17.81(4)
	ntc	0.4304(14) [22.20]	0.4261(7) [6.77]	0.2942(16) [42.82]	0.6035(3) [17.00]	0.3791(2) [13.30]	0.2891(15) [2.88]	0.2101(1) [-3.49]	0.3761(10) 14.50(5)
	ntn	0.3861(17) [16.68]	0.3442(17) [3.21]	0.2768(17) [51.09]	0.5394(17) [14.09]	0.3221(15) [15.08]	0.2958(13) [1.58]	0.1762(14) [-11.59]	0.3344(17) 12.88(6)
	dnb	0.4338(12) [9.16]	0.4079(13) [6.11]	0.3309(10) [36.91]	0.5765(14) [18.06]	0.3282(14) [13.06]	0.3329(9) [2.43]	0.1767(13) [-6.31]	0.3696(14) 11.35(7)
↑	**ltc**	0.4494(3) [15.38]	0.4344(3) [3.43]	0.3365(8) [29.27]	0.6104(1) [15.41]	0.3761(3) [8.67]	0.3071(12) [-1.73]	0.2074(2) [-9.35]	0.3888(4) 8.73(8)
향	**atc**	0.4559(1) [16.39]	0.4182(11) [3.57]	0.3255(12) [12.86]	0.6018(4) [15.11]	0.3345(10) [17.45]	0.2755(16) [-1.18]	0.187(10) [-13.10]	0.3712(13) 7.30(9)
상	**otb**	0.4519(2) [7.31]	0.4439(2) [-0.43]	0.3439(3) [9.63]	0.6047(2) [13.45]	0.3464(9) [-4.52]	0.3671(1) [-1.87]	0.1917(8) [-12.70]	0.3928(1) 1.55(10)
률	**ltf**	0.4379(11) [8.20]	0.4207(9) [-1.71]	0.3346(9) [20.75]	0.5898(11) [11.83]	0.3727(5) [-3.19]	0.3452(6) [-11.17]	0.1997(4) [-13.89]	0.3858(6) 1.55(11)
↓	**dtu**	0.4471(6) [5.67]	0.4242(8) [-2.44]	0.3467(2) [17.80]	0.5966(6) [13.44]	0.3644(8) [-1.54]	0.3647(2) [-8.78]	0.1918(7) [-14.03]	0.3908(2) 1.45(12)
	ltu	0.4418(9) [8.20]	0.4198(10) [-1.76]	0.3378(7) [17.95]	0.5932(8) [13.14]	0.3668(7) [-4.33]	0.3452(7) [-10.34]	0.1956(6) [-13.68]	0.3857(7) 1.31(13)
	htn	0.4446(7) [6.16]	0.4267(6) [-2.54]	0.3417(4) [10.98]	0.5921(10) [11.53]	0.3698(6) [-6.00]	0.3544(3) [-7.95]	0.1976(5) [-15.91]	0.3896(3) -0.53(14)
	atn	0.4483(4) [5.61]	0.3973(14) [-5.27]	0.341(6) [4.03]	0.58(13) [10.60]	0.3282(13) [-12.41]	0.3323(10) [-6.37]	0.1803(12) [-19.08]	0.3725(12) -3.27(15)
	otu	0.444(8) [4.91]	0.373(15) [-3.49]	0.3305(11) [-0.93]	0.5514(16) [10.21]	0.3091(16) [-13.83]	0.3299(11) [-5.72]	0.1679(17) [-20.99]	0.3580(15) -4.26(16)
low	**dtn**	0.432(13) [2.30]	0.3525(16) [-4.45]	0.3181(15) [4.81]	0.5562(15) [9.83]	0.3001(17) [-9.58]	0.2489(17) [-19.00]	0.1733(16) [-15.55]	0.3402(16) -4.52(17)
	표준편차	7.1341	5.0435	18.1716	8.2039	33.715	8.2311	7.1562	

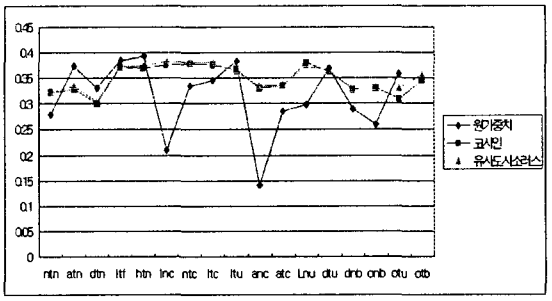
질의확장은 각 가중치 알고리즘과 컬렉션에 따라 다르기 때문에 최적의 성능을 얻기 위해서는 경험적으로 얻어질 수밖에 없다. 본 논문에서는 질의를 확장할 경우 가중치 알고리즘이 검색효율성에 미치는 영향을 분석하기 위한 것이 주된 목적으로 다른 변수들을 고정시키고 동일한 조건하에서 질의를 확장시켰다. 질의확장 결과 한글 컬렉션에서는 KRIST에서 *otu(-0.93%)* 가중치 알고리즘을 제외하고 KT95에서는 최고 25.05%(*anc*), KRIST에서는 51.09%(*ntn*)의 향상률을 나타냈다. 그러나 영문컬렉션에서 특히 CISI, LISA 컬렉션에서는 질의확장의 경우 대부분의 가중치에서 성능이 향상되지 않았다. 이는 질의의 길이가 영향을 미치는 것으로 질의확장에 부정적인 영향을 미치는 것으로 분석된다. CISI, LISA 컬렉션은 질의당 평균단어수가 각각 19.08, 18.58개로 다른 컬렉션에 비해 질의벡터길이가 길다.

질의확장에서 특이한 사항은 모든 컬렉션에서 코사인 정규화 가중치 기법을 적용한 알고리즘들이 성능향상이 높을뿐만 아니라 순위 5이 이내에 있는 것이 많다. 특히 KT95(*atc*), CRAN(*lnc*), MED(*ltc*), CISI(*ntc*) 컬렉션에서는 코사인 정규화가 가장 높은 검색효율을 보이고 있다. 반면에 원질의어를 이용한 실험에서 좋은 성능을 보인 알고리즘들은 질의확장후 대부분이 낮은 검색효율과 향상률을 보였고 오히려 향상률이 감소한 경우도 많았다. 비록 코사인 정규화가 질의확장에서 좋은 검색효율을 보였으나 역시 문제가 발견되었다. 즉, CRAN, CACM, LISA, CISI 컬렉션에서는 원질의어를 이용하여 가장 좋은 성능을 보였던 알고리즘들의 검색성능과 비교하면 오히려 낮은 성능을 보이고 있다.

4. 결과

기존의 질의확장 혹은 적합성 피드백 연구에서 코사인 정규화를 사용하여 검색성능을 향상시킨 연구들이 많다(Salton & Buckley 1990; Qiu & Frei 1993; Buckley, Salton and Allan 1994). 본 논문에서 실험한 결과를 근거로 하였을 때 이는 낮은 검색성능을 보였던 것이 검색공간의 확장으로 성능이 크게 향상되었을 가능성이 있다. 이를 검증하기 위해 Qiu & Frei(1993)가 질의확장의 성능을 크게 향상시킨 바 있는 유사도 시소러스를 이용하여 추가 실험을 하였다. <그림 2>를 보면 본 논문에서 사용한 코사인 유사도 모델과 Qiu & Frei(1993)가 사용했던 유사도 시소러스 모델에서 색인어 가중치로 *htn*를 적용하여 CACM 컬렉션을 대상으로 비교한 것이다. 각 가중치 유사도 모델간의 커다란 차이는 보이지 않고 유사도 시소러스도 코사인정규화 가중치 알고리즘에서 상당한 성능향상이 있다는 것을 확인할 수 있다.

따라서 지역적인 적합성 피드백이나 전역적 질의확장의 경우 검색성능이 좋은 것으로 알려진 최신 가중치 알고리즘들을 적용하여 실험하는 것을 고려해야 할 것이다.



<그림 2> 코사인 유사도와 유사도 시소러스 비교

참고문헌

최성환. 2002. "용어 가중치 결합의 검색효율성에 관한 연구", 연세대학교 석사학위논문.

Allan, J., J. Callan, W.B. Croft, L. Ballesteros, D. Byrd, R. Swan, and J. Xu. 1998. "*INQUERY* does battle with TREC-6", <http://trec.nist.gov/pubs/trec6/>

Buckley, C., J. Allan, and G. Salton. 1995. "Automatic routing and retrieval using SMART: TREC-2", Information Processing & Management, 31(3), pp. 315-326.

Buckley, C., Salton G., Allan J., and Singhal A. 1995. "Automatic query expansion using expansion using SMART: TREC3", <http://trec.nist.gov/pubs/trec3/>

Fox, E. A., and J.A. Shaw. 1994. "Combination of multiple searches", <http://trec.nist.gov/pubs/trec3>

Harman, D. 1992. "Relevance feedback revisited", Proceedings of the 15th Annual International ACM SIGIR Conference on Research and Development in Information Science, pp. 1-10.

Qiu, Yonggang, and Frei, Hans-Peter. 1993. "Concept based query expansion", In Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval, 160-169.

Salton, G., and C. Buckley. 1988. "Term-weighting approaches in automatic retrieval", Information Processing & management, 24(5), pp. 513-523.

Savoy, J. 1997. "Ranking Schemes in Hybrid Boolean Systems: A New Approach", JASIS, 48(3), pp. 235-253.

Singhal, A., C. Buckley, and M. Mitra. 1996. "Pivoted document length normalization", Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Science, pp. 21-29.

Singhal, A., Gerard Salton, Mandar Mitra, and Chris Buckley. 1996. "Document length normalization", Information Processing and Management, 32(5), pp. 619-633.