오류 데이타에 강한 자질 투영법 기반의 문서 범주화 기법

(Text Classification based on a Feature Projection Technique with Robustness from Noisy Data)

고 영 중 † 서 정 연 ††
(Youngjoong Ko) (Jungyun Seo)

요 약 본 논문은 자질 투영법을 사용한 새로운 문서 분류기를 제안한다. 제안된 문서 분류기는 학습 문서를 각 자질로의 투영으로써 표현한다. 문서를 위한 분류 작업은 투영된 각 자질로부터의 투표(voting)에 기인한다. 실험을 통해서 본 제안된 문서 분류기는 단순한 구조에도 불구하고 높은 성능을 보이고 있으며, 특히 기존의 문서 범주화 기법에서 높은 성능을 보여왔던 최근린법(k-NN)과 지지벡터기계(SVM)와 비교했을 때 빠른 수행 속도와 오류 데이타가 많을 환경에서 높은 성능을 보인다는 장점이 있다. 또한 제안된 문서 분류기의 알고리즘이 매우 단순하기 때문에 분류기의 구현과 학습 과정이 쉽게 수행될 수 있다. 이러한 이유로 제안된 문서 분류기는 빠른 수행 속도와 견고성(robustness), 그리고 높은 성능을 요구하는 문서 범주화 응용 영역에 유용하게 사용될 수 있을 것이다.

키워드 : 문서 범주화, 자질 투영법, 문서 분류기

Abstract *This paper presents a new text classifier based on a feature projection technique. In feature projections, training documents are represented as the projections on each feature. A classification process is based on individual feature projections. The final classification is determined by the sum from the individual classification of each feature. In our experiments, the proposed classifier showed high performance. Especially, it have fast execution speed and robustness with noisy data in comparison with k-NN and SVM, which are among the state-of-art text classifiers. Since the algorithm of the proposed classifier is very simple, its implementation and training process can be done very simply. Therefore, it can be a useful classifier in text classification tasks which need fast execution speed, robustness, and high performance.*

Key words : Text Classification, Feature Projections, Text Classifier

1. 서 론

인터넷이 폭 넓게 보급되어 온라인(on-line)상에서 얻을 수 있는 텍스트(text) 정보의 양이 급증함에 따라 효율적인 정보 관리 및 검색이 요구되고 있으며, 이를 위한 기법으로 자동 문서 범주화(automatic text categorization)가 중요하게 사용되고 있다. 자동 문서 범주화는 미리 정의된 범주(category)에 문서를 자동으로 할당하는 기법과 관련된 연구분야로서, 대량의 문서의 효율적인 관리 및 검색을 가능하게 하는 동시에 방대한 양의 수 작업을 감소시키는 데 그 목적이 있다.

많은 지도 학습 기반의 기계 학습 알고리즘들이 문서 범주화에 적용되었는데, 그 예로는 베이지언 확률 모델(bayesian probabilistic approach)[1,2], 결정트리(decision tree)[3], 지지 벡터 기계(SVM : Supprot Vector Machine)[4,5], 최근린법(k-nearest neighbors classifier)[6], 선형 모델(linear model)[7], 신경망(neural networks)[8] 등이 있다.

이들 알고리즘 중에 최근린법(k-NN)과 지지 벡터 기계(SVM)가 문서 범주화 영역에서 가장 좋은 성능을 보이는 것으로 보고되어 왔다[9]. 하지만 이들 알고리즘은 높은 성능을 보이고는 있지만 수행 속도와 오류 데이타로부터의 견고하지 못하다는 약점들을 가지고 있다. 특

· 본 연구는 한국과학재단 목적기초연구(R01-2003-000-11588-0) 지원과 서강대학교 산업기술연구소 후원으로 수행되었음
† 정 회 원 : 서강대학교 산업기술연구소 연구원
kyj@nlpzodiac.sogang.ac.kr
†† 종신회원 : 서강대학교 교수
seojy@ccs.sogang.ac.kr
논문접수 : 2003년 8월 4일
심사완료 : 2003년 12월 26일

히, k-NN은 수행속도가 너무 느리고, SVM은 오류데이타가 많은 학습 데이타로부터 학습되었을 때 성능이 많이 낮아지는 경향을 보인다. 그러므로, 본 논문에서는 k-NN과 SVM의 약점들을 보완하고 비슷한 성능을 보이는 새로운 형태의 문서 분류기를 제안한다.

제안된 문서 분류기는 자질 투영법(feature projection)을 사용하여 학습 문서를 각 자질로의 투영으로써 표현한다. 각 문서에 대한 분류 작업은 투영된 각 자질로부터의 투표(voting)에 기인하며, 최종적인 문서 분류는 각 자질의 분류결과의 합으로 결정된다. 실험을 통해 살펴본 제안된 문서 분류기는 간단한 구조임에도 불구하고 높은 성능을 보이고 있으며 k-NN과 SVM에 비교했을 때 빠른 수행속도와 오류데이타로부터의 견고함을 보이고 있다.

본 논문의 구조는 다음과 같다. 2장에서는 자질 투영법을 사용하는 관련 연구에 대해서 기술하고, 3장에서는 제안된 문서 분류 기법에 대해 자세히 설명한다. 4장에서는 실험을 통해 제안된 문서 분류 기법의 성능을 평가하고 분석하며, 마지막으로 결론을 내리고 향후 연구에 대해서 기술한다.

2. 관련 연구

자질 투영법을 패턴 분류에 사용하기 위한 연구가 기계 학습 분야에서 연구되었다[10,11]. Akkus와 Gunivenir는 [11]에서 자질 투영법을 k-NN 분류기에 적용하였으며 새로운 알고리즘을 k-NNFP(k-NN on Feature Projections)라고 명명하였다. k-NNFP 알고리즘은 먼저 각 자질로 학습 문서들을 그림 1과 같이 투영시키고, 각 자질 별로 실험 문서의 자질 값과 가장 비슷한 k개의 값(k nearest neighbors)을 찾아내서 그들의 투표(voting)에 의해서 분류를 수행한다. 각 자질에서의 가장 비슷한 k개의 값을 찾기 위해서는 다음 식 (1)을 사용하여 계산된 거리값(distance)을 사용하고, 이 식에 의해 선택된 k개의 자질들은 각 자질이 포함되어 있던 학습 문서의 범주로 투표를 수행한다. 그러므로, 자질의 수가 n개 있다면 총 $n \times k$개의 투표값을 이용해 문서를 분류한다. 이에 반해, k-NN은 단지 실험 문서와 가장 비슷한 k개의 학습 문서들의 투표값을 사용한다.

$$dist_m(t_m(i), t_m(j)) = |w(t_m, \vec{d}_i) - w(t_m, \vec{d}_j)| \quad (1)$$

위의 식 (1)에서 $t_m(i)$는 문서 $\vec{d}_i$의 m번째 자질 t를 표현한다.

실험을 통해 살펴본 k-NNFP의 성능은 기존의 k-NN보다 수행속도 면에서 많은 향상을 얻을 수 있었고, 또한 성능 면에서도 k-NN과 거의 비슷함을 보였다.

하지만, 이러한 자질 투영법은 문서 범주화에는 아직 적용되지 않았으며, 기존의 연구[11]에서 사용한 UCI 실험 데이타와는 달리, 문서 분류 데이타는 많은 다른 특성들을 가지고 있다. 예를 들어, 문서 집합은 다른 분류 집합들보다 많은 자질들을 가지고 있으며, 또한 각 자질 별로 문서에 나타난 빈도가 전부 다르므로 기존의 연구와 똑 같은 방법을 적용하기에는 어려움이 따른다. 그러므로, 본 논문에서는 자질 투영법을 문서 분류에 사용하기 위한 새로운 형태의 알고리즘을 제시하고 성능을 평가한다.

3. 자질 투영법을 기반으로 한 문서 범주화 시스템

3.1 자질 투영법을 이용한 학습 집합의 표현

문서를 벡터공간에 표현하기 위해서, 정보 검색에서 일반적으로 사용되는 TF-IDF기법을 이용하여 각 문서 벡터의 자질 가중치를 계산하고, 각 문서는 각 자질 가중치의 벡터로 다음과 같이 표현될 수 있다[12].

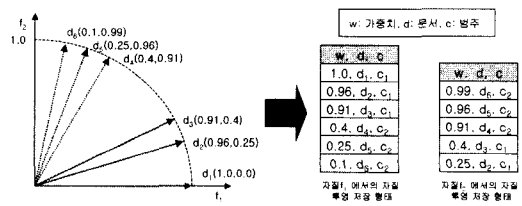
$$\vec{d} = \langle w(t_1, \vec{d}), w(t_2, \vec{d}), \ldots, w(t_n, \vec{d}) \rangle \quad (2)$$

식 (2)에서 d는 문서를, t는 그 문서에 포함되어 있는 자질(단어)을 표현한다. 여기서 각 자질의 가중치 값은 다음의 식 (3)로 계산된다.

$$w(t, \vec{d}) = \frac{(1 + \log tf(t, \vec{d})) \times \log(N/n_t)}{\|\vec{d}\|} \quad (3)$$

여기서 $tf(t, \vec{d})$는 자질 빈도수(term frequency)를 나타내고 $\log(N/n_t)$는 역문헌 빈도수(inverted document frequency)를 나타낸다. N은 학습 집합의 총 문서 수이고, n_t는 자질 t를 포함하고 있는 학습 문서의 수이다.

자질 투영법을 이용하여 학습 집합에서의 문서를 표현하기 위해서는 이렇게 벡터공간에 표현되어 있는 각 문서 벡터들을 각 자질로 투영해서 각 자질 별로 값들을 저장한다.



A.벡터 공간에서의 문서 표현　　B.자질 투영법을 통한 문서의 각 자질에서의 저장 표현

그림 1 문서 범주화에서 자질 투영법을 사용했을 경우의 문서의 각 자질에서의 저장 형태

위의 그림 1에서 자질 투영법에 의한 문서 저장 형태의 변화를 간단한 예를 통하여 표현하였다. 그림 1에서

는 각 문서는 2개의 범주만을 가질 수 있으며 2개의 자질로 표현된다. 각 문서는 식 (2)과 식 (3)에 의해 그림 1의 A와 같이 벡터 공간에 표현될 수 있다. 여기서 각 범주는 3개의 문서를 가지고 있다. ($c_1=\{d_1,d_2,d_3\}$, $c_2=\{d_4,d_5,d_6\}$) 그림 1의 B는 A의 각 문서들이 자질 투영법에 의해 각 자질 별로 저장되어 있는 표현 형태를 나타낸다. 각 자질에서의 개체(entity)는 그 자질이 어떤 문서(d)에서 어떤 범주(c)를 가지고 얼마의 가중치 값(w)으로 출현했는가를 표현하고 있다. 여기서 주의해야 할 점은 문서 표현에서 가중치가 0.0인 자질은 필요없는 자질이기 때문에 그림 1의 B처럼 f_2의 구성 요소의 수가 f_1의 수보다 1개가 적다는 것이다. 즉, 각 자질 표현의 구성 요소의 수는 각 자질(단어)의 문헌 빈도수(Document Frequency)와 같다.

3.2 자질 투영법을 사용한 새로운 문서 분류 알고리즘 (TCFP)

본 장에서는 본 논문에서 제안되는 새로운 알고리즘을 자세히 소개한다. 우리는 제안된 알고리즘을 TCFP (Text Categorization using Feature Projections)라고 명명한다. TCFP를 소개하기 위해서 먼저 자질 투영법을 문서 분류에 사용하기 위해서 고려해야 할 몇 가지 사항들을 기술하고, TCFP 알고리즘을 소개한다.

(1) TF-IDF 가중치 특성을 고려한 정규화된(normalized) 투표 방식

문서의 자질들의 TF-IDF 가중치는 문서의 내용을 구분하기 위해서 사용되는 그 자질의 중요도 값이다 [12]. 그렇기 때문에, 자질 투영법에 의해서 각 자질에 저장된 각 개체들은 높은 TF-IDF 값을 가지고 있을수록, 문서 분류를 위해서 더욱 유용한 개체가 된다. 이러한 성질을 TCFP 알고리즘에 반영하기 위해서 각 자질로 투영된 개체 중에서 평균이상의 TF-IDF값을 가진 개체만을 투표에 참여할 수 있게 한다. 또한 선택된 개체들은 각 개체들의 TF-IDF값과 같은 가중치로 투표에 참여하게 된다.

문서 분석(text analysis)에서의 또 다른 문제점은 자질로서 단어가 사용되기 때문에 자질의 수가 많고 또한 자질이 각 문서에 출현한 문헌 빈도가 무척 다양하다는 점이다. 여기서 자질의 문헌 빈도는 자질 투영법을 사용했을 때에 각 자질에 속해있는 개체 수와 같기 때문에, 2장 관련 연구에서의 k-NNFP와 같이 각 자질에 속해 있는 개체 중에 일부를 선택해서 투표에 참여하게 하는 방식에는 문제가 있다. 그러므로, TCFP 알고리즘에서는 다음 식 (4)과 같이 각 자질별로 정규화된 투표방식을 사용한다.

$$r(c_j,t_m)=\sum_{t_m(l)\in I_m} w(t_m,\vec{d}_l)\cdot y(c_j,t_{m(l)})/\sum_{t_{m(l)}\in I_m} w(t_m,\vec{d}_l) \quad (4)$$

위 식에서 I_m은 각 자질에 속한 개체 중에 TF-IDF 값의 평균보다 높은 값을 가지고 있어서 투표에 참여하도록 선택된 개체집합을 나타내고, $y(c_j,t_{m(l)})\in\{0,1\}$은 개체 $t_{m(l)}$의 범주가 c_j와 같으면 1이고 다르면 0을 출력하는 함수이다.

(2) 문맥 정보의 반영

자질 투영법을 문서 범주화에 사용했을 때의 또 다른 문제점은 문서 분석에서는 각 자질들 사이의 문맥 정보가 유용하게 사용되지만, 자질 투영법을 사용했을 경우에는 각 자질별로 투표를 수행하기 때문에 문맥 정보가 전혀 반영되지 않는다는 것이다. 이를 보완하기 위해서 공기 정보를 이용하여 TCFP 알고리즘에 문맥 정보를 반영하도록 하였다. 공기 정보를 계산하기 위해서 간단한 방법을 사용하였는데, 먼저 학습 문서에서 두 자질이 같은 문서에 출현한 빈도를 각 범주별로 계산하고 다음과 같이 범주별 공기 정보 중 최대 값이 그 두 자질의 대표 공기 정보 값이 된다.

$$co(t_i,t_l)=\max_{c_j} co(t_i,t_l,c_j) \quad (5)$$

식 (5)에서 $co(t_i,t_j,c_j)$는 각 범주 c_j에서의 자질 t_i과 t_l의 공기 정보 값이고, $co(t_i,t_l)$는 대표 공기 정보값이다.

이렇게 계산된 자질 쌍의 공기 정보 값은 입력 테스트 문서에서 출현하는 자질 쌍의 TF-IDF 값을 다음 식 (6)과 같이 조정함으로써 TCFP 알고리즘에 반영된다. 즉, 분류하고자하는 입력 테스트 문서에 출현한 자질 쌍이 학습 문서 집합에서도 같은 문서에서 많이 출현하면 할수록, 그리고 적은 범주에서 출현했을수록 높은 가중치 값을 갖도록 조정된다.

$$tw(t_i,\vec{d})=w(t_i,\vec{d})\cdot\left(1+\left(\frac{1}{\log(cf+1)}\right)\cdot\left(\frac{\log(co(t_i,t_j)+1)}{\log(\max co(t_k,t_l))}\right)\right) \quad (6)$$

위 식에서 $tw(t_i,\vec{d})$는 공기 정보에 의해 조정된 자질 t_i의 자질 가중치 값이고, cf는 자질 t_i와 t_j가 출현한 범주 빈도수를 나타낸다. 또한 $\max co(t_k,t_l)$는 전체 자질 쌍의 공기 정보값 중에 가장 큰 값을 나타낸다.

(3) 자질 정보량을 반영하는 각 자질의 최종 투표 계산 방식

자질 투영법을 사용했을 경우에 각 자질별로 분류를 위한 투표에 참여하기 때문에, 미리 자질 선택 과정을 통해 계산된 각 자질의 정보량을 TCFP 알고리즘에 적용할 수 있다. 본 논문에서는 자질 선택을 위해서 χ^2 통계량을 사용하였고, 그 과정을 통해 계산된 자질 정보량을 각 자질의 최종 투표 값에 다음 식 (7)과 같이 반영하였다.

$$vs(c_j,t_m)=tw(t_m,\vec{d})\cdot r(c_j,t_m)\cdot\log(1+\chi^2(t_m)) \quad (7)$$

위 식에서 $\chi^2(t_m)$는 χ^2통계량 기법에 의해 계산된 자질 t_m의 정보량이다.

(3) TCFP 알고리즘

새로 제안된 TCFP 알고리즘의 pseudo 코드는 다음과 같다.

```
Given: test document: d=<t1, t2, ..., tn>, a category set:
C={c1, c2, ..., cm} Begin
    for each category cj
        vote[cj] = 0
    for each feature ti
        tw(ti, d) is calculated by formula (6)

    /*Majority Voting*/
    for each feature ti
        vote[cj]=vote[cj]+ vs(cj, ti) by formula (7)

    prediction= arg max_cj vote[cj]

    return prediction
End
```

TCFP는 위에서 보는 바와 같이 단순한 알고리즘이며, 학습 과정에서는 문서를 각 자질별로 투영한 후, 자질 별로 정규화 된 투표 비율을 계산하고 자질 쌍의 공기 정보 값 만을 계산하면 된다.

4. 실험 및 결과

4.1 실험 데이타 및 실험 환경

실험에서 사용한 테스트 문서 집합은 문서 범주화 영역에서 주로 사용되는 대표적인 두 가지를 사용한다. 첫 번째 문서 집합은 뉴스 그룹(UseNet discussion group)의 문서들을 모아 놓은 테스트 문서 집합(Newsgroups)[13,14]으로써, 20개의 범주에 총 20,000개의 문서들로 구성되어 있다. 두 번째 문서 집합은 CMU의 WebKB 프로젝트로부터 생성되었다. 이 문서 집합은 대학의 컴퓨터학과 웹 문서들을 수집한 것으로 웹 문서들은 course, faculty, project, student, department, staff, other의 7개 범주로 나누어져 있다[15]. 본 논문에서는 이 중에 다른 논문들에서 주로 사용하는 4개의 범주들(course, faculty, project, student)을 사용한다[2,13]. 하지만, 뉴스 그룹 문서 집합과 WebKB 문서 집합은 학습 문서와 테스트 문서의 구분이 없으므로, 공정한 평가를 위해서 five-fold cross validation 기법으로 평가하였다. 즉, 전체의 20%를 테스트 문서로 하고 나머지를 학습문서로 사용하여, 총 다섯개의 학습 문서와 테스트 문서의 집합을 만들어 각각 실험하고, 실험 결과의 평균값으로 성능을 평가하는 기법이다. 불용어 사전을 사용하였으며 스테밍(stemming)은 사용하지 않았다.

실험에서는 제안된 TCFP 알고리즘의 성능을 비교하기 위하여 일반적으로 많이 사용되는 문서 분류기들을 구현하고 비교하였다. 실험에서 사용된 문서 분류기는 k-NN, Naive Bayes, Rocchio, SVM이다. k-NN을 위해서 k값을 30으로 사용하였으며, Rocchio 문서 분류기를 위해서는 α=16 그리고 β=4가 사용되었다. 또한 SVM을 위해서는 SVM^{light} 툴을 이용하여 문서 분류기를 구현하였다. 자질 투영법을 사용한 k-NNFP와의 성능 비교를 위해서 k-NNFP를 구현하고 실험하였다.

4.2 성능 평가 방법

평가 방법으로는 정보 검색 분야에서 일반적으로 사용되는 정확율(precision)과 재현율(recall)을 사용하였으며, 정확율과 재현율을 하나의 값으로 표현해주기 위해서 다음 식 (8)과 같이 F_1-measure를 사용하였다.

$$F_1(r,p)=\frac{2rp}{r+p} \tag{8}$$

식 (8)에서 r은 재현률에 해당하고 p는 정확률에 해당한다.

모든 범주의 성능을 통합하여 평가하기 위한 기법으로는 문서 범주화 기법의 성능 평가에 주로 사용되는 micro-averaging기법을 사용한다[16].

4.3 실험 결과

4.3.1 뉴스 그룹 문서 집합에서의 성능 평가

다음 그림 2와 표 1은 TCFP와 다른 문서 분류기와의 성능을 비교하고 있다. 우리는 여기서 TCFP 알고리즘의 다른 형태(TCFP without context)를 실험에 포함시켰다. TCFP without context는 식 (6)에 해당하는 문맥 정보를 반영하지 않은 알고리즘으로 이런 형태의 알고리즘을 실험에 포함시킴으로써 문맥 정보의 반영이 성능에 미치는 영향을 분석할 수 있으며, 또한 문맥 정보를 반영하지 않고도 k-NN과 비슷한 성능과 빠른 수행속도를 TCFP 알고리즘이 지님을 보일 수 있다.

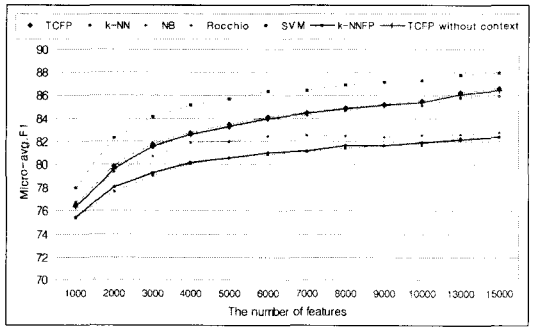


그림 2 뉴스 그룹 문서 집합에서의 자질 수에 따른 문서 분류기의 성능 변화

표 1 뉴스 그룹 문서 집합에서의 각 문서 분류기의 최고 성능 비교

	TCFP	TCFP without context	k-NNFP	k-NN	Naive Bayes	Rocchio	SVM
micro-avg. F_1	85.52	85.42	81.93	85.15	82.51	81.68	87.32

비록 현재 문서 범주화에서 가장 높은 성능을 보이고 있는 SVM보다는 낮은 성능을 보이고 있지만 k-NN을 비롯한 다른 문서 분류기보다는 높은 성능을 보이고 있으며, 특히 TCFP와 같이 자질 투영법을 사용하는 k-NNFP보다 많은 성능 향상을 얻을 수 있었다.

4.3.2 WebKB 문서 집합에서의 성능 평가

WebKB 문서 집합은 뉴스 그룹 문서 집합과는 달리 각 범주에 문서가 균일하게 들어 있지 않다. 즉, 가장 큰 범주는 1,641개의 문서를 가지고 있는 반면 가장 작은 범주는 단지 503개의 문서만을 지니고 있다. 이렇게 각 범주의 크기가 다른 경우에 TCFP 알고리즘을 사용했을 때 문제가 발생할 수 있는데, 그 이유는 범주의 크기가 다른 경우에는 결국 많은 문서를 가지고 있는 범주가 많은 투표 후보자를 갖게 되기 때문에 큰 범주로 많이 할당되는 경향이 발생한다. 이를 개선하기 위해서 우리는 정규화 된(normalized) 투표 방식을 도입하는데, 다음 식 (9)을 이용해서 최종 투표 값을 조절한다.

$$vote[c_j] = vote[c_j] \cdot \left(\max_{c_i} \{N(d, c_i)\}/N(d, c_j) \right) \quad (9)$$

식 (9)에서 $N(d, c_i)$는 범주 c_i에서의 학습 문서 수이다.

다음 그림 3과 표 2는 WebKB에서의 각 분류기의 성능을 보이고 있다.

WebKB 문서 집합에서도 뉴스 그룹 문서 집합과 비슷한 결과를 얻을 수 있었다.

4.3.3 TCFP 알고리즘 분석

앞의 실험에서 관찰한 바와 같이 TCFP 알고리즘은 성능 면에서 현재 문서 범주화에서 가장 좋은 성능을 보이고 있는 SVM보다는 조금 낮았지만 k-NN과 다른 문서 분류기보다는 높은 성능을 보이고 있다. 본 절에서는 k-NN의 약점인 느린 수행 속도와 오류데이타로부터 약한 SVM의 약점이 어떻게 TCFP에서 보완되었는가를 실험으로 보인다. 즉, TCFP 알고리즘은 단순함으로 인해 구현하기 쉽고 빠른 수행 속도를 가지고 있으며, 오

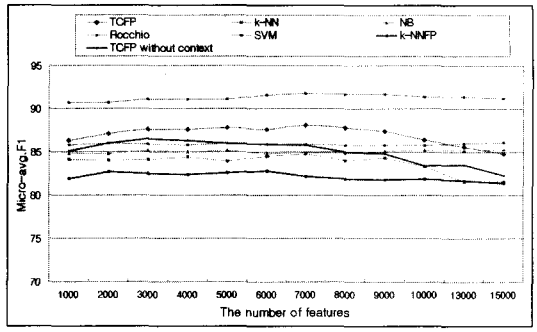


그림 3 WebKB에서의 자질 수에 따른 문서 분류기의 성능 변화

류 데이타가 많은 영역에서 보다 나은 성능을 보인다. 첫 번째로 수행 속도를 관찰하기 위해서 뉴스 그룹 문서 집합의 실험에서의 각 문서 범주기의 수행 속도를 측정하였으며, 두 번째로는 오류 데이타를 증가시키며 성능을 관찰하여 각 문서 분류기의 견고함을 관찰하였다.

(1) 수행 속도 관찰

수행 속도를 관찰하기 위해서 두 가지 문서 집합에서 문서 분류 시 CPU seconds를 관찰하였다. 수행 속도는 자질의 수에 의존적이기 때문에 1,000개부터 10,000개까지 1,000개의 간격으로 10번의 수행시간을 관측해서 평균값을 표 3에 보였다.

표 3에서와 같이 TCFP의 수행 속도는 일반적으로 빠른 수행 속도를 가지고 있는 Rocchio와 Naive Bayes 문서 분류기와 비슷한 수행 속도를 보이고 있다. 특히, k-NN보다는 뉴스 그룹 문서 집합의 경우에 거의 100배 넘게 빠른 알고리즘이며 SVM과도 많은 차이를 보이고 있다. 또한, TCFP without context는 실험에 참가한 문서 분류기 중에 가장 빠른 수행속도를 보이고 있다. 이러한 결과는 TCFP without context의 수행속도는 단지 실험 문서의 자질의 수(m)와 범주의 수(c)에

표 2 WebKB 문서 집합에서의 각 문서 범주기의 최고 성능 비교

	TCFP	TCFP without context	k-NNFP	k-NN	Naive Bayes	Rocchio	SVM
micro-avg. F_1	88.07	86.52	82.78	84.83	85.22	85.98	91.75

표 3 각 문서 집합에서의 문서 분류기의 수행 속도 비교

	TCFP without context	k-NNFP	Rocchio	TCFP	Naive Bayes	SVM	k-NN
Newsgroups	0.7	0.85	0.8	1.29	1.22	14.71	142.54
WebKB	0.14	0.23	0.14	0.55	0.17	2.72	15.25

만 비례하기 때문이다($O(mc)$).

(2) 오류 데이타로부터의 견고성 분석

우리는 여기서 TCFP가 오류데이타로부터 가장 견고한 문서 분류기임을 보인다. 실험을 위해서 뉴스 그룹 문서 집합으로부터 10%부터 40%까지 오류데이타를 포함하는 4개의 문서 집합을 새로 생성하였다. 오류 문서는 각 범주로부터 임의로 추출해서 각 범주에 임의로 할당하였다. 실험 결과가 다음 그림 4에 보여진다.

그림 4에서 보는 바와 같이 TCFP는 20%의 오류데이타를 가지고 있는 문서 집합에서부터 문서 분류기 중에 가장 좋은 성능을 보인다. 특히, SVM은 오류데이타가 증가함에 따라 급속히 성능이 저하됨을 알 수 있다. 실제적으로 문서 범주화 기법을 사용할 때, 해당 영역의 학습 데이타가 많은 오류 문서를 포함하는 것은 일반적인 일이다. 결과에서 보는 바와 같이 오류데이타의 발생은 문서 범주화의 성능에 많은 영향을 끼친다. 이러한 TCFP의 견고성은 각 자질 별로 독립된 투표 방식에 기인한다. 즉, 자질 별 투표 방식이 분류과정에서 가능한 오류데이타의 영향을 줄일 수 있다. 또한, TCFP는 오류 자질(irrelevant feature)에도 견고한 특성을 지니고 있는데 k-NN과 SVM은 오류 자질에 의해 벡터공간에서 문서의 각도가 바뀌지만 TCFP는 단지 그 자질의 투표값에만 영향을 미치기 때문이다.

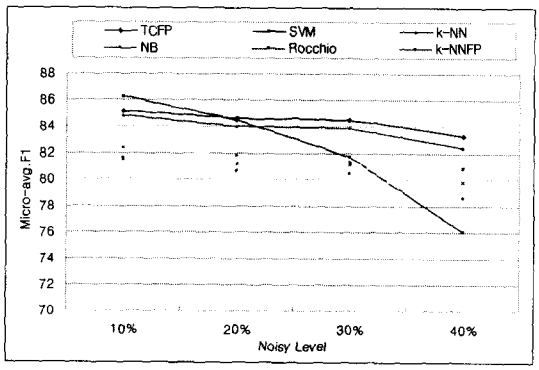


그림 4 오류데이타를 포함하는 4개의 문서 집합에서의 성능 비교

5. 결론 및 향후 과제

본 논문에서는 새로운 형태의 문서 분류기인 TCFP를 소개하였다. TCFP 알고리즘은 단순하기 때문에 구현이 쉬우며, 빠른 수행 속도와 오류데이타로부터 매우 견고하다는 특징을 가지고 있다. 실험을 통해서 본 TCFP의 성능은 현재 문서 범주화 분야에서 최고의 성능을 보이고 있는 SVM보다는 조금 낮은 성능을 보이고 있지만 수행 속도와 알고리즘의 단순성, 오류데이타로부터의 견고성에서 SVM보다는 매우 좋은 장점을 보이고 있다. 또한, k-NN 보다는 성능 뿐만 아니라 견고성과 수행속도 모두에서 더 좋은 결과를 보이고 있다.

TCFP 문서 분류기의 이러한 장점들로 인해 오류데이타에 강하고 빠르고 높은 성능을 요구하는 문서 범주화 영역에서 유용하게 사용될 수 있을 것이다.

향후 과제로는 TCFP의 투표 방식은 각 자질의 자질 가중치에 의존하기 때문에 더 좋은 자질 가중치를 적용함으로써 더 나은 성능을 보일 수 있을 것이라고 생각한다.

참 고 문 헌

[1] D. D. Lewis. "Naive (bayes) at forty: The independence assumption in information retrieval," *European Conference on Machine Learning*, 1998.

[2] A. McCallum and K. Nigram, "A comparison of event models for naive bayes text classification," *AAAI '98 workshop on Learning for Text Categorization*, 1998.

[3] D. D. Lewis and M. Ringuette, "A comparison of two learning algorithms for text categorization," *Proceedings of the 3rd Annual Symposium on Document Analysis and Information Retrieval*, 1994.

[4] C. Cortes and V. Vapnik. "Support vector networks," *Machine Learning*, 20:273-297, 1995.

[5] T. Joachims. "Text categorization with support vector machines: learning with many relevant features," *European Conference on Machine Learning (ECML)*, 1998.

[6] Y. Yang. "Expert netword: Effective and efficient learning from human decisions in text categorizatin and retrieval," *17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'94)*, pp.13-22, 1994.

[7] D. D. Lewis, R. E. Schapire, J. P. Callan and R. Papka, "Training algorithms for linear text classifiers," *Proceedings of the 19th International Conference on Research and Development in Information Retrieval(SIGIR'96)*, pp.289-297, 1996.

[8] E. Wiener, J. O. Pedersen, and A. S. Weigend. "A neural network approach to topic spotting," *Proceedings of the Fourth Annual Symposium on Document Analysis and Information Retrieval (SDAIR'95)*, 1995.

[9] Y. Yang and X. Liu, "A re-examination of text categorization methods," *Proceedings of ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR'99*, pp. 42-49, 1999.

[10] I. Sirin and H. A. Guvenir, "An algorithm for

classification by feature partitioning," Technical Report, Department of Computer Engineering and Information Science, Bilkent University, 1993.

[11] A. Akkus and H. A. Guvenir, "K nearest neighbor classification on feature projections," Proceedings of ICML'96, Italy, pp. 12-19, 1996.

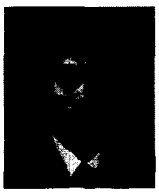
[12] G. Salton and M. J. McGill, *Introduction to modern information retrieval*, McGraw-Hill, Inc, 1983.

[13] K. Nigam, A. McCallum, S. Thrun, T. Mitchell, "Learning to classify text from labeled and unlabeled documents," *Proceedings of 15th National Conference on Artificial Intelligence (AAAI-98)*, 1998.

[14] Y. Ko, J, Park, and J, Seo, "Automatic text categorization using the importance of sentences," *Proceedings of the 19th International Conference on Computational Lin- guistics (COLING'2002)*, pp.474-480, 2002.

[15] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery, "Learning to construct knowledge bases from the world wide web," *Artificial Intelligence*, 118 (1-2), pp. 69-113, 2000.

[16] Y. Yang, "An evaluation of statistical approaches to text categorization," *Information Retrieval Journal*, May, 1999.



고 영 중

1996년 서강대학교 수학과 학사. 1996년~1997년 LG-EDS 근무. 2000년 서강대학교 컴퓨터학과 석사. 2003년 서강대학교 컴퓨터학과 박사. 2003년~현재 서강대학교 산업기술연구소 연구원. 관심분야는 한국어 정보 처리, 문서 범주화, 문서 요약, 대화처리, 소프트웨어공학, 바이오인포메틱스 등

서 정 연

1981년 서강대학교 수학과 학사. 1985년 미국 Univ. of Texas, Austin 전산학과 석사. 1990년 미국 Univ. of Texas, Austin 전산학과 박사. 1990년~1991년 미국 Texas Austin, UniSQL Inc. Senior Researcher. 1991년 한국과학기술원 인공지능 연구 센터 선임연구원. 1991년~1995년 한국과학기술원 전산학과 조교수. 1996년~현재 서강대학교 정교수. 관심분야는 한국어 정보 처리, 자연어처리, 대화처리, 지능형 정보 검색 등