DOI: 10.3745/KIPSTB.2008.15-B.2.171

메타 검색에서 외래어 질의 정제 효과

이 재 성†

요 약

문서에서 외래어가 일관되게 사용되지 않고 여러 이형태로 사용되고 있기 때문에, 정확한 질의어 일치를 지원하는 검색 시스템에서 외래어 질의로 문서를 검색하는데 어려움이 많다. 본 논문에서는 하나의 외래어로 질의할 경우, 원 질의어와 같은 뜻의 다양한 이형태 외래어 질의로 자동 확장하고 정제하여 더 많은 관련 문서를 손쉽게 검색할 수 있는 메타 검색 방법을 제안한다. 이 방법은 1차로 원 질의어에서 다양한 외래어 이형태를 통계적 방법으로 확장하고, 2차로 그 결과를 각 검색 엔진에게 질의하여 일정 개수 이상의 질의어가 문서에 나타났는지, 원 질의어의 문맥과 유사한 문맥에서 그 질의어가 쓰였는지를 비교하여, 같은 뜻의 유효한 외래어를 판별해 내고 이를 이용하여 검색할 수 있도록 한다. 실험 결과, 기준점으로 쓰인 1차로 만든 이형태로 검색했을 때 F값은 평균 38%이었으나, 제안된 방법인 2차로 정제된 질의어로 검색했을 때의 F값은 평균 81%로 매우 향상된 결과를 보였다.

키워드 : 질의어 확장, 외래어 질의, 질의어 정제, 메타 검색, 정보 검색

The Refinement Effect of Foreign Word Transliteration Query on Meta Search

Jae Sung Lee†

ABSTRACT

Foreign word transliterations are not consistently used in documents, which hinders retrieving some important relevant documents in exact term matching information retrieval systems. In this paper, a meta search method is proposed, which expands and refines relevant variant queries from an original input foreign word transliteration query to retrieve the more relevant documents. The method firstly expands a transliteration query to the variants using a statistical method. Secondly the method selects the valid variants: it queries each variant to the retrieval systems beforehand and checks the validity of each variant by counting the number of appearance of the variant in the retrieved document and calculating the similarity of the context of the variant. Experiment result showed that querying with the variants produced at the first step, which is a base method of the test, performed 38% in average F measure, and querying with the refined variants at the second step, which is a proposed method, significantly improved the performance to 81% in average F measure.

Key Words : Query Expansion, Foreign Word Transliteration Query, Query Refinement, Meta Search, Information Retrieval

1. 서 론

우리나라와 외국과의 교류가 증가함에 따라 새로운 용어가 많이 사용되고 있고, 특히 외래어의 사용이 급증하고 있다. 외래어는 고유명사로 쓰이거나 의미상으로 전문적인 용도로 쓰이는 경우가 많아 전문 문서 등에서 중요 키워드로 많이 사용되고 있다[1, 2, 3].

외래어는 하나의 원어에서 온 것이라도 다양하게 표기되는 경우가 많다. 그 원인은 여러 가지를 들 수 있는데, 같은 단어라도 어느 언어를 원어로 정하는가에 따라 발음이 달라지고, 발음이 표기되는 과정에서도 한국어와의 언어 구조 차이로 인해 정확하게 대응되는 표기가 없어 비슷한 자소로 표현되는 경우가 있으며, 발음보다는 사람들의 직관에 의해 외래어 글자의 외형적 발음을 그대로 표기하는 경우 (소위 '눈말표기') 등이 있기 때문이다[4, 5].

외래어에 대한 표준 표기 및 표기 규칙을 국립국어원에서 제정하여 공표하고 있으나, 대중들이 쉽게 그 표준 표기나 표준 표기 규칙을 따르지 않는 경우가 많고, 새로운 모든 용어에 대해 표준 표기를 제정하는 것도 불가능하여 다양한 표기가 여러 문서에 존재하게 된다.

이러한 외래어의 다양한 표기로 인해, 키워드 기반 정보검색시 일반 사용자들이 불편한 경우가 많이 있다. 특히, 인터넷 정보검색시 대부분 포탈 사이트의 정보검색 시스템이

※ 이 논문은 2006년도 충북대학교 학술연구지원사업의 연구비지원에 의하여 연구되었음.
† 종신회원 : 충북대학교 컴퓨터교육과 부교수
논문접수 : 2007년 11월 12일, 심사완료 : 2008년 1월 3일

정확한 일치만을 통해 검색을 하는 경우가 많으므로, 외래어를 정확하게 입력하지 않으면 원하는 결과를 얻지 못하는 경우가 많다.

본 연구에서는 현재 우리나라에 존재하는 여러 포탈 사이트를 활용한 외래어 질의 메타 검색 시스템을 구현하고 그 성능을 평가한다. 외래어 질의 메타 검색 시스템은 하나의 외래어를 입력받아, 그 입력으로부터 다양한 이형태의 외래어를 생성하고, 이 외래어를 기초로 여러 포탈 사이트를 자동 검색할 수 있도록 한다. 외래어의 이형태 생성은 특성상 실제 사용되지 않는 외래어가 생성될 수 있을 뿐만아니라 다른 의미의 외래어도 생성할 수 있으므로, 이를 처리하기 위해 자동으로 포탈 사이트에 1차 검색을 하고 그 결과를 분석하여 사용자에게 적합한 이형태 질의어만을 제시하도록 하였다. 이에 대한 효과적인 처리 방식을 본 논문에서 제안하고 실험을 통해 검증한다.

2장에서는 외래어 질의어 처리에 대한 기존 연구를 살펴본 후, 본 연구에서 사용하고 있는 외래어 이형태 자동 생성 방법에 관해 간단히 소개한다. 3장에서는 본 연구에서 제안하는 외래어 질의 메타 검색 시스템과 그 구현 구조를 설명한다. 4장에서는 외래어 질의 메타 검색 시스템의 성능 향상을 위해, 기존 포탈 사이트의 검색 시스템에 1차 질의를 하고, 이 결과를 이용하여 외래어 이형태를 효과적으로 생성하는 방법을 제시한다. 5장에서는 외래어 질의 처리 효과에 대한 실험 결과를 중심으로 외래어 질의 메타 검색 시스템의 성능을 평가하며 6장에서 결론을 맺는다.

2. 관련연구

검색 시스템에서 외래어 이형태를 검색하기 위한 방법으로서 대상 단어를 찾아 질의어와 유사도를 비교하는 방법과 질의어를 근거로 먼저 가능한 외래어 이형태를 생성하고 이를 대상 단어와 비교하는 방법이 있다.

첫 번째 방법은 다시 1. 문자열 차이를 이용하는 방법과 2. 중간 표현으로 변환하여 비교하는 방법으로 나눌 수 있다. 1의 방법으로는 외래어 질의어와 이형태 단어(target word)의 편집 거리(edit distance)나 n-그램을 계산하여 이형태 외래어를 찾는 방법[6] 등이 있고, 2의 방법으로는 외래어 질의어와 이형태 단어를 음성적 유사도에 기반한 코드로 변환하거나 원어 단어(예를 들어 영어)로 복원(back transliteration)하여 대조(match)하는 방법[1, 7, 8] 등이 있다. 메타 검색시스템에서는 기존 포탈 사이트의 검색 시스템을 이용하지만, 그 검색 시스템에서 제공하는 색인어, 즉 대상 단어를 직접 대조할 수 없으므로, 이 방법들을 사용하기에는 부적합하다.

두 번째 방법은 질의어에서 바로 외래어 이형태를 생성하는 방법으로 이형태 생성 과정을 전처리기로 이용하면 주 시스템을 변형시키지 않고 다양한 이형태 외래어를 처리할 수 있다. 메타 검색 시스템에서도 이 방법을 이용하면 하나의 외래어 질의어에 대해 다양한 외래어 이형태 질의를 할 수 있다. 즉, 외래어 이형태를 메타 검색 시스템에서 생성한 후 이를 각각의 새로운 질의어로 필요한 포탈 사이트 검색시스템에게 넘겨주고, 그 결과를 받아 메타 검색 시스템에서 종합하면 된다.

기존의 외래어 이형태 생성 기법은 대부분 우선 순위에 대한 고려가 없거나 간단한 우선 순위를 사용하여 이형태를 생성하였고, 그 생성 규칙도 단순한 교환 규칙을 사용하였으며, 수동으로 구축되었다[9, 10, 11]. 이 방법들은 외래어 이형태를 찾아내는데 이용되었지만, 실제 사용되지 않은 너무 많은 불필요한 외래어가 생성되어 전처리기로 사용하기에는 비효율적이다.

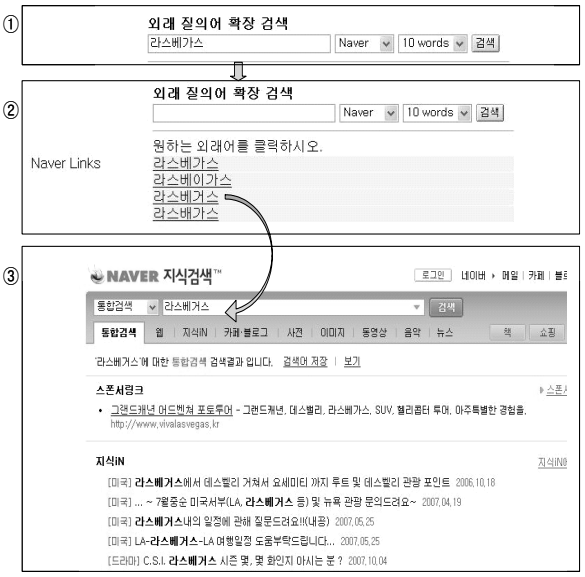
이를 해결하기 위해 논문[12]에서는 확률 문맥 의존 치환(PCSR) 방법을 제안하였다. PCSR은 단어를 자소로 나열하고, 각 자소가 그 단어에서 바뀔 확률(전역적 요소)과 그 자소가 어떤 자소로 바뀔 확률(지역적 요소)을 구분하여, 두 확률의 곱을 이용하여 변이체 생성의 우선 순위를 결정하였다. 각각의 확률은 또 좌우의 글자(문맥)을 고려하고 학습하여 정확성을 높였다. PCSR 방법은 기존에 사용한 단순 치환에 의한 방법보다 조기에 많은 실제 사용되는 외래어 이형태를 생성해 내었다. 이 방법이 기존 방법보다 매우 향상된 것이기는 하지만 여전히 실제 사용하지 않는 단어를 많이 생성하고 있어 이를 해결할 필요성이 있었다. 본 논문에서는 이 방법을 실제 메타 검색에 사용하고, 1차 검색 결과를 이용하여 성능을 높이는 방법을 제안한다.

3. 외래어 질의 메타 검색 시스템

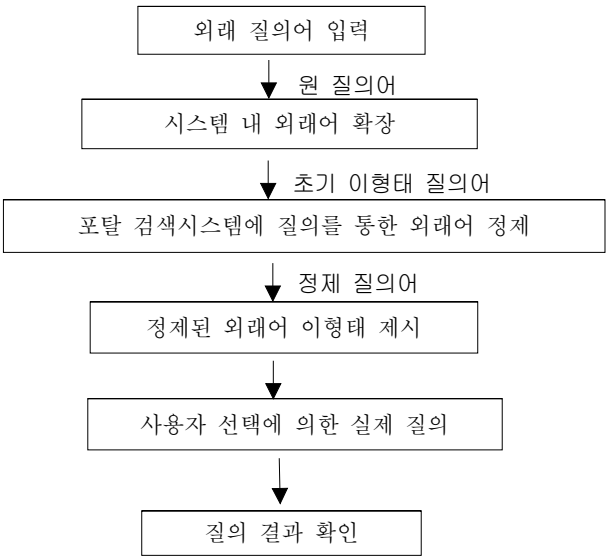
대부분의 한국어 검색 포탈 사이트에서는 외래어 질의를 할 경우, 같은 뜻을 가진 유사 외래어를 검색하지는 않는다. 그러나 그 외래어의 이형태를 각각 입력할 경우는 그에 대한 검색 결과를 제공한다. 이는 특별히 외래어 이형태에 대해 처리하지 않고, 이형태라도 각각 독립적인 키워드로 처리했기 때문이다. 따라서, 특별한 외래어에 대해 입력을 할 경우 정확하게 일치된 외래어 표기가 아닐 경우, 다양하게 표현된 웹 문서를 찾을 수 없는 경우가 많다.

외래어 질의 메타 검색 시스템은 이러한 불편을 해결하기 위해 제안되고 개발되었다. 즉, 하나의 외래어를 입력하면 그에 해당되는 이형태를 자동 생성해 주고, 사용자가 필요한 이형태를 클릭함으로써 손쉽게 다양한 이형태에 대한 검색을 수행할 수 있다.

메타검색은 대개 다른 검색시스템들의 검색 결과를 종합하여 적합한 결과만을 제시하는 것이 일반적이다[13]. 외래어 질의 메타 검색 시스템은 현재 초기 버전으로 하나의 외래 질의어(외래어로 된 질의어)만을 받고 있고 검색도 하이퍼링크를 클릭하여 검색하도록 하고 있다. 현재는 실험적으로 3개의 검색 엔진 사이트(Naver, Google, Yahoo)만을 연결하였고, 질의어 정제 실험에 대한 효과 분석에는 편의상 그 중 2개(Naver, Google)만을 사용하였다. 메타 검색에서 호출하는 검색 엔진은 그 검색 사이트에 대한 질의 방법만



(그림 1) 외래 질의어 처리 과정



(그림 2) 외래 질의어의 내부 처리 단계

알면 간단히 추가할 수 있으며, 사용자의 편의를 위해 앞으로 더 많은 검색 엔진을 추가할 예정이다.

(그림 1)의 ①은 외래어 질의 입력 화면이다. 주어진 질의어 입력창에 해당 외래어를 입력한 후, 원하는 검색 포탈 사이트와 원하는 이형태 생성 갯수 (원 질의어 포함)를 선택하고 검색 키를 누르면 (그림 1)의 ②처럼 해당 이형태가 자동으로 생성된다. 사용자는 해당 이형태를 선택하여 클릭하면 자동으로 포탈 사이트의 검색 결과를 (그림 1)의 ③처럼 볼 수 있다.

시스템내에서의 이형태 생성은 크게 2단계로 처리되고 있다. 첫 단계는 이형태의 생성 프로그램을 통한 이형태의 생성이다[12]. 이 단계에서 생성된 외래어를 '초기 이형태 질의어'로 부르며 여기에는 사용되지 않는 외래어나 다른 뜻의 이형태도 포함될 수 있다. 두번째 단계인 정제과정에서는 외래어를 직접 포탈 사이트에서 검색하고 그 결과를 받아온다. 받아온 결과를 비교하여 사용되지 않는 외래어를 제거하고, 다른 뜻으로 사용될 가능성이 많은 외래어를 제거한다. 이 결과로 나온 외래어들을 '정제 질의어'라고 부른다. 정제과정이 끝나면 이형태 생성 결과를 사용자에게 표시한다. 이런 전체 단계를 도표로 표시한 것이 (그림 2)이다.

4. 외래어 질의 정제

문맥정보를 이용한 외래어 이형태 생성 방법[12]은 기존의 방법에 비해 효과적으로 외래어 이형태를 생성하기는 하지만, 아직도 많은 불필요한 이형태를 포함하고 있다. 따라서, 사용자가 더 편리하게 이형태를 선택하기 위해 실제 사용되고 있고, 올바른 검색 결과를 보여주는 이형태만을 사용자의 관여 없이 자동으로 제시하는 방법이 필요하다. 외래 질의어 정제 과정에서는 프로그램 내부에서 검색 사이트에 1차 질의를 하고 그 결과를 분석하여 이형태 질의어 중 적합한 것만을 선택하여 사용자에게 제시한다.

초기 이형태 외래어에는 다음과 같은 2 종류의 잘못된 외래어가 포함되어 있다.

1. 부재 외래어: 외래어로서 실제 사용되지 않아 존재하지 않는 단어.
2. 이의 외래어: 외래어로 존재하지만, 원래 외래어와 뜻이 다른 단어.

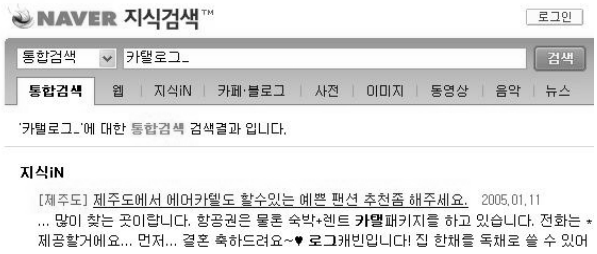
이러한 불필요한 외래어들에 대응되는 외래 질의어들을 질의 과정에서 자동으로 제거하기 위해 검색 시스템 환경으로 다음과 같이 정의한다.

3. 부재 외래 질의어: 실제 포탈 사이트에서 검색을 하여 검색 결과가 없는 외래어.
4. 이의 외래 질의어: 실제 포탈 사이트의 검색 결과에서 원 질의어와 다른 의미로만 쓰인 외래어.

부재 외래 질의어의 경우 비교적 간단하게 판단할 수 있다. 외래 질의어를 포탈 사이트에 입력하여 검색한 결과를 살펴 보면, (그림 3)의 예처럼 명확하게 검색 결과가 없다고 표시하는 경우와 (그림 4)처럼 '카탤로그'에 대한 질의를 '카탤'과 '로그' 분리하여 다른 검색 결과를 보여주는 경우가 있다. 이 모든 경우를 부재 외래어로 판별한다. 프로그램에서는 간단하게 결과에 질의어가 일정 갯수 이하로 포함되면 부재 외래 질의어로 판별한다. 포함된 질의어의 기본 갯수는 검색 사이트마다 다르므로 검색 사이트 선택시 자동으로 기본 갯수를 정하도록 했다.

이의 외래 질의어의 경우, 검색 결과에 나온 문맥 정보를 이용하여 프로그램으로 자동 판별한다. 이는 같은 뜻의 외

(그림 3) 해당 문서가 없는 부재 질의어 예



(그림 4) 질의어가 분리되어 검색된 예

래어가 사용되었을 경우, 이 단어 좌우에 사용된 단어들, 즉 문맥도 같을 것이라는 가정하에 처리하는 것이다. 예를 들어, 질의어 '데이터'를 확장하여 '데이타', '데이트'가 나왔고 다음과 같은 문맥이 추출되었다고 하자.

데이터 처리 시스템 (1)

데이타베이스 시스템 (2)

데이트 주선 이벤트 (3)

원 질의어 '데이터' (1)과 확장 질의어 '데이타' (2)에 대해 '시스템'이라는 단어가 2 단어 거리내에서 같이 사용되었다. 반면에 다른 확장 질의어 '데이트' (3)은 원 질의어의 문맥과 공통 단어가 없다. 즉, '데이터'와 '데이타'는 같은 뜻으로 사용되었기 때문에 문맥이 비슷한 반면, '데이트'는 다른 뜻으로 사용되기 때문에 문맥이 다르다. 따라서 문맥 정보를 이용하여 유사도를 측정하면 잘못된 확장 질의어, 즉 이의 외래 질의어를 제거할 수 있을 것이다.

유사도 측정을 위해 질의어의 좌우에 나오는 일정한 범위(윈도 크기)의 어절들을 추출하여 이를 문맥 정보로 보고 이들 어절들을 토큰으로 하여 계산한다. 실제 검색 결과는 많은 HTML 태그가 포함되어 있으므로 좌우 문맥 추출 전에 태그들을 모두 제거한 후, 순수한 텍스트의 어절만을 추출한다.

유사도는 원 질의어로 나온 문서와 초기 이형태 질의어로 나온 문서들을 비교하여 계산한다. 이 과정에서 원 질의어는 항상 사용자의 의도가 제대로 반영된 올바른 질의어라는 가정을 한다. 실제 원 질의어가 잘 쓰이지 않는 부재 외래 질의어인 경우도 드물게 있는데, 이는 본 연구의 기본 가정에서 벗어남으로 제외하였다.

정제 질의어를 생성하기 위해서는 포함된 질의어 갯수가 일정 수준 이상이고, 문맥 유사도도 일정 이상인 경우의 이형태 질의어에 대해서만 선택한다. 문맥 유사도는 어절 집합들을 문서 벡터로 보고 이를 다음과 같은 문서 유사도 계산식 (4), (5), (6)을 각각 이용하여 계산한다[14].

$$COSINE(X,Y)=\frac{\sum x_i \cdot y_i}{\sqrt{\sum x_i^2 \cdot \sum y_i^2}} \quad (4)$$

$$DICE(X,Y)=\frac{2\times\sum x_i \cdot y_i}{\sum x_i^2+\sum y_i^2} \quad (5)$$

$$JACCARD(X,Y)=\frac{\sum x_i \cdot y_i}{\sum x_i^2+\sum y_i^2-\sum x_i \cdot y_i} \quad (6)$$

5. 실험 및 토의

실험을 위해 [12]의 연구에서 사용된 결과를 사용하였다. 즉, 총 277개 등가(equivalent) 외래어 그룹으로부터 추출한 학습 데이터 및 미학습 데이터에서 다시 각각 27개의 단어를 추출하여 검색 결과가 없는 7단어를 제외하고 총 47 단어를 테스트 데이터로서 검색을 위한 원 질의어로 사용하였다.

실험 결과를 확인하기 위해 수동으로 각각의 검색 결과를 확인한 후, 앞장에서 설명한 기준에 따라 부재 외래 질의어와 이의 외래 질의어를 제외하여 정답 외래 질의어 집합을 만들었다.

각 질의어에 대해 9개의 이형태를 더 생성하여 총 10개의 질의를 하였고, 2개의 사이트(Google[15], Naver[16])별로 질의를 하였다. (질의어 정제 효과를 판단하는데 2개의 사이트면 적당하다고 판단하여 편의상 2개로 제한하였다.) 따라서, 질의 결과 만들어진 html문서는 총 940 (=2*10*47)개이다.

평가는 재현율과 정확률 및 이 둘을 결합한 F값을 이용하였다. 그 수식은 아래의 (7), (8), (9)와 같다[17]. 실험결과에서는 각각의 질의어 종류에 따라 F값을 계산한 후, 그 평균을 각 방법의 F값으로 나타냈다.

$$\text{재현율}(R)=\frac{\text{정제된 관련 외래 질의어 수}}{\text{전체 관련 외래 질의어 수}} \quad (7)$$

$$\text{재현율}(P)=\frac{\text{정제된 관련 외래 질의어 수}}{\text{정제된 외래 질의어 수}} \quad (8)$$

$$F\text{값}=\frac{2RP}{R+P} \quad (9)$$

실험의 기준으로 사용하기 위해 원 질의어를 그대로 사용했을 경우에 찾은 초기 이형태 질의어를 이형태 외래어의 전부라고 가정하였다. (즉 이때의 재현율을 1.0으로 계산함)

성능 평가를 위해, 문맥 윈도 크기를 1부터 20까지 변화

〈표 1〉 각 방법에서의 최고 F값

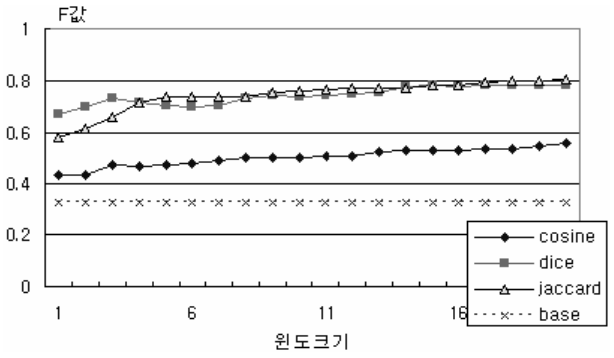
사이트	유사도 계산식	윈도 크기	유사도 한계값	F값
Google	cosine	19	0.15	0.80311
	dice	20	0.07	0.80242
	jaccard	**20**	**0.04**	**0.80501**
	count	-	0.0	0.49117
	base(1차 질의 결과)	-	-	0.32553
Naver	cosine	12	0.04	0.73078
	dice	**17**	**0.01**	**0.82158**
	jaccard	18	0.01	0.78956
	count	-	0.0	0.63670
	base(1차 질의 결과)	-	-	0.43617

〈표 2〉 각 방법의 최고 F값 평균

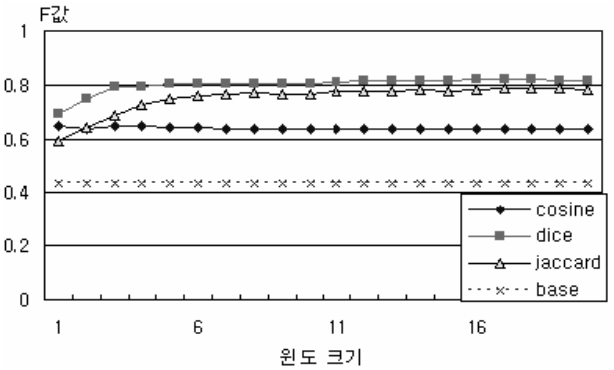
방법	F값 평균
cosine	0.76695
dice	**0.81200**
jaccard	0.79729
count	0.56394
base(1차 질의 결과)	**0.38085**

시키고 각각의 유사도 계산식에 대해 유사도 한계값을 0.01부터 0.2까지 0.01씩 변화시키면서 평가를 하였다. 즉, 윈도 크기 20가지와 유사도 한계값 20가지에 대한 총 400가지의 경우에 대해 각 사이트별로 각 유사도 계산식 3가지(cosine, dice, jaccard)를 계산하여 최상의 결과를 나타낼 때의 윈도 크기와 유사도 한계값을 구하였다. (실제 질의를 한 횟수는, 질의어 개수(940) * 윈도크기 종류(20) * 유사도 한계값 종류(20) * 사이트(2) * 유사도 계산식(3) = 2,256,000 이다.) 또, 기준(base)이 되는 초기 이형태 질의어, 즉 1차 질의어에 대한 F값과 해당 질의어가 검색되었는지만을 판별한 방법(count)에 대한 F값을 포함하여 <표 1>에 나타내었다. 또, 2개의 사이트에서 나온 결과를 종합하기 위해 <표 1>에 나타낸 각 방법에 대한 최고 F값을 평균하여 <표 2>에 나타내었다.

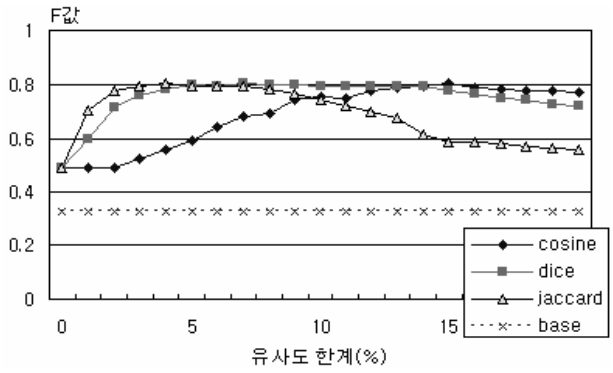
성능은 유사도 계산식, 윈도크기, 유사도 한계값, 사이트에 따라 각각 다르게 나타났으며, 사이트에 관계없이 문맥 유사도를 사용한 방법들(cosine, dice, jaccard)이 단순한 방법(base, count)보다 훨씬 우수했다. <표 1>에서 보듯이 각 사이트의 특성에 따라 윈도 크기 값이나 유사도 한계값, 유사도 계산식를 다르게 할 경우, Google은 jaccard 방식이 윈도 크기 20, 유사도 한계값 0.04에서 최대 F값 0.80501을 나타냈고, Naver의 경우, dice 방식이 윈도 크기 17, 유사도 한계값 0.01에서 최대 F값 0.82158을 나타냈다. <표 2>에서와 같이 두 사이트의 최대값에 대한 평균 F값은 81%(dice



(그림 5) 윈도 크기 변화에 따른 성능 변화: Google (유사도 한계 0.04)



(그림 6) 윈도 크기 변화에 따른 성능 변화: Naver (유사도 한계 0.01)



(그림 7) 유사도 한계값 변화에 따른 성능 변화: Google(윈도크기 20)

방법)이고, 이는 단순하게 검색결과가 있는 문서만을 계산한 방법(count)의 평균값 56%보다 25%포인트 증가한 값이며, 기준값(base)으로 비교했을 때는 그 평균값 38%보다 무려 43% 포인트 증가한 값이다.

(그림 5)와 (그림 6)은 유사도 한계값을 최고의 F값을 나타내는 점으로 고정해 두고, 윈도 문맥 크기를 변경시킬 경우에 따른 F값의 변화를 보여준다. (그림 5)는 Google의 경우로 유사도 한계를 0.04로 했고, (그림 6)은 Naver의 경우로 유사도 한계를 0.01로 했을 경우이다. 두 경우 모두, 문맥의 크기가 클수록 성능이 향상되어가지만, 초기에만 약간 빨리 증가하고 10이상에서는 거의 증가가 없거나 약간 있었다.

(그림 7)과 (그림 8)은 유사도 한계값 변화에 따른 성능 변화를 보여준다. 유사도가 0인 경우는 count방법을 사용하는 경우로 질의어 검색 결과가 있는 경우를 모두 정답으로 처리한 경우이다. 따라서 다른 유사도 한계값의 경우보다 낮은 성능을 보였고, (그림 8)의 Naver의 경우, 매우 급격한 성능 변화를 보였다. 두 사이트 모두 약간 한계값이 변화해도 성능 변화가 많았으며, dice와 jaccard 함수가 작은 한계값에서 좋은 성능을 낸 반면, cosine값은 상대적으로 큰 한계값에서 좋은 성능을 나타냈다. 이는 함수의 특성에 따른 것으로 생각된다. 또, (그림 8)의 Naver가 (그림 7)의 Google보다 더 작은 한계값에서 최고의 성능을 나타내는데, 그 이유는 Naver에서 추출한 문맥 어절들이 Google에서 추출한 것보다 상대적으로 많기 때문에 약간의 유사도 차이가 성능에 크게 영향을 미치는 것으로 분석된다. (실제 Naver에서는 Google에 비해 스폰서 링크, 지식iN, 블로그 등을 포함한 더 많은 관련 링크를 제공한다.)

<표 3>은 제안된 방법으로 정제한 외래 질의어 예를 임의로 하나 선정한 것이다. 이 표에서 보듯이 원 질의어 '라스베가스'에 대해 초기 이형태 질의어 10개가 만들어지고, 이 중 사람이 검색결과를 확인하여 정답 질의어로 7개를 판단하였고, 이를 '*'로 표시하였다. '락스베가스'와 같은 이형태는 실제 사용될 것 같아 보이지는 않지만, 검색 결과 문서에서 사용되고 있어 정답으로 처리했다. 본 논문에서 제안한 방법 중 가장 성능이 우수한 파라미터 값으로 설정하여 정제 질의어를 추출한 결과 4개만을 추출했다. 이 4개는 모두 정답 질의어이므로 정확률은 매우 높았다. 하지만, 3개의 정답 질의어를 추출하지 못했는데, 검색된 결과 문서를 살펴본 결과, 그 단어가 많이 사용되지는 않아, 검색 결과

〈표 3〉 정제 질의어 생성 예

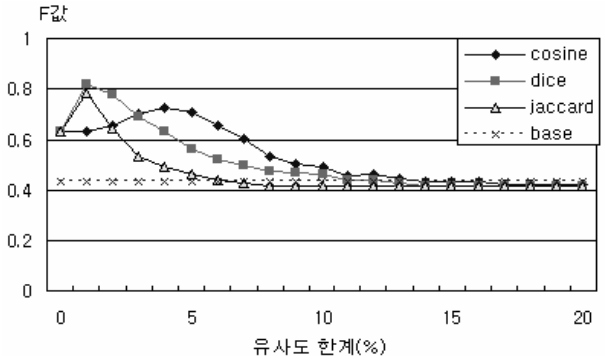
(Naver, dice, 윈도크기 17, 유사도 한계 0.01)

초기 이형태	정답 질의어	정제 질의어
라스베가스	*	*
라스베이가스	*	*
래스베가스	*	
라스베거스	*	*
라스베개스		
라스배가스	*	*
라스베각스		
락스베가스	*	
러스베가스	*	
라스비가스		

〈표 4〉 질의어 정제 처리 시간 평균(단위, 초)

(윈도크기 18, 한계값 0.04, dice, Google, 10개 질의어 확장)
(컴퓨터:HP A1129KR, Pentium 4, 3.06GH, OS:freeBSD 6.3, LAN:100Mbps)

질의어 확장	질의어 정제	총 처리시간
0.515	4.256	4.771



(그림 8) 유사도 한계값 변화에 따른 성능 변화: Naver(윈도크기 17)

문서로서의 중요성도 낮아 보였다. 따라서 제안된 방법이 비교적 정확하게 관련 이형태 외래어를 추출해 냄을 알 수 있었다.

대개 메타 검색 사이트는 다른 검색 엔진 사이트의 결과를 받아와 처리한 다음 그 결과를 보여주기 때문에 일반 검색 사이트보다 속도가 느리다. <표 4>는 본 실험에서 사용된 질의어 47개에 대한 평균 처리 시간을 서버측에서 측정한 것이다(PHP 프로그램의 microtime 함수 사용). <표 4>에서 보는 바와 같이 원 질의어를 10개의 질의어로 확장하는데 걸리는 시간(1차 질의어 생성 시간)은 평균 0.515 초였고, 그 다음 단계로 확장된 10개의 질의어를 검색 사이트에 보내 그 결과를 분석하고 정제하는데 걸리는 시간은 평균 4.256 초가 걸렸다. 따라서 정제 질의어를 생성하는데 걸린 시간(2차 질의어 생성 시간)은 전체 처리 시간인 평균 4.771 초가 걸렸다. 현재는 시험용으로 구축하였으므로 이런 속도도 충분하지만, 대량의 질의어 처리를 위해서는 보다 좋은 성능의 서버가 필요하고, 프로그램을 더 최적화해야 하며, 다른 검색 사이트에 대한 접속 처리 속도를 향상시키기 위한 방법 등이 더 연구되고 개선되어야 할 것이다.

6. 결 론

본 논문에서는 하나의 외래어 질의로부터 그에 관련된 유사한 이형태 외래어를 자동 생성하고 정제하여 쉽게 관련 문서를 찾아 볼 수 있는 메타 검색 방법을 제안하였다. 이 방법은 우선 외래어 질의를 통계적 방법으로 확장(base 방법)한 후, 이를 검색 시스템에 미리 검색하여 그 결과가 있는 질의어만을 선택하고(count 방법), 또 그중에서도 원 질의어와 같은 문맥에서 사용한 질의어만을 선택하여 (제안한 방법) 사용자에게 제시함으로써 검색시 정확도를 높일 수 있도록 하였다. 확장된 외래어 질의의 성능을 평가하기 위해 두 개의 포탈 사이트를 대상으로 실제 실험한 결과, 평균 F값은 base 방법이 38%, count 방법이 56%, 본 논문에서 제안한 정제 방법이 81%로 기존 방법에 비해 매우 만족할만한 결과를 보였다.

참 고 문 헌

[1] Jeong, K., S. H. Myaeng, J. S. Lee and K.-S. Choi, "Automatic Identification and Back-Transliteration of Foreign Words for Information Retrieval," Information Processing and Management, Vol.35, No.4, pp.523-540, 1999.

[2] Lee, J. S., K. Choi, "English to Korean Statistical Transliteration for Information Retrieval," Computer Processing of Oriental Languages, Vol.12, No.1, pp. 17-37, 1998.

[3] 이재성, "다국어 정보검색을 위한 영한 음차 표기 및 복원 모델," 박사학위논문, 한국과학기술원, 1999.

[4] 이희승, 안병주, "한글 맞춤법 강의-고친판," 신구문화사, 1994.

[5] 이현복, "외래어 표기법 개정 시안의 문제점," 어학연구 15.1, pp.39-59, 1979.,

[6] SERI/KIST, "지능형 정보처리기의 개발에 관한 연구," 제1차년도 최종 보고서, 과학기술처, 1995.

[7] 강병주, 이재성, 최기선, "외국어 음차 표기의 음성적 유사도 비교 알고리즘," 정보과학회 논문지(B), 제26권, 제10호, pp.1237-1246, 1999.

[8] 강병주, "한국어 정보검색에서 외래어와 영어로 인한 단어 불일치문제의 해결," 박사학위논문, 한국과학기술원, 2001.

[9] Cheon, S. M. "Construction of English Loanwords Contents for the Development of Educational Tools: a Step Towards the Prosperity of CALL Courseware," Ph. D dissertation. Hankuk University of Foreign Studies, 2005.

[10] Mettler, M. "TRW Japanese Fast Data Finder," TIPSTER Text Program Phase I Proc., Sep., pp.113-116, 1993.

[11] 김병혜, "영어단어의 알파벳표기로부터 한글표기로의 자동변환," 석사학위논문, 서강대학교 공공정책대학원, 1991.

[12] 이재성, "효과적인 외래어 이형태 생성을 위한 확률 문맥의존 치환 방법," 한국 콘텐츠학회논문지, 제7권, 제2호, pp. 73-83, 2007.

[13] Aslam, J. A., M. Montague, "Models for metasearch," Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 276-284, New Orleans, USA, 2001.

[14] Salton, G., "Automatic Text Processing - The transformation, analysis, and retrieval of information by computer," pp.318-319, Addison Wesley Publishing Company, 1989.

[15] Google, http://www.google.co.kr/.

[16] Naver, http://www.naver.com/.

[17] Manning, C., H. Schutze, "Foundations of Statistical Natural Language Processing," pp.268-269, The MIT Press, 1999.

이 재 성

e-mail : jasonl@cbu.ac.kr

1983년 2월 서울대학교 컴퓨터공학과(학사)
1985년 2월 한국과학기술원 전산학과(석사)
1999년 2월 한국과학기술원 전산학과(박사)
1985년～1988년 큐닉스컴퓨터 개발부 과장
1988년～1989년 미국 마이크로소프트 개발부 software design engineer
1989년～1993년 마이크로소프트 개발부 차장
1999년～2000년 한국전자통신연구원 선임연구원/팀장
2005년 7월～2006년 6월 University of Arizona, research scholar
2000년 9월～현재 충북대학교 컴퓨터교육과 부교수
관심분야 : 정보검색, 자연언어 처리, 컴퓨터교육