바이그램이 문서범주화 성능에 미치는 영향에 관한 연구

이 찬 도* · 최 준 영**

A Study on the Effectiveness of Bigrams in Text Categorization

Chan-Do Lee* · Joon-Young Choi**

Abstract

Text categorization systems generally use single words (unigrams) as features. A deceptively simple algorithm for improving text categorization is investigated here, an idea previously shown not to work. It is to identify useful word pairs (bigrams) made up of adjacent unigrams. The bigrams it found, while small in numbers, can substantially raise the quality of feature sets. The algorithm was tested on two pre-classified datasets, Reuters-21578 for English and Korea-web for Korean. The results show that the algorithm was successful in extracting high quality bigrams and increased the quality of overall features. To find out the role of bigrams, we trained the Naïve Bayes classifiers using both unigrams and bigrams as features. The results show that recall values were higher than those of unigrams alone. Break-even points and F1 values improved in most documents, especially when documents were classified along the large classes. In Reuters-21578 break-even points increased by 2.1%, with the highest at 18.8%, and F1 improved by 1.5%, with the highest at 3.2%. In Korea-web break-even points increased by 1.0%, with the highest at 4.5%, and F1 improved by 0.4%, with the highest at 4.2%. We can conclude that text classification using unigrams and bigrams together is more efficient than using only unigrams.

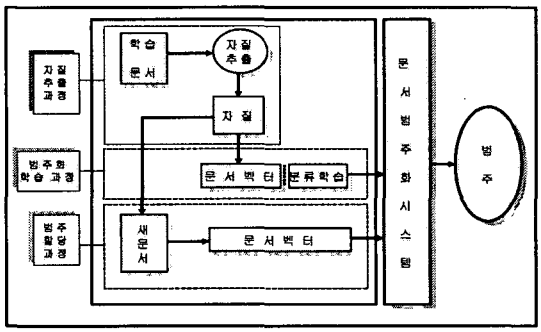
Keywords : Automated Text Categorization, Text Classification, Machine Learning, Bigram Algorithm

논문접수일 : 2004년 2월 23일 논문게재확정일 : 2005년 5월 20일
※ 본 연구는 한국과학재단 목적기초연구(R05-2002-000-00751-0) 지원으로 수행되었음.
* 교신저자, 대전대학교 정보통신인터넷공학부, (300-716)대전시 동구 용운동 96-3, Tel : 042-280-2551, e-mail : cdlee@dju.ac.kr
** 대전대학교 혜화의료원

1. 서 론

정보의 바다라고 일컬을 정도로 과거와는 비교할 수 없을 만큼 많은 정보가 인터넷에 쏟아져 나오고 있는 지금 신속하고 정확하게 정보를 찾아 사용하는 일은 쉽지 않아 졌다. 효율적인 정보검색 기술에 덧붙여 정보 자체에 대한 체계적인 분류와 관리가 이루어진다면, 좀 더 효율적이고 효과적인 정보 서비스를 제공할 수 있을 것이다. 정보를 체계적으로 분류하기 위해서는 인터넷 검색 엔진 야후나, 라이코스 등에서 사용하는 방법처럼 체계적인 범주를 설정하고 이 범주에 해당문서를 분류해 넣어야 한다. 이렇게 잘 정리된 범주를 가지고 있을 때 사용자는 자신이 원하는 정보를 손쉽게 찾아 볼 수 있고, 이 정보를 바탕으로 더 나아가 정보 창조 활동도 할 수 있다. 요컨대, 인터넷의 폭발적인 확장에 따라 문서를 포함한 콘텐츠 또한 기하급수적으로 늘고 있으며, 효과적인 정보관리 및 검색을 위해서는 내용별 분류작업이 필요하다. 그러나 이 범주화 작업을 수작업으로 처리할 경우에는 막대한 경비가 들고 일관성이 결여될 가능성이 높으므로, 컴퓨터가 자동으로 분류, 관리하는 자동문서범주화가 필요하다. 자동문서범주화 기술의 응용분야는 문서검색을 위한 인덱싱, 텍스트에서 특정 내용 추출, 웹페이지 분류, 메일의 필터링 등 다양하며, 기술을 향상시키기 위하여 많은 연구가 진행되어 지고 있다[Sebastiani, 2002].

과거 수년간의 연구결과는 괄목할 만한데, 예를 들면 Apté, Damarau, and Weiss[1998]는 Reuters-21578 데이터를 사용한 실험에서, 87.8%의 break-even point값을 보고하고 있다. 최근의 일반적인 연구동향은 핵심어 기반(keyword-based) 기술인데, 대부분의 연구자들은 문서를 단어의 꾸러미(bag of words : BOW)로 보고, 한 문서에 나오는 모든 단어 중 일정기준을 거친 핵심어의 유무로 그 문서를 분류한다. 많은 연구자들(예를 들면 [Apté, Damarau and Weiss, 1998 ; Lewis, 1992]) 이 단어보다 더 의미가 많다고 여겨지는 구(phrases)를 사용하여 범주화를 행할 경우 성능이 감소한다는 점을 지적하고 있다. 그럼에도 불구하고 단어 이상의 자질을 사용하여 범주화 성능을 향상시키고자 하는 노력은 계속 진행되고 있다. 본 연구에서는 나란히 나타나는 단어쌍(바이그램 : bi-gram)중 일정기준을 만족하는 고품질 바이그램 추출알고리즘을 제안하고, 추출된 바이그램을 단어에 추가하여 범주화를 행하였을 때 성능의 향상을 가져옴을 보여준다.



〈그림 1〉 자동문서범주화 과정

<그림 1>에서 보는 바와 같이 자동문서범주화 과정은 크게 자질추출과정, 범주화 학습과정, 범주할당과정으로 나뉜다. 자질추출과정에서는 문서에 나타난 단어 중 기계학습에 필요한 자질들만을 추출하는데 이를 위해서는 term frequency, document frequency, information gain 등의 수치가 사용된다. Term frequency는 한 문서 내에서 어떤 단어의 출현 빈도수를 나타내며, document frequency는 해당 단어가 한번이라도 출현한 문서의 수를 나타낸다. Information gain은 자질의 중요도를 측정하는데 널리 사용되는 기준으로서, 해당 단어를 선택했을 때 감소되는 entropy의 양을 말한다. 범주화 학습과정에서는 이러한 자질을 이용해 이미 분류된 문서를 범주화하는 분류기(classifier)

를 훈련시키게 되는데, Naïve Bayes, maximum entropy, neural network 등의 방법을 사용한다. 이렇게 해서 얻어진 문서범주화 시스템을 이용하여 새로운 문서를 범주화하는데 사용한다. 자동문서범주화의 세 단계 중 본 논문에서는 자질추출과정에 중점을 두고, 어떻게 자질을 추출했을 때 시스템의 성능을 향상시킬 수 있는가를 연구하고자 한다.

2. 구(phrases)를 사용한 문서범주화

문서범주화에 관련된 대부분의 연구는 핵심어를 기반으로 한 연구이며, 구를 사용한 문서범주화가 단어보다 더 나쁜 결과를 가져 왔다는 연구보고도 있다[Apté, Damarau, and Weiss, 1998 ; Lewis, 1992]. 구를 사용하면 의미의 애매성은 줄일 수 있지만(예를 들어 "computer"나 "science" 단독을 핵심어로 사용하는 것보다 "computer + science"를 사용하는 경우 그 문서가 전산학 관련 문서임을 쉽게 결정할 수 있다) 탐색공간의 확장, 빈도의 축소, 동의성의 증가 등 부정적인 측면이 더 강하게 작용하기 때문이다. 여러 연구자들이 위에 든 문제점들을 해결하기 위하여 노력해오고 있는데, n-gram(단어가 n개 모인 자질)을 BOW에 추가 할 때 향상된 성능을 가져왔음을 보여주는 희망적인 연구결과들이 보고되고 있다[Fürnkranz, 1998 ; Mladenić and Grobelnik, 1998 ; Schapire, Singer, and Singhal, 1998 ; Schütze, Hull, and Pederson, 1995].

Lewis는 "두 개 이상의 단어가 존재하는 자질" [Lewis, 1992, p.35]인 구를 사용한 문서범주화를 행하고 심도 있게 분석을 하였다. 그는 두 번 이상 나타나는 모든 명사구를 자질로 사용하였는데, 구가 단어보다 더 나쁜 결과를 가져왔음을 보고하고 있다. 구를 사용하면 애매성은 줄일 수 있지만, 탐색공간의 확장 등 단점이 많기 때문이다. 문서범주화에 사용하는 자질의 조건은 다음과 같다 :

1. 가능하면 숫자가 적을수록 좋다.
2. 골고루 분포되어야 한다.
3. 중복은 가능하면 피한다.
4. 잡음이 적어야 한다.
5. 애매성이 없어야 한다.
6. 속하는 범주와 관련이 많아야 한다.

구를 사용하면 5번 조건은 잘 해결할 수 있다. 그러나 1~4번 조건에는 오히려 악영향을 미친다. 첫째, 숫자가 늘어난다. 예를 들어 n개의 단어가 있다면 최대 n^2개의 구가 생길 수 있다. 둘째, n^2개의 구 대부분은 모든 문서나, 또는 거의 모든 문서에 대해 0값을 갖는다. 즉, 몇 개 문서에 편중된 분포를 보인다. 셋째, 구는 높은 중복도를 보인다. 만약 어떤 구에 속하는 개별 단어가 각각 k개의 동의어를 갖고 있다면, k^2개의 구가 같은 의미를 가질 수 있다. 넷째, 구는 잡음이 많은 성질을 보인다.

Mladenić and Grobelnik[1998]은 새로운 형태의 BOW를 이용하여 문서를 분류하는 연구를 하였다. 새로운 형태의 지식기반 시스템은 단어열(word sequence)의 길이를 1에서 5까지 달리하여 실험을 하였다. Term frequency가 높은 자질벡터를 사용하여 Naïve Bayesian 분류기를 훈련하였으며, 데이터로는 Yahoo 문서들을 사용하였다. 연구의 결과는 단어열의 길이가 3일 때까지는 핵심어만 사용했을 경우보다 더 좋은 결과 값을 보였으며, 그보다 긴 단어열은 성능향상에 별로 기여하지 못했음을 보여주었다.

Fürnkranz[1998]의 연구도 Mladenić and Grobelnik [1998]와 비슷한 결과를 보여준다. 자질은 term frequency와 document frequency를 사용하여 추출하였으며, 연구 결과는 단어열의 길이가 2 또는 3일 때 가장 유용하고, 더 긴 단어열을 사용하는 것

은 오히려 성능을 저하시킴을 보여준다.

Schütze, Hull and Pederson[1995]는 문서 라우팅 문제를 풀 때 계산비용과 과대적합(overfitting)을 줄이기 위하여 공간축소(dimensionality reduction) 방법을 사용하였다. Term frequency에 따라 추출된 단어와 2개의 단어로 이루어진 구를 사용하여 실험한 결과 향상된 성능을 나타냈음을 보여준다.

Schapire, Singer, and Singhal[1998]은 Rocchio 알고리즘을 이용한 문서 필터링 문제에 단어와 구를 사용하였다. Term frequency를 근거로 구를 선택하였으나, 어떤 것이 구에 해당하는지에 관한 명확한 정의가 주어지지 않아서 단어열의 길이가 얼마나 되는지 알 수 없다.

국내에서도 자동문서범주화 연구가 진행되어지고 있는데 한글문서를 자동으로 범주화하는 데에는 한글이 내포하고 있는 고유의 문제점들이 있다. 예를 들면, 핵심어를 자질로 추출하기 위해서는 문서의 내용과 관련도가 적은 조사, 수사, 어미 등의 기능어(function word) 처리가 선행되어져야 한다.

장병규[1997]는 군집화된 연어를 문서 범주화의 자질로 이용하였는데, 이 연구에서 사용한 연어는 서로 동시에 더욱 많이 나타나는 단어들의 조합으로 정의하여 실제 사용되는 복합명사를 사용하였다.

강원석, 강현규[1999], 임종묵 외 3인[1999], 이경순, 최기선[1999 ; 2000]은 사전을 이용하여 자질의 문서범주화 성능을 향상시키는 방법을 사용하였다.

본 연구에서는 위에 든 관련연구들과 몇 가지 점에서 차별성과 독창성을 가지고 있는 자질추출 방법을 사용하여 문서범주화를 행한다. 첫째, 바이그램을 자질로 사용한다. 즉, 길이 2의 단어열인 나란히 나타나는 단어쌍을 자질에 추가한다. 둘째, 바이그램 선정기준으로서는 document frequency, term frequency 뿐만 아니라 information gain을 사용한다. Information gain은 문서에서의 출현 빈도뿐만 아니라 출현하지 않은 빈도까지 고려해서 각 범주에서의 자질 정보량을 계산하는데, 이러한 특징으로 인해 문서 범주화 기법에서 좋은 성능을 보인다. 이렇게 여러 개의 바이그램 선정기준을 적용함으로써 잡음이 적고 범주를 잘 구별해 낼 수 있는 바이그램을 추출 할 수 있다. 셋째, 범주화학습과 범주할당 과정에서 개별 단어를 바이그램으로 대치하는 것이 아니라 개별 단어에 바이그램을 추가함으로 바이그램이 가지고 있는 의미 애매성 해소 능력을 활용한다. 넷째, 탐색공간을 줄이기 위해 추가하는 바이그램수를 전체 단어수의 2% 이내로 엄격히 선정한다.

본 연구의 목적은 위와 같은 기준으로 엄격히 선정된 바이그램을 단어에 추가하여 문서범주화를 행함으로써 성능을 향상시키고자 하는데 있다. 본 연구에서 사용한 방법은 단어만을 사용했을 때 생기는 애매성을 해소하며, Lewis[1992]가 지적한 구가 가져오는 단점인 1~4번 조건을 완화시킬 수 있다.

3. 바이그램 추출 알고리즘

본 연구에서 사용한 알고리즘의 기본 아이디어는 한 카테고리에서라도 최소한의 document frequency를 갖는 단어를 찾아 이를 모아 바이그램으로 만든다는 것이다. 알고리즘을 설명하면, 먼저 어느 정도 여러 문서에 나타나는 단어들을 seed로 하여 이들 단어가 훈련문서에 한번이라도 나타나는 바이그램을 추출한 후 빈도수(document frequency, term frequency)와 information gain이 높은 바이그램을 찾아내어 자질에 추가하여 범주화를 수행한다. 이렇게 얻어진 바이그램은 개별 단어를 대신하는 것이 아니라, 강화해주는 역할을 한다.

Information gain을 정의하기 위해서는 먼저 문서집합에 나타나는 문서들의 동질성을 나타내는

entropy를 정의하여야 한다. 어떤 문서가 n개의 서로 다른 범주에 속할 수 있을 때, 문서집합(S)의 entropy는

$$Entropy(S) = \sum_{i=1}^{n} -p_i log_2 p_i$$

와 같이 정의된다. 여기에서 p_i는 범주 i에 속하는 문서의 비율이다. 어떤 속성 A가 문서집합 S를 나눌 때 얻어지는 information gain은

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

와 같이 정의된다. 여기에서 *Values(A)*는 속성 A가 가질 수 있는 가능한 모든 값을 뜻하고, S_v는 A의 값이 v인 S의 부분 집합이다.

<그림 2>는 바이그램 추출 알고리즘의 의사코드이다. 알고리즘에 사용된 파라미터는 다음과 같다 :

1. *df_seed* : 단어가 seed로 사용되기 위한 document frequency의 최소수 조정
2. *df_thresh* : 바이그램 중 document frequency가 작은 수를 제거하기 위해 조정
3. *tf_thresh* : 바이그램 중 term frequency가 이보다 작은 수는 제거
4. *ig_thresh* : 바이그램 중 information gain이 이보다 작은 수는 제거

```
Find S = {x | x의 빈도수 > 총 문서의 수 * df_seed}  // seed 집합
Set B = {}.                                          // 바이그램 집합
For each 훈련문서 A
{
        For each 인접 단어 (w1, w2) in A
         if (w1 ∈ S OR w2 ∈ S)   // w1이나 w2가 seed이면
         add bigram "w1+w2" to B.  // w1과 w2를 더하여 바이그램 후보자로 한다.
}
For each b in B
{
        For each 카테고리 c          // 기준에 미달되는 바이그램은 제거
         if (b의 빈도수 < 카테고리 c에 속하는 문서의 수 * df_thresh)
         OR (카테고리 c에 속하는 모든 문서에서, b의 빈도수 < tf_thresh)
                B의 집합에서 b를 제거한다.
         if (제거되지 않은 b의 infogain < ig_thresh)
                B의 집합에서 b를 제거한다.
}
B를 출력.   // 바이그램을 자질에 추가, 범주화 실행
```

〈그림 2〉 바이그램 추출 알고리즘

프로그램은 C로 구현하였으며, UNIX에서 실행하였다. 바이그램 추출 알고리즘의 성능을 좌우할 수 있는 여러 파라미터는 pilot study를 통하여 최적값을 결정하였다. *df_seed*는 0.01, *df_thresh*는 0.005, *tf_thresh*는 3, 그리고 *ig_thresh*는 총 단어수의 1%에 해당하는 단어가 갖는 information gain으로 정하였다. 최종적으로 추출된 바이그램 수는 총 단어수의 1.5%가 넘지 않도록 하였다.

4. 실 험

4.1 실험 데이터

본 연구에서 제안한 알고리즘이 데이터베이스에 상관없이 일관되게 높은 향상률을 보이는가, 특히

한글문서에도 적용되는 가를 보기 위하여, Reuters-21578.corpus와 Korea-web corpus를 사용하여 실험을 하였다.

Reuters-21578 corpus는 1987년에 나타난 Reuters 통신회사의 문서들을 모아 분류 해놓은 corpus인데 총 135 범주에 23460문서로 되어있으나, 이 실험에서는 일부 테스트 문서가 없는 범주를 제외한 나머지 93 범주 13343 문서(9592 학습문서, 3751 테스트문서)로 실험을 하였다. 모든 문서는 헤더 등의 불필요한 정보를 제외하고 본문만 유지하기 위하여 전처리를 하였다. 숫자와 구두점은 삭제하고, 대문자는 소문자로 변환하였으며, 불용어들(stopwords)도 모두 제거하였다.

한글문서 실험으로는 고려대, 호서대, 한남대에서 함께 만든 문서분류학습 및 테스트용 문서집합인 Korea-web corpus를 이용하였다. 이 corpus는 소분류(79), 중분류(19), 대분류(8)로 나누어져 있으며, 전체 문서 수는 4786개로서 3198 학습문서와 1588 테스트문서를 가지고 각 분류별로 실험을 진행하였다. 모든 문서는 우선 html 분석기를 사용하여 프로토콜 헤더, html 태그, " " 따위의 포맷명령어 등을 제거하였으며, 사전을 사용하여 숫자와 구두점, 불용어들도 모두 제거하였다. 이 경우 간단한 형태소 분석이 선행되었다.

4.2 실험 방법

우선 바이그램 추출 알고리즘을 실행하여 범주화의 성능 향상을 가져올 수 있는 고품질 바이그램을 추출하였다. 각 범주마다 실험은 2 회씩 수행되었는데, 먼저 단어만 사용하여 범주화 성능을 측정하였고, 다음으로 단어에 바이그램을 추가하여 범주화를 하였다. 본 연구에서 사용한 범주화는 어떤 문서가 그 범주에 속하는가 아닌가를 판정하는 2진 범주화 (binary categorization)이다. 주어진 학습문서를 대상으로 Naïve Bayes 분류기를 사용하여 학습하였으며, 테스트문서를 대상으로 성능을 측정하였다.

4.3 성능 평가

성능 평가를 위해서는 정확률(precision), 재현율(recall), break-even point와 F1을 사용하였다. 정확률은 시스템이 제시한 범주 중 정확한 범주의 비율을 의미하고, 재현율은 전체 범주 중 정확하게 제시한 범주의 비율을 의미한다. 정확률마다 그에 해당하는 재현율이 연관 지어 지는데, 일반적으로 정확률이 높으면 재현율이 낮다. 정확률과 재현율이 같은 점을 break-even point(이하 BEP)라 한다. F-measure는 정확률과 재현율 모두를 이용하여 하나의 척도로 평가하는 방법인데, 그 식은 다음과 같다.

$$F=\frac{(r^2+1)PR}{r^2P+R}$$

여기에서 r의 의미는 정확률(P)와 재현율(R)의 비중을 정하는 변수로써, $r > 1$이면 정확률의 비중을 높게 하고, $r < 1$이면 재현율의 비중을 높게 둔다는 의미이다. 일반적으로 r값으로는 1, 2, 5를 사용하는데, 본 연구에서는 1(F1)을 사용하였다. 평가치는 흔히 범주별로 계산한 후에 평균을 내는데, 범주별로 낸 값에 대한 평균은 macro-averaging, 전체를 한 범주로 간주하고 낸 평균은 micro-averaging이라 한다. Macro-averaging은 모든 범주에 동일한 가중치를 줌을 의미하고, micro-averaging은 모든 문서에 동일한 가중치를 부여함을 의미한다.

4.4 바이그램 추출 결과의 분석

제안한 알고리즘은 <표 1>에서 보는 바와 같이 "생활+정보", "벤처+기업" 등과 같이 명확한 개념을 나타내는 바이그램들을 성공적으로 추출하였다.

〈표 1〉 Korea-web에서 추출된 바이그램의 예

생활+정보	오늘+날씨	생활+문화	TCP+IP	벤처+기업
정보+HTTP	제목+검색	학위+논문	합숙+활동	자동차+물가

〈표 2〉 바이그램 추출 결과

자질 / corpus	전체 단어 수	바이그램 수	100위 내 바이그램 수	자질 향상도
Reuters-21578	40,000	531	45	46%
Korea-web	778,545	4,405	17	7%

또한 <표 2>에서 보는 바와 같이 바이그램의 수는 전체 단어 수에 비해 매우 적었지만 information gain 수치에 따라 정렬한 결과 상당수가 100위안에 들었다. 추출된 바이그램들을 분석한 결과 바이그램은 information gain을 증가시키고 따라서 전체 자질들의 질을 향상시킨다는 것을 확인할 수 있었다. 예를 들면 바이그램 "생활+정보"의 information gain (0.017)은 이 바이그램을 구성하고 있는 개별 단어 "생활"(0.015)이나 "정보"(0.011)보다 훨씬 높았다.

4.5 실험 결과 및 분석

바이그램을 단어에 추가하여 실험한 결과 <표 3>과 같이 단어만 사용한 경우보다 정확률은 다소 감소하였으나, 재현율은 증가하였다.

BEP과 F1 계산결과 역시 단어와 바이그램이 함께 사용될 때 향상 하였는데, <표 4>는 전체 문서에 대한 실험결과이다.

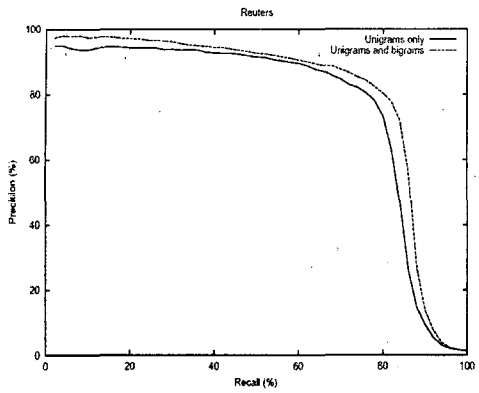
Reuters-21578과 Korea-web 전체문서에 대해 정확률-재현율 그래프를 그리면 <그림 3>과 같이 나타내어진다. <그림 3>에서 점선은 단어+바이그램의 정확률-재현율을 나타낸 것이고, 실선은 단어의 정확률-재현율을 나타낸 것인데 전반적으로 점선이 더 효율적임을 볼 수 있다.

〈표 3〉 정확률과 재현율

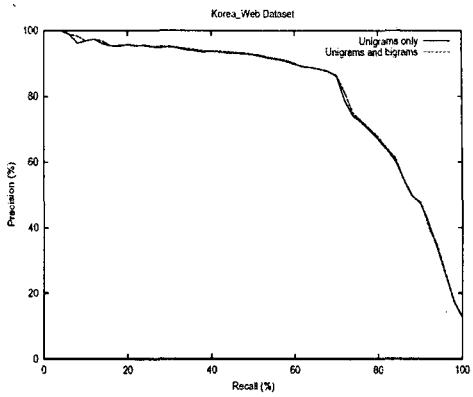
성능 / corpus	정확률		재현율	
	단어	단어+바이그램	단어	단어+바이그램
Reuters	71.5%	70.8%	81.0%	84.4%
Korea-web	81.3%	80.7%	71.1%	71.9%

〈표 4〉 Reuters와 Korea-web 전체문서에 대한 BEP과 F1

성능 / corpus	BEP			F1		
	단어	단어+바이그램	향상률(%)	단어	단어+바이그램	향상률(%)
Reuters	0.784	0.800	2.041	0.759	0.770	1.449
Korea-web	0.739	0.746	0.947	0.758	0.761	0.396

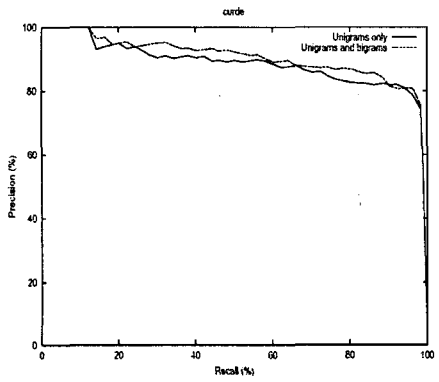


(a) Reuters-21578

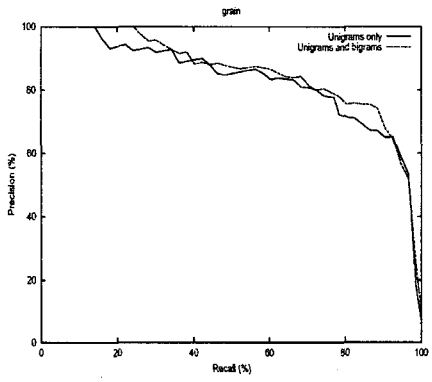


(b) Korea-web

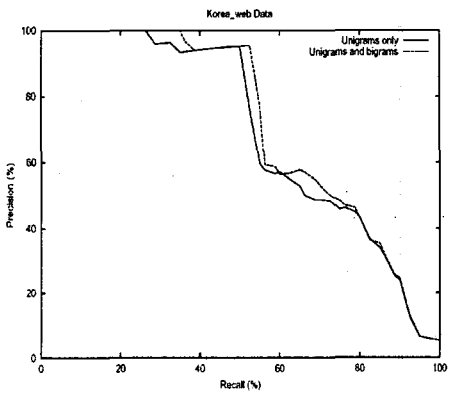
〈그림 3〉 정확률-재현율 그래프(전체)



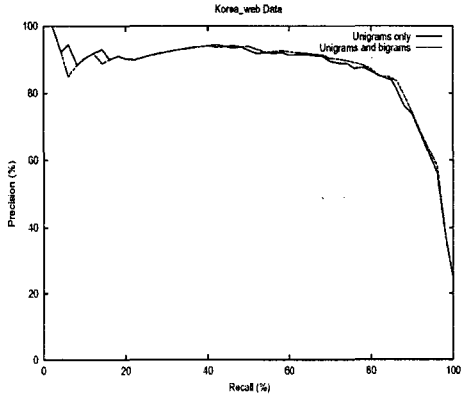
(a) Reuters-21578 crude



(b) Reuters-21578 grain



(c) Korea-web 교육



(d) Korea-web 건강과 의학

〈그림 4〉 범주별 정확률-재현율 그래프의 예

<그림 4>는 일부 범주의 정확률-재현율 그래프인데, Korea-web의 경우에도 Reuters-21578만큼 높지는 않지만 많은 향상을 나타냄을 볼 수 있다.

<표 5>는 Reuters-21578 데이터 중 문서수가 많은 상위 10개 범주에 대한 정확률, 재현율, BEP, F1을 보여주는데, 대부분의 경우 단어와 바이그램을 함께 사용하였을 때 향상한 것을 볼 수 있다.

<표 6>은 Korea-web 대분류 각 범주에 대한 실험결과인데, 전반적으로 향상되었음을 보여준다.

Korea-web에 대한 중분류와 소분류 실험결과는 <표 7>, <그림 5>와 같다.

〈표 5〉 Reuters-21578 상위 10개 범주에 대한 정확률, 재현율, BEP, F1

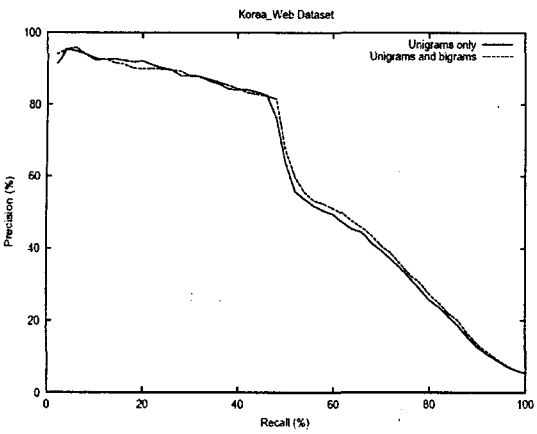
성능 / 범주	단 어				단어+바이그램				향상률(%)	
	정확률	재현율	BEP	F1	정확률	재현율	BEP	F1	BEP	F1
earn	0.906	0.982	0.969	0.943	0.961	0.972	0.970	0.967	0.103	2.546
acq	0.942	0.974	0.962	0.958	0.931	0.976	0.956	0.953	-0.624	-0.537
money-fx	0.548	0.977	0.674	0.702	0.564	0.983	0.752	0.716	11.573	2.005
grain	0.550	0.959	0.776	0.699	0.523	0.953	0.863	0.676	11.211	-3.300
crude	0.548	0.977	0.823	0.702	0.564	0.983	0.881	0.716	7.047	2.005
trade	0.386	0.871	0.638	0.535	0.391	0.940	0.758	0.552	18.809	3.236
interest	0.560	0.923	0.684	0.697	0.566	0.946	0.723	0.708	5.702	1.600
ship	0.752	0.887	0.793	0.814	0.738	0.887	0.838	0.806	5.675	-1.020
wheat	0.437	0.939	0.643	0.594	0.417	0.929	0.755	0.576	17.418	-3.056
corn	0.331	0.857	0.584	0.447	0.322	0.892	0.673	0.473	15.240	-0.769

〈표 6〉 Korea-web (대분류 전체 데이터)의 BEP과 F1

성능 / 범주	단 어		단어+바이그램		향상률(%)	
	BEP	F1	BEP	F1	BEP	F1
의학	0.845	0.844	0.849	0.849	0.473	0.610
공학	0.694	0.692	0.694	0.698	0.000	0.840
스포츠	0.763	0.751	0.763	0.754	0.000	0.369
사회	0.702	0.714	0.708	0.716	0.855	0.235
경제	0.723	0.744	0.723	0.745	0.000	0.126
교육	0.559	0.617	0.584	0.642	4.472	4.184
문화	0.753	0.761	0.757	0.752	0.531	-1.123
컴퓨터	0.840	0.811	0.835	0.809	-0.595	-0.251
macro 평균	0.735	0.758	0.746	0.761	0.947	0.396
micro 평균	0.722	0.742	0.726	0.746	0.554	0.624

〈표 7〉 Korea-web (중분류, 소분류 전체데이터)의 BEP과 F1

성능 / 분류별	단 어		단어+바이그램		향상률(%)	
	BEP	F1	BEP	F1	BEP	F1
중분류	0.537	0.562	0.546	0.569	1.676	1.246
소분류	0.168	0.178	0.195	0.205	15.902	14.878



〈그림 5〉 Korea-web 중분류의 정확률-재현율 그래프

<표 8>은 BEP의 향상률이 높은 Korea-web 중분류 상위 5개 범주에 대해 정확률, 재현율, BEP, F1을 보여준다.

<표 9>는 BEP의 향상률이 높은 Korea-web 소분류 상위 5개 범주에 대해 정확률, 재현율, BEP, F1을 보여준다.

대분류와 마찬가지로 중분류, 소분류 역시 바이그램을 이용한 값이 전반적으로 향상한 것을 볼 수 있다. <표 10>에서 볼 수 있듯이 대분류 범주 안에 속해있는 중분류 범주로, 중분류 범주 안에 속해있는 소분류 범주로 즉, 큰 주제를 갖는 범주에서 전문적인 주제(예 : 컴퓨터(대분류) -> 하드웨어(중분류) -> 인공지능_및_신경망(소분류))를 갖는 범주로 진행되어 가면서 BEP과 F1은 더 많은 향상을 보였다. 그러나 전반적인 정확률은 감소하였다. 이와 같은 이유는 문서를 분류하기 전에 사전 인식, 즉 학습 과정에서 데이터 수가 적어짐에 따라 학습 과정의 바이그램 수 역시 적어지기 때문에 정확도는 감소한 것으로 분석된다. 그러나 좀더 전문적인 주제를 갖는 범주로 실험을 할 경우 정확률은 감소하였지만 BEP과 F1의 향상률은 증가한다. 이와 같은 이유는 주제의 폭이 좀더 전문적으로 되어짐에 따라 발생하는 전문지식의 전문용어가 증가하기 때문이

〈표 8〉 Korea-web 중분류 상위 5개 범주에 대한 정확률, 재현율, BEP, F1

성능 / 범주	단 어				단어+바이그램				향상률 (%)	
	정확률	재현율	BEP	F1	정확률	재현율	BEP	F1	BEP	F1
하드웨어	0.364	0.267	0.295	0.308	0.391	0.300	0.361	0.340	22.373	10.378
공학	0.476	0.333	0.364	0.392	0.500	0.367	0.413	0.423	13.462	7.885
법률	0.304	0.175	0.198	0.222	0.320	0.200	0.222	0.246	12.121	10.769
사회과학	0.667	0.275	0.422	0.389	0.676	0.288	0.447	0.404	5.924	3.628
의학	0.754	0.430	0.547	0.548	0.758	0.470	0.577	0.580	5.484	5.929

〈표 9〉 Korea-web 소분류 상위 5개 범주에 대한 정확률, 재현율, BEP, F1

성능 / 범주	단 어				단어+바이그램				향상률 (%)	
	정확률	재현율	BEP	F1	정확률	재현율	BEP	F1	BEP	F1
연극_뮤지컬_공연장	0.083	0.050	0.050	0.063	0.154	0.100	0.150	0.121	200	93.939
인공지능_및_신경망	0.071	0.050	0.050	0.059	0.133	0.100	0.100	0.114	100	94.286
게임_도박	0.083	0.050	0.050	0.063	0.154	0.100	0.100	0.121	100	93.939
동물_애완동물_곤충	0.083	0.050	0.050	0.063	0.154	0.100	0.100	0.121	100	93.939
성인병	0.077	0.050	0.050	0.061	0.133	0.100	0.100	0.114	100	88.572

〈표 10〉 Korea-web 데이터의 대, 중, 소 범주별 향상률 예

성 능 / 범 주	단 어		단어+바이그램		향상률 (%)	
	BEP	F1	BEP	F1	BEP	F1
컴퓨터(대분류)	0.840	0.811	0.835	0.809	-0.595	-0.251
하드웨어(중분류)	0.295	0.308	0.361	0.340	22.373	10.378
인공지능_및_신경망(소분류)	0.050	0.059	0.100	0.114	100	94.286

다. 즉, 전문용어 같은 경우는 일반적인 단어보다 바이그램의 구로 나타냄으로서 그 뜻을 정확히 알 수 있기 때문에 의미의 애매성이 감소하여 향상률은 증가하게 된다.

5. 결 론

5.1 연구결과의 요약

본 연구에서는 일정기준을 만족하는 고품질 바이그램 추출알고리즘을 제안하고, 단어만을 사용한 경우보다 단어+바이그램을 사용한 자동문서범주화 시스템이 더 효율적이라는 것을 보여주었다. 바이그램의 성능을 측정한 결과 "생활+정보" 등과 같이 명확한 개념을 나타내는 바이그램들을 성공적으로 추출하였고, 바이그램의 수는 단어 수에 비해 적었지만 information gain 수치에 따라 나열한 결과 많은 수가 높은 순위를 차지하였다. 추출된 바이그램들을 분석한 결과 information gain 값이 증가함을 발견할 수 있었다. 즉 전체 자질들의 질을 향상시킨다는 것을 알 수 있었다. 바이그램을 자질에 추가하여 Naïve Bayes 분류기를 학습시킨 결과 단어만 사용한 경우보다 recall값이 증가하였다. BEP과 F1 계산 결과 역시 많은 문서에 대하여 단어와 바이그램을 함께 사용할 때 성능이 향상하였다. 대분류에서 소분류로 즉, 큰 주제를 갖는 범주에서 전문적인 주제를 갖는 범주로 진행되어 가면서 BEP과 F1은 더 많은 향상을 보였다.

5.2 기존연구 결과와의 비교

본 연구에서는 Naïve Bayes 분류기를 이용하여 단어만을 사용하여 범주화를 행할 때보다 바이그램을 추가할 때 성능이 향상함을 보여주었다. 본 연구에서 사용한 방법이 maximum entropy나 neural network 등의 다른 방법을 사용할 때에도 타당한지는 추후 연구해야할 과제이다. Korea-web을 이용한 문서범주화에 대한 기존연구는 찾을 수가 없어 비교할 수 없었다. Reuters-21578의 기존연구와 비교할 때 본 연구에서 얻어진 BEP 값인 0.800은 전체비교대상 22개 연구결과 중 14위로서 중간정도에 속했다[Sebastiani, 2002, Table 6, #4 참조].

5.3 성능저하의 분석

한글문서를 분류별로 분석한 결과 정확률은 감소하였다. 그 이유는 실험 데이터 문서수가 대분류에서 소분류로 갈수록 적어지기 때문이다. Reuters-21578 같은 경우도 학습 문서의 수가 적으면 정확률이 대체적으로 떨어지는 것을 볼 수 있다. 따라서 문서 범주화 연구의 발전을 위해서는 일반성을 가질 수 있고 좀더 정확한 범주를 갖는 많은 학습 문서 및 테스트 문서를 토대로 한 corpus가 구축되어야 할 것이다.

한글 문서의 경우 바이그램을 추가했을 때 영어 문서의 경우보다 향상률이 낮은 이유는 한글이 갖고 있는 특성에 기인한다. 하나의 형태소가 하나의

단어를 이루는 굴절어인 영어와 달리 교착어인 한국어는 한 어절이 하나 이상의 형태소로 이루어져 있기 때문에 색인어를 추출하기 위해서는 형태소 분석이 필수적이다. 또한 한국어의 복합명사는 띄어쓰기도 하고 붙여 쓰기도 하므로 문서의 색인에서 형태의 불일치로 인해 문제를 야기한다. 아울러 단어를 자질로 추출하기 위해서는 문서의 내용과 관련도가 적은 조사, 수사, 어미 등의 불용어 처리가 선행되어져야 한다. 따라서 단어추출과정에서 한국어의 특성상 형태소 분석, 복합명사 인식 및 분해, 불용어 제거 등에서 미흡한 점이 있어 많은 향상을 보이지 않고 있다. 이러한 문제점을 보완한다면 본 논문에서 제안한 알고리즘을 사용하여 바이그램을 추출하고 기존의 단어에 추가하여 한글문서 범주화를 행할 때 더 많은 향상률을 가져올 수 있을 것으로 기대하며, 이 부분은 향후 계속 연구되어야 할 과제이다.

참 고 문 헌

[1] 강원석, 강현규, "시소러스 도구를 이용한 실시간 개념 기반 문서 분류 시스템", *정보과학회논문지 (B)*, 제26권 제1호, 1999, pp. 167-176.

[2] 이경순, 최기선, "문서분류에서 의미영역지식에 기반한 문서표현", *제 11회 한글 및 한국어 정보처리 학술대회*, 1999, pp. 79-84.

[3] 이경순, 최기선, "사전간 계층관계를 이용한 전문용어 자동 추출 기법", *2000년 한국인지과학회 춘계 학술대회*, 2000, pp. 131-136.

[4] 임종묵, 오효정, 맹성현, 이만호, "카테고리 정보 활용을 통한 링크 기반 검색의 속도 향상", *한국정보과학회 봄 학술발표논문집*, 1999, pp. 324-326.

[5] 장병규, *문서 범주화에서 연어를 기반으로 한 문서 표현*, 1997, 한국과학기술원 석사 학위 논문.

[6] Apté, C., Damerau, F. and Weiss, S., "Automated learning of decision rules for text categorization", *ACM Transactions on Information Systems*, 12(3), 1994, pp. 233- 251.

[7] Apté, C., Damerau, F. and Weiss, S., "Text Mining with Decision Trees and Decision Rules", Presented at *the Conference on Automated Learning and Discover*, 1998.

[8] Fürnkranz, J., *A Study Using n-gram features for Text Categorization*, Technical Report OEFAI-TR-98-30, Austrian Research Institute for Artificial Intelligence, Vienna, Austria, 1998.

[9] Lewis, D., *Representation and Learning in Information Retrieval*, Technical Report UM-CS-1991-093, Department of Computer Science, University of Massachusetts, Amherst, MA, 1992.

[10] Mladenić, D. and Grobelnik, M., "Word sequences as features in text learning", *Proceedings of the 17th Electrotechnical and Computer Science Conference (ERK-98)*, 1998, pp. 145-148.

[11] Schapire, R, Singer, Y. and Singhal, A., "Boosting and Rocchio Applied to Text Filtering", *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval, 1998*, pp. 215-223.

[12] Schütze, H., Hull, D. and Pederson, J., "A Comparison of Classifiers and Document Representations for the Routing Problem", *Proceedings of SIGIR-95, 15th ACM International Conference on Research and Development in Information Retrieval*, 1995, pp. 229-237.

[13] Sebastiani, F., "Machine learning in automated text categorization", *ACM Computing Surveys*, 34(1), 2002, pp. 1-47.

□ 저자소개

이 찬 도

저자는 서울대학교(B.A.)와 Arizona State University (M.A.)에서 독일어를 전공하였으며, Indiana University에서 Computer Science 전공으로 M.S.와 Ph.D.를 취득하였다. KAIST 인공지능연구센터에서 연수연구원으로 근무하였으며, 현재 대전대학교 정보통신인터넷공학부 교수로 재직 중이다. 최근에 University of California, Santa Barbara에서 방문연구를 수행하였다. 주요 연구 관심분야는 인공지능(자연어 처리, 지능형 에이전트, 문서 범주화), 소프트 컴퓨팅, 하이퍼 문학 등이다.

최 준 영

저자는 대전대학교 정보통신공학과에서 학사 및 석사학위를 취득하였으며, 현재 대전대학교 혜화의료원에 재직 중이다. 주요 연구 관심분야는 지능형 에이전트, 문서 범주화 등이다.

◈ 이 논문은 2004년 2월 23일 접수하여 2차 수정을 거쳐 2005년 5월 20일 게재확정되었습니다.