논문번호 04-01-17

SVM을 이용한 디렉토리 기반 기술정보 문서 자동 분류시스템 설계

Design of Automatic Document Classifier for IT documents based on SVM

강 윤 희*, 박 용 범**
Yun-Hee Kang*, Young B. Park**

요 약

인터넷 상의 정보가 급증하여 필요한 정보를 찾고 관련된 정보를 조직화하는데 많은 시간이 소요된다. 따라서 정보접근 부하를 줄일 수 있는 자동적인 문서 분류의 중요성과 필요성이 증가하고 있다. 본 논문에서는 웹 문서의 자동 분류 시스템의 설계와 구현을 기술한다. 디렉터리 내의 학습 문서 집합을 기반으로 구성된 대표 단어 집합을 이용하여 문서 분류 모델을 학습하기 위해 SVM을 사용하였다. 본 시스템에서는 정보통신 웹 디렉터리 내의 문서로부터 추출된 단어 집합을 기반으로 SVM을 학습 시킨 후 신규 문서에 대해 문서 분류를 수행한다. 또한 TFiDF를 기반으로 특성을 표현하기 위해 벡터공간 모델을 사용하였고 학습 데이터는 가중치를 갖는 특성 집합으로 표현되어진 긍정 및 부정 집합으로 구성하였다. 실험에서는 문서분류의 결과 및 벡터길이의 관련성을 보인다.

Abstract

Due to the exponential growth of information on the internet, it is getting difficult to find and organize relevant informations. To reduce heavy overload of accesses to information, automatic text classification for handling enormous documents is necessary. In this paper, we describe structure and implementation of a document classification system for web documents. We utilize SVM for documentation classification model that is constructed based on training set and its representative terms in a directory. In our system, SVM is trained and is used for document classification by using word set that is extracted from information and communication related web documents. In addition, we use vector-space model in order to represent characteristics based on TFiDF and training data consists of positive and negative classes that are represented by using characteristic set with weight. Experiments show the results of categorization and the correlation of vector length.

Keyword: Feature Selection, Vector Space, SVM, Representative Term, Document Classification

I. 서론

최근 인터넷의 급속한 성장과 보급에 따라 전자우편과 웹을 통해 제공되어지는 정보의 양은 기하급수적으로 증가하고 있다. 또한 정보의 종류도 개인정보, 상품 카탈로그, 기술정보, 특허 등 매우 다양하다. 그러나 웹을 통해 접근 가능한 정보에서 실제 사용자에게 필요한 정보는 극히 일부분이며 이러한 현상을 정보과부하(Information Overload)라고 한다[11]. 정보과부하 문제의 해결을 위해서는 인터넷상의 유용한 정보를 확인하기 위한 필요 정보를 분류하는 작업이 필수적이다[1,5,6].

현재 대부분의 문서 분류는 수작업으로 이루어지고 있

* 천안대학교 정보통신학부 조교수
(Assistant professor, Division of Information and Communication)
** 단국대학교 전자컴퓨터학부 부교수
(Associate professor, Computer Science)
接受日:2004年 5月 18日, 修正完了日:2004年 12月 17日

다. 문서 및 주제 분야의 증가에 따라 수동 분류는 반복 작업에 따른 비용, 오분류 증가 및 과다한 시간 소요 등의 문제점이 있다. 이러한 문제점을 해결하기 위해 미리 정의된 카테고리[1]에 텍스트 문서를 한 개 또는 그 이상의 카테고리로 분류하는 자동 분류가 정보검색 분야에서 매우 중요한 분야로 등장하였다[5,6]. 감독자 기반의 기계학습 기법인 SVM(Support Vector Machine)은 최적화 해를 구하기 위한 기법으로 소개된 후 이미지 내의 특징을 찾기 위한 패턴인식(Pattern Recognition) 분야에서 높은 성능을 보인다. 최근에는 자동 문서분류를 위해 SVM을 적용하는 연구가 이루어지고 있다. SVM 기반 문서 분류의 성능은 특징 선택 기법에 의해 결정된다[3,4,5,6].

기존연구에서는 주제 및 대상이 서로 상이한 분야의 문서에 대한 자동분류를 수행하였으나, 본 연구에서는 유사한 관련 분야로 구성되어진 정보통신 분야 10개 대분류 디렉터리의 기술 문서 및 보고서를 대상으로 문서 자동분류를 수행하였다. 특히 정보통신 부분의 기술문서는 신기술의 등장에 의해 신규 전문용어 및 관련 분야 별로 신규 공통 용어들을 다수 포함하는 특징을 갖는다.

기존의 특징 벡터 구성을 위한 키워드 선택은 출현 빈도, 카테고리 별 빈도수 등 제한적인 추출기법을 사용한다[1,2,5]. 그러나 본 실험에서는 전체 디렉토리 내에서 디렉토리별로 특정 단어의 중요도에 따라 대표용어를 선출한 후 이에 따른 가중치를 고려하여 확장된 추출기법을 사용한다. 본 대표 용어 추출 기법은 기술문서의 분류를 위해 카테고리의 대표용어를 추출한 후 용어에 가중치를 결정하는 방식으로 대표적인 키워드 추출 기법인 DF(Document Frequency)을 확장하였다. SVM 기반 문서자동분류에 대한 실험을 수행한다.

본 논문의 2장에서는 문서분류 기법 및 특성벡터 구성을 키워드 추출을 위한 관련연구를 기술하고, 3장에서는 새롭게 제안하는 디렉토리 기반 문서 분류 기법을 기술한다. 4장에서는 실험 및 결과에 대한 평가를 하고 5장에서는 결론 및 향후연구를 기술한다.

II. 관련 연구

클러스터링(Clustering)은 데이터 마이닝(Data mining)방법의 기법으로서 자동적인 질의 형성과 유사문서 검색을 위해 관련된 문서의 특징을 추출하는 것을 목적으로 한다. 전통적인 클러스터링 알고리즘은 거리 또는 확률에 기반을 두어 유사도를 정의한다. 예를 들어 문서간의 벡터가 이루는 각도, 문서와 문서간의 거리, 확률적 베이지언 모델을 사용한다. 또한 문서 구조에 대한 선험적(Priori) 지식을 사용한다[8]. 또한 문서 클러스터링은 중심점(Centroid) 근처의 문서를 클러스터링하는 비계층 클러스터링과 모든 문서와 클래스간의 유사도를 계산하는 계층적 클러스터링 기법으로 구분한다.

텍스트 분류 또는 문서 분류는 문서 카테고리 중에 주어진 텍스트를 특정 카테고리에 할당하는 작업이다. 텍스트 분류는 인터넷의 급격한 확대로 인해 정보 검색에서 매우 중요한 분야로 등장하고 있다.

텍스트 분류에 대한 연구는 신경망과 통계적 접근으로 이루어지고 있다. E.D Wiener 는 텍스트 분류에 역전파(Back propagation) 알고리즘을 적용하였으며, 샘플 텍스트로부터 추출되어진 벡터는 최소 200 차원이었다[9]. Lewis와 Schapire는 학습가능 선형 모델인 perceptron과 EM 알고리즘을 텍스트 분류에 적용하였다[12]. Joachims는 SVM(Support Vector Machine)의 응용으로 차원의 문제를 완화하기 위해 텍스트 분류에 적용하였다[6]. Ng, Goh 와 Low는 Reuter 뉴스 기사의 분류의 기법으로 규칙 기반 학습을 기반으로 기법을 제안하였다[13]. D. Merkl은 SOM(Self Organizing Map)의 비 감독(Unsupervised) 학습을 문서 분류에 적용하였다[14]. Larkey는 k-NN(K Nearest Neighbor)와 Bayesian 독립 분류기를 사용하여 에세이를 분류하였다[15]. Yang과 Liu는 k-NN, SVM, 역전파와 LLSF(Linear Least Square Fit)의 통계적 기준을 사용하는 텍스트 분류의 성능을 비교하였다. Yang은 K-NN, Linear Least Square Fit 와 WORD를 성능 측면에서 분석하였다[5].

문서분류의 대상이 되는 문서인 텍스트는 자연어로 쓰인 비 구조화된 데이터이다. 이를 처리하기 위해서는 구조적인 데이터로 표현할 필요가 있다. 텍스트 분류를 위해 텍스트는 특성 벡터로 표현하며 벡터내의 특성은 단어와 각 특성의 값으로 구성된다. 특성 값은 빈도수, 존재 유무 및 가중치이다[1,2]. 특성 벡터는 문서를 구성하는 전체 단어를 대상으로 불용어 및 빈도수에 따른 중요도를 고려하여 구성함으로써 차원을 줄인 후 문서 분류에 사용한다.

감독 학습 기반의 색인어 추출은 카테고리 별로 사전

1) 본 논문에서 카테고리, 클래스, 디렉터리는 같은 의미로 사용된다.

에 분류된 문서를 대상으로 대표 색인어를 추출하는 기법이다. 이 방법은 사전에 분류된 학습용 문서를 이용하기 때문에 비 감독학습 기법 보다 비교적 정확하게 색인어 추출이 가능하다. 그러나 사전에 각 카테고리 별로 학습용 문서를 분류하기 위한 비용과 카테고리를 재구성할 경우에 대표 색인어를 재구성해야 하는 어려움이 있다. 감독 학습 기법에 대한 색인어 추출 기법에는 DF, IG(Information Gain), MI(Mutual Information), χ^2 (chi Square), TS(Term Strength), LSI(Latent Semantic Indexing) 등이 있다[5]. DF 기법은 단어가 출현한 문서의 절대 빈도수만을 고려하는 기법으로 단순하고, 분류 성능도 비교적 우수하지만 출현 빈도만으로 문서의 카테고리를 비교할 수 없는 경우도 많이 발생할 수 있다. IG 기법은 정보 검색에서 주로 사용되는 색인어 추출 기법으로 카테고리 별 단어의 평균 빈도수를 고려하는 방법이다. χ^2 기법은 우연성 테이블을 이용하여 단어와 카테고리의 독립성을 고려하는 기법으로 텍스트 분류에서 비교적 높은 정확도를 나타낸다. 그러나 출현 빈도가 낮은 단어들에 대해서는 고려하기 어려운 단점을 가지고 있다. MI 기법은 텍스트 분류에서 많이 사용되는 색인어 추출 기법으로 단어와 카테고리 간의 독립성을 고려하여 대표 단어를 추출한다. TS 기법은 코사인 계수와 같은 식에 의한 유사도 계산을 통해 문서들을 사전에 클러스터링 한 후, 유사한 문서 쌍 내에서 출현 확률이 높은 단어만을 대표 색인어로 추출하는 기법이다.

신경망에 관련된 기계학습(Machine Learning)에는 흔히 MLP(Multi-Layer Perceptron), RBF(Radial Basis Function) 등이 사용되는데 이들을 이용한 최적화 알고리즘은 과적합(Over Fitting) 으로 인한 일반적인 문제해결에 어려움이 생길 수 있다. 이러한 제약들을 해결하기 위해 제안된 것이 SVM(Support Vector Machine) 이다. SVM 은 Marti A. Hearst 에 의해 최근에 소개된 기계학습 알고리즘으로서 여러 분류 프로그램들에 응용되고 있으며 높은 성능을 보여주고 있다[3,4,6].

본 연구에서는 SVM 기반 문서 분류를 위한 전처리 단계인 특징 선택에 대한 다양한 기법의 특징 및 성능을 제시하고 있는 [5]의 실험을 기반으로 비교적 높은 성능과 색인어 추출이 단순한 DF 기법을 사용하여 특징을 선택한다. 또한 DF는 디렉토리 대표 용어 추출 기법을 사용하여 색인어의 가중치 계산을 확장한다.

III. 제안된 디렉터리 기반 문서 분류

본 연구에서는 여러 개의 SVM을 병렬적으로 활용하여 문서분류를 수행한다. 먼저 정보 통신 관련 디렉터리로부터 추출된 데이터를 이용하여 각 클래스에서 학습 수행한다. 그리고 새로운 문서에 대해 학습된 SVM이 어떤 임계값을 넘을 경우 해당 클래스에 속하는 것으로 판단한다. SVM의 수는 디렉터리를 구성하는 카테고리의 수와 동일하다.

분류기 구성을 위해 디렉터리로부터 추출된 데이터를 이용하여 전처리(Preprocessing) 과정을 수행한다. 전처리 과정에서는 문서로부터 추출된 용어 중 불필요한 용어 및 빈도수에 따라 학습 및 분류를 위한 단어 집합인 특성벡터를 구성한다. 특성벡터를 구성하는 용어는 각 디렉터리의 대표 용어들을 이용하여 양의 가중치(Positive Weight) 또는 음의 가중치(Negative Weight)를 부여한다. 학습기는 특성 값이 결정된 학습 데이터(Training Data)와 목적값(Desired Value)의 쌍을 이용하여 각 클래스의 학습데이터로 SVM을 사용하여 학습한 후 새로운 데이터를 분류하기 위한 모델을 생성한다. 분류기는 생성된 학습 모델을 기반으로 새로운 문서들을 분류한다.

본 논문에서는 전체 대상문서에서 특징추출(Feature Extraction) 과정을 거친 후 만들어진 특성 벡터를 입력값으로 하여 각 클래스의 SVM에 입력하여 출력값이 임계값 이상인 클래스가 그 문서를 포함하는 것으로 결정한다.

3.1 벡터공간 모델

벡터공간 모델에서 질의와 각 문서는 용어 공간 내의 벡터로서 표현한다. 벡터 $w_{ij} \geq 0$를 (k_i, d_j)쌍의 가중치라고 하며, 이 가중치는 문서의 의미적 내용을 설명하기 위한 색인어의 중요도를 정량화한다. 시스템 내의 색인어 수를 t라 하고 k_i를 색인어라 하면 모든 색인어 집합 K={k_1, k_2,... ,k_t}이다. 문서 d_j에서의 색인어 k_i의 가중치는 색인어 k_i의 가중치는 $w_{ij} \geq 0$이고 따라서, 문서 내에 한번도 출현하지 않은 색인어의 가중치는 0 이 된다. 문서 $\overrightarrow{d_j}$ 는 색인어 벡터 $\overrightarrow{d_j}$ = <w_{1j}, w_{2j},..., w_{tj}>로 표현된다. 문서 내의 용어에 대한 가중치 벡터(d)의 계산은 용어 빈도(TF)와 역 문서 빈도수(IDF)로서 정의한다[2].

3.2 문서 분류 시스템

<그림 1>은 문서 분류 모델에 따라 구성된 문서 분류 시스템으로 전처리기, 학습기 및 분류기의 3개의 서브 시스템으로 이루어진다. 전처리기는 학습을 위해 분류를 위한 학습 벡터 벡터를 구성한다. 또한 전처리기는 분류를 위해 테스트 문서를 입력으로 분류를 위한 문서 벡터를 생성한다. 학습 벡터는 학습기에 입력으로 제공되고 학습기는 클래스 분류를 위한 모델을 생성한다. 대표용어는 키워드 추출 작업 과정에서 클래스들 간의 상대적 출현 빈도에 의해 결정되며 추출된 키워드가 클래스에 대표용어의 구성키워드인 경우에 키워드의 출현 빈도에 따른 TF*IDF에 가중치를 곱한다. 대표용어의 생성은 4장의 학습벡터 생성규칙에서 상세히 기술한다.

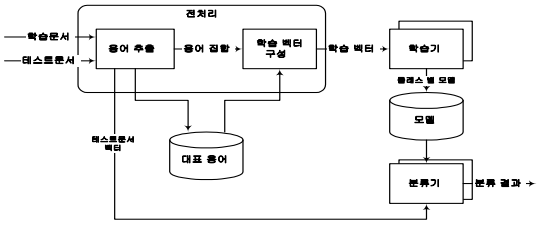

그림 1. 전체 시스템 구성도
Fig. 1. Architecture of the Overall System

3.3 실험 모델

본 연구에서는 재현율(Recall)과 정확률(Precision) 값의 결정을 위해 규칙 1을 사용한다. 규칙 1로부터 임의의 문서가 자신이 속한 클래스에 포함되었는가, 또는 포함되지 못하였는가를 고려할 때 각 클래스가 갖는 threshold 값을 기준으로 먼저 재현율 값을 결정하고 이후 해당 문서가 정확히 자신이 원하는 클래스에 속하는지에 관한 정확률의 값을 결정한다.

규칙 1 : 만약 문서 D1이 클래스 C1에 1순위로 속한다면 정확률은 증가한다. 또한 D1이 순위에 관계없이 일단 C1 클래스에 속한다고 결정이 되면 재현율은 증가되며 만약 D1이 클래스 C1에 1순위로 속하지 않는다면 정확률은 감소한다.

규칙 1에 따라 클래스 간에 강한 연관성을 갖는 클래스의 경우 문서의 정확율은 낮아지지만 재현율은 높아질 수 있다. 본 실험에서는 불필요한 특성을 제거하기 위한 특성 선택(Feature Selection) 기준을 "단 stop-word는 제외하고 문서 내에 3번 이상 발견되는 경우 "로 하였으며 카테고리 간에 많은 연관성을 갖는 정보통신 문서 집합을 대상으로 한다.

IV. 실험 결과 및 평가

본 실험을 위해 SVM은 SVM Lite[7]를 사용하여 10-11개의 분류기를 구성하였다. 개별 SVM 분류기는 학습 문서를 사용하여 문서 분류를 위한 모델을 구성한다. 모델 구성을 위한 학습은 긍정문서 집합과 부정문서 집합 모두를 사용하여 수행하며 다수 클래스의 문서 분류에 적용할 수 있도록 모델링 한다. 본 실험에서는

1) 학습 특성 벡터에 따른 재현율과 정확율의 성능 비교
2) 잡음을 갖는 카테고리에 대한 문서 분류의 강건성
3) 특성 벡터의 희소성 구성에 따른 문서 분류의 성능 비교
4) 학습 문서 수에 따른 성능 비교

위 4가지 평가 기준에 따라 실험 시나리오를 작성하여 실험을 수행한 후 결과를 평가한다.

4.1 학습을 위한 특성 벡터 구성

본 실험의 문서는 정보통신 분야 디렉터리 서비스 시스템인 "itfind"로부터 수집된 문서를 대상으로 하였으며 4가지 시나리오에 따라 수행하여 각 시나리오 별로 재현율/정확률 및 미분류율을 성능 평가 요소로 삼았다. 학습은 SVM Light를 사용하여 10개의 feature vector 파일로부터 10개의 모델을 만든다. 기본적으로 모델의 구성을 위한 커널 함수는 linear함수를 사용하며 특정 클래스에 대해서는 radial 함수를 적용한다.
본 실험에서는 문서 클래스내의 출현 용어와 클래스간의 연관성을 고려하여 클래스 대표용어를 추출하였으며 (규칙 2), 추출된 용어는 학습 벡터 내의 해당 용어의 가중치 조정에 규칙 4를 적용한다. 테스트 문서를 위한 키워드 데이터에 대해 특성벡터를 만들고 10-11개의 모델에 따라 문서분류를 수행한다.

규칙 2: 클래스 대표용어 추출을 위해 class document frequency/total document frequency 의 값이 threshold를 넘는 키워드를 모아 클래스 별로 대표용

어 리스트를 구성한다.

문서 학습을 위해 트레이닝 데이터를 수동으로 추출하여 이용하였으며 규칙 3을 사용하여 학습 벡터를 구성하였다. 학습 벡터의 차원 학습에서 벡터 크기를 500, 1000, 2000으로 설정하였다. 수집된 문서에 대한 색인 구축은 문서 내에 발생된 용어의 빈도수와 디렉터리 대표용어를 기반으로 하여 가중치를 설정한다. 기존의 TF*IDF 수식은 규칙 3-3과 같이 개선하였다.

규칙 3:
1. 발생 빈도가 높은 키워드부터 정렬하여 t 차원 벡터를 구성한다.
2. 각 학습 문서로부터 학습 벡터의 키워드들의 빈도수를 구해서 TF * IDF 값으로 벡터를 구성한다.
3. 키워드가 해당하는 directory 키워드 리스트에 있으면 DF * TF 값에 1보다 큰 값을 곱하고 다른 클래스의 키워드 리스트에 있으면 1보다 작은 값을 곱한다.

학습을 위한 문서 벡터는 테스트 데이터를 이용하여 학습벡터 구성 단계에서 구한 벡터 값과 테스트 데이터에서 얻은 빈도수를 곱하여 학습 벡터를 구성한다.
문서 학습을 위한 학습문서는 수동으로 추출하여 이용하였으며 학습을 위한 특성 벡터를 구성하였다. 학습 벡터의 구성요소 값은 규칙 3을 통해 얻어진 값이다. 마지막으로 디렉터리의 특성을 기반으로 학습을 위한 SVM 학습기의 입력 문서벡터 집합 구성을 위해 규칙 4를 적용하였다.

규칙 4:
학습 문서 230개 파일에 대해 특성벡터를 생성한다. 생성된 각 클래스의 긍정 학습 특성 벡터와 부정 특성 벡터를 통합하여 개별 클래스에 대한 종합 feature vector를 클래스 수만큼 구성한다.

4.2 학습 수행

SVM 알고리즘은 2개의 클래스를 최적으로 나누는 hyper-plane을 찾기 위해 구조화되어진 위험 최소화를 사용한다. 결정 경계에 가장 가까운 벡터들이 support 벡터가 된다. 즉, support vector는 구분 가능한 hyper-plane을 결정한다. SVM 은 학습모듈과 분류 모듈로 구성된다. 학습 및 분류를 위한 입력 파일은 다음과 같은 형식을 갖는다.

<class>::= +1 | -1 | 0
<feature> ::= integer
<value> ::= real
<feature-value> ::= <feature>:<value>
<line> ::= <class> <feature-value>$^{+}$

다음은 SVM 입력을 위한 긍정 예제 학습 특성벡터의 일부이다. 구성 벡터는 클래스 의 값으로 '+'는 긍정 예를 표현하며 특성번호 및 해당 특성값의 쌍으로 구성된다. 1:0241480에서 1은 특성번호를, 0241480는 이에 대한 특성값을 나타낸다. 특성 번호는 해당 키워드 집합의 단어와 연관된다. 현재 1번 특성번호의 키워드의 가중치는 규칙3은 통해 얻게 된다. 다음은 입력 문서의 일부분의 특성 정보중 1에서 7을 구성하는 인터넷, 데이터, 네트워크, 멀티미디어, 바이러스, 암호화, 정보통신의 특성 값을 보인 것이다.
+1 1:0.241480 2:0.241480 3:-114.326101 4:0.241480 5:-114.326101 6:0.241480 7:0.241480

4.3 실험 평가

본 실험은 잡음에 대한 문서 분류의 성능 평가와 특성 벡터의 크기에 따른 성능 비교를 수행하였다. 잡음을 갖는 문서집합과 정제 된 문서에 대한 문서 분류 성능을 분석하기 위해 전문가를 통하여 정보보호 센터(CERT-KR)와 정보보호진흥원(KASA)에서 문서를 103개 수집하였다. 이를 가지고 정보보호 카테고리에 대한 학습을 수행하였다. 또한 문서 분류의 성능 측정의 성능 측정의 강건성을 보이기 위해 반도체 및 부품의 경우 학습문서는 관련 분야에 지식이 없는 초보자에 의해 구성하도록 하였다.
<그림 2>와 같이 10번 클래스의 정보보호의 경우 신규 학습문서를 추가하고 커널 함수으로 radial 함수를 적용하여 미분류 없이 100%의 재현율과 90% 이상의 성공률을 얻었으며 9번 클래스인 정보가전 역시 100%의 재현율을 얻었다. 현재 높은 문서 분류가 수행되어진 클래스로는 3, 7, 8, 9, 10 클래스이고 적절한 문서 분류가 수행되어진 클래스는 1, 2, 4, 6 클래스이며 낮은 정확률 및 재현율을 갖는 경우는 5, 8번 클래스이다. 특정 클래스에 대한 분류가 70% 미만인 것들이 10개 중 3개가 있었으며 이들 클래스의 경우 학습문서 구성이 적절하지 못하였으며 클래스간의 연관성이 높은 경우 낮은 성능을 보였다. 그러나 추가적인 학습문서의 정재 후 3개 클래스의 경우는 재현율이 90% 이상을 보였다.

<그림 3>은 기존의 클래스에 잡음을 갖는 클래스[2]인 4번 'XML' 관련 클래스가 추가되는 경우를 보인 것으로 전체적으로 재현율과 정확률이 떨어짐을 알 수 있었다. 그러나 분류 성능이 높은 클래스의 경우 정확율과 재현율에 영향을 주지 못함을 알 수 있었다. 잡음을 갖는 문서에 대한 분류에서는 현저한 문서 분류 오류를 보인다. (69/93)으로 평균 문서 분류성능을 밑도는 결과를 얻었다. 본 실험을 통해 SVM을 사용한 문서분류 모델이 잡음에 강함을 보인다는 것을 알 수 있었다. 또한 현재 잡음을 갖는 클래스를 '인터넷'과 유사한 하위 클래스로 'XML'로 대치하였을 때 전체적으로 재현율과 정확률이 높아졌으며, 반도체와 같이 10번 클래스에서 좋지 못한 분류 성능을 갖는 클래스의 경우 역시 정확률이 높아짐을 보인다. 앞서와 동일한 조건에서 XML에 대한 학습문서의 수를 추가할 때 전체적으로 재현율과 정확률이 이전 실험결과 보다 개선됨을 볼 수 있으며 특히 연관성이 높은 4번 클래스의 경우 재현율의 증가 및 미분류에 따른 실패를 줄일 수 있었다.

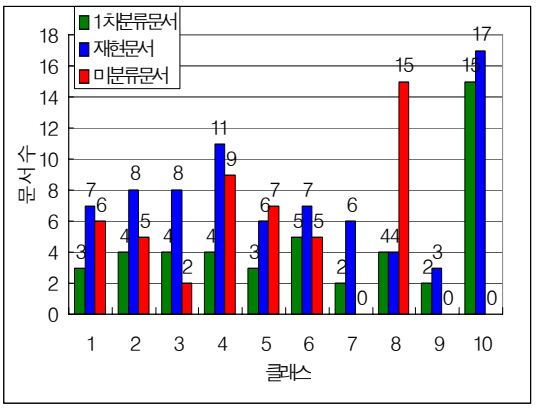

그림 2. 문서 분류 성능 비교
Fig. 2. Comparison of document classification

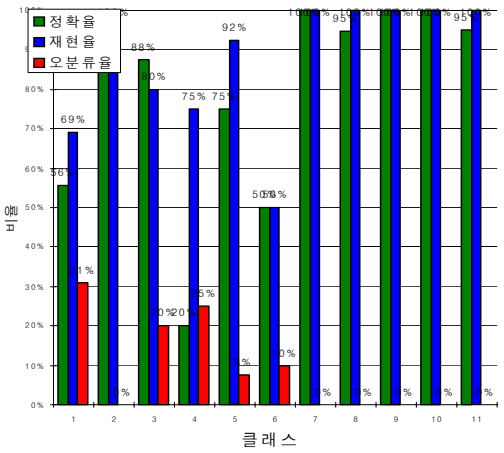

그림 3. 문서 분류 성능 비교
Fig. 3. Comparison of document classification with a noise class

본 실험을 통해 효율적인 커널 함수를 사용함으로써 문서 분류의 성능을 높일 수 있음을 알 수 있었으며 높은 문서 분류를 위해서는 많은 수의 학습문서 적용이 요구됨을 알 수 있다. 기존 연구에서도 80% 이상의 문서 분류의 성공을 위해 3000 여개 이상의 문서를 학습에 사용하였으나 본 실험은 <그림 3>와 같이 상호 연관성이 높은 문서 클래스에서의 효과적인 문서 분류를 위한 기법을 찾는 것을 목적으로 하여 앞의 실험보다 낮은 성능을 보였다. 전체적으로 실험 결과로 85.7%의 정확률과 78.8%의 높은 재현율을 얻었다. 또한 제안 기법을 사용한 분류의 정확률 및 재현율은 DF의 경우 76%과 75%을 보였으며, IG의 경우 정확률 및 재현율은 75% 및 70%을 보였다.

<그림 5>와 <그림 6>은 특성벡터 크기에 따른 성능을 보인 것으로 새로운 카테고리가 추가된 <그림 6>에서는 특성 벡터의 크기가 증가됨에 따라 재현율과 정확율 모두에서 높은 성능을 보였다.

2) 특정 클래스에서 분류하기 어려운 문서들의 집합으로 구성된 문서 클래스

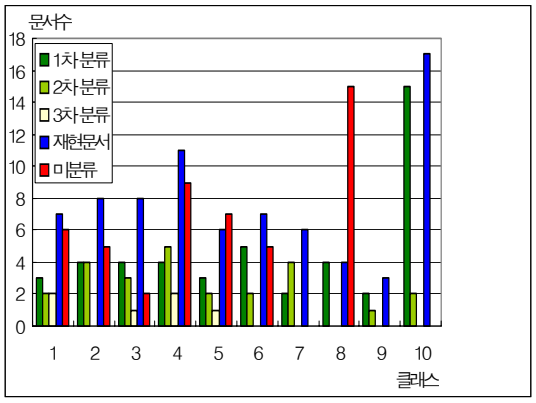

그림 4. 특성 벡터의 크기-500

Fig. 4. the Length of Feature Vector-500

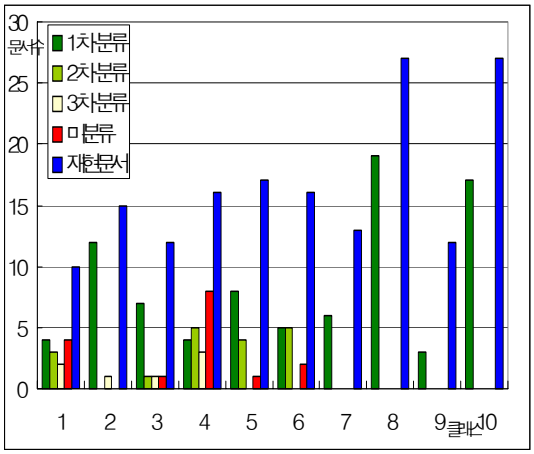

그림 5. 특성 벡터의 크기-2000

Fig. 5. the Length of Feature Vector-2000

4.4 분류 유형 평가

단위 문서의 분류 결과는 유사 분야 클래스로 대한 분류의 특징 및 학습 모델의 분류 결과인 임계 값을 기반으로 이루어진다. [표 1]은 문서 분류를 보인 것으로 SVM을 이용한 문서 분류 시 유사 분야 클래스 간의 상호 연관성은 문서 분류에 반영되게 되며 이에 대한 조정을 필요로 한다. 이를 위해 2가지 방법이 적용될 수 있다. 1)학습 과정에서의 클래스 유사도를 반영하는 방법과 2)분류되어진 결과에 대한 조정을 통한 방법으로 구분할 수 있다. 문서분류 결과 분석은 2)번째 방법인 결과에 대한 조정을 통한 클래스 유사도 방법을 목적으로 한다.

이를 위한 분류결과의 특징에 따라 4가지의 유형으로 나눈다.

1) 정상 분류(A) : 1순위 분류값이 양의 값이고 2순위와의 임계값 차이 0.7 이상인 경우
2) 정상 분류(B) : 1 순위 분류값이 음의 값이고 2 순위와의 임계값 차이가 0.7 이상인 경우
3) 미분류 : 1 순위 분류값이 음의 값이고 2 순위와의 차이가 0.2 이하인 경우
4) 오분류 : 1), 2) 3)의 경우 중 1,2,3 순위 내에 정상적인 클래스를 찾지 못한 경우

[표 1]에서 문서 8은 정상분류 (B)에 속하는 문서로 분류되어지며 그 외의 문서인 1,2,..,7 은 미분류로 분류되어진 문서집합이다. 분류 결과 집합을 구성하기 위해 문서 분류 인터페이스의 처리 결과는 4개로 구분하여 1순위에서 11순위까지의 분류 값을 유지한다. <그림 6>은 정상 분류와 미분류에 대한 분류 임계값

문서 번호 \ 순위	1	2	3	4	5	6	7	8	9	10	11
1	-0.83	-0.85	-0.88	-0.91	-0.95	-0.99	-0.99	-0.99	-0.99	-1.04	-1.13
2	-0.83	-0.92	-0.93	-0.97	-0.98	-0.99	-1.00	-1.00	-1.01	-1.06	-1.09
3	-0.89	-0.96	-0.97	-0.97	-0.98	-0.99	-1.00	-1.00	-1.00	-1.01	-1.07
4	-0.95	-0.98	-0.99	-1.00	-1.00	-1.00	-1.00	-1.00	-1.01	-1.01	-1.08
5	-0.87	-0.94	-0.95	-0.99	-1.00	-1.00	-1.00	-1.00	-1.01	-1.05	-1.06
6	-0.36	-0.56	-0.95	-0.95	-1.00	-1.00	-1.05	-1.05	-1.07	-1.09	-1.12
7	-0.85	-0.97	-0.98	-0.98	-1.00	-1.01	-1.01	-1.01	-1.02	-1.03	-1.04
8	-0.15	-0.95	-0.97	-1.00	-1.01	-1.03	-1.04	-1.04	-1.04	-1.08	-1.11

표 1. 문서 분류 결과

Table 1. the Result of Classification

과 값의 분포를 보인 것이다. 미분류 편차는 미분류 되어진 문서 8개의 값의 평균을 기준으로 얻은 값이고 정산 분류 및 편차는 정상 분류되어진 분류 결과를 보인 것이다. 본 분류 특성으로 규칙 5을 얻을 수 있다.

규칙 5 : 1순위와 2순위의 편차가 (0.3) 보다 작은 경우 미분류 되었다고 할 수 있으며 이 경우 순위간 분류차는 보다 적어진다.

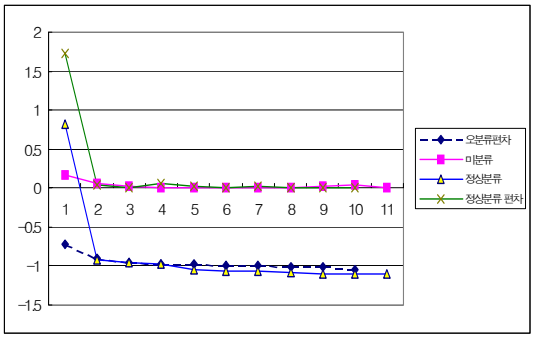

그림 6. 정상분류(A) 와 미분류 분류결과 비교
Fig. 6 the Comparison of Correct Classification and Misclassification

V. 결론 및 향후 연구

본 논문에서는 유사한 관련 분야로 구성되어진 정보통신 분야 10개 대분류 디렉터리의 기술 문서 및 보고서를 대상으로 문서 자동분류를 수행하였다. 본 논문에서는 기술정보 분야의 웹 문서에 대한 텍스트 자동 분류 시스템의 특성 추출 기법을 중심으로 기술하였다. 학습 문서 벡터는 웹 디렉터리 내의 문서로부터 추출된 용어 및 관련 문서를 기반으로 구성하였으며 학습 문서 구성 후, SVM 학습기를 통해 모델을 구성하여 문서 분류를 수행하였다. 본 실험을 통해 학습 벡터 구성과정에서 잡음에 의한 다른 클래스의 문서 분류에 미치는 영향을 고려한 SVM 기반 문서 분류 기법이 강건함을 보았다. 클래스간의 연관성이 높은 경우 낮은 성능을 보였으나 추가적인 학습문서의 정재 후 전체 클래스 성능 향상을 보임을 알 수 있었다. 또한 SVM 기반 문서 분류의 성능은 특징 선택 기법에 의해 결정됨을 DF와 IG를 사용한 특성벡터의 분류 실험을 통해 제안한 디렉토리 기반 기법이 우수함을 보였다. 향후 전체 시스템에 대한 자동 학습 기능을 기반으로 하는 지능형 에이전트에 적용할 예정이며 사용자 피드백을 통해 문서 분류 시스템의 성능 향상을 이룰 예정이다.

참고 문헌

[1] Tak W.Yan, Hector Garcia-Molina, "Sift - A Tool for Wide-Area Information Dissemination," In Proceedings of the 1995 USENIX Technical Conference, pp. 177-186, 1995.

[2] Salton, G. Automatic Text Processing: The Transformation, Analysis and Retrieval of Information by Computer, Addison-Wesley, 1989

[3] Chapelle, O., Haffner, P. and Vapnik, V., "SVM for histogram-based image classification," IEEE Trans. on Neural Networks, 10(5), pp.1055-1065,1999.

[4] T. Doszkocs, J. Reggia, and X. Lin. "Connectionist models and information retrieval," Annual Review of Information Science & Technology 25:209-260, 1990.

[5] Yang Y., J.O. Pdedersen, "A Comparative Study on Feature Selection in Text Categorization," Proc. Of the 14th Internatinal Conference on Machine Learning ICML-97, pp.412-429, 1997.

[6] Joachims, T., "Text categorization with support vector machines: Learning with many relevant features," Proc. European Conference on Machine Learning (ECML), pp. 137-142,1998

[7] Joachims, T., SVMLight, http://ais.gmd.de/~thorsten/svm_light, 1998.

[8] J. Martin, "Clustering full text documents," Proc. IJCAI-95 workshop on Data Engineering for Inductive Learning, 1995.

[9] E. Wiener, J. O. Pedersen and A. S. Weigend, A neural network approach to topic spotting, Proc. SDAIR '95, pp. 317-332, Las Vegas, NV, 1995.

[10] D. Lewis, W. A. Gale, A sequential algorithm for training text classifiers, Proc. SIGIR '94, pp. 3-12. Dublin, Ireland. 1994.

[11] Pattie Maes, "Agents that reduce work and information

overload," Communications of the ACM, 37(7), July 1994.
[12] D. Lewis, R. Schapire, J. Callan, and R. Papka, "Training Algorithms for Linear Text Classifiers," Proceedings of ACM SIGIR, pp.298-306, 1996.
[13] T.H.Ng, W.B.Goh, and K.L. Low, "Feature selection, perceptron learning and a usability case study for text categorization", 20 th ACM SIGIR Conference, 1997.
[14] Merkl, D., Exploration of text collections with hierarchical feature maps, In: Proceedings of the Int'l ACM SIGIR Conference on R&D in Information Retrieval, Philadelphia, PA, 186--195.Merkl, D., 1997.
[15] Leah Larkey. Automatic essay grading using text categorization techniques. In Proceedings of the 21st ACM/SIGIR (SIGIR-98), pages 90--96. ACM, 1998.
[16] Yang, Y., & Pederson, J. O. (1997). Feature selection in statistical learning of text categorization.

저 자 소 개

강 윤 희(Yun-Hee Kang)

1989년 2월 : 동국대학교 컴퓨터공학과 (공학사)
1991년 8월 : 동국대학교 컴퓨터공학과 (공학석사)
2002년 8월 : 고려대학교 컴퓨터과학과 (이학박사)
1991 7월 ~ 1994.4 : 한국전자통신연구원(연구원)
1994 4월 ~ 1997.2 : 한국문화예술진흥원, 전산개발부 (선임연구원)
1997 3월 ~ 2000.2 : (주)오롬정보 개발부(과장)
2000년 3월 ~ 현재 : 천안대학교 정보통신학부 조교수

연구분야 : 정보검색, 디지털라이브러리, 에이전트 시스템, 분산 시스템

박 용 범(Young B. Park)
제 7 권 2 호 논문 03-02-17 참조
단국대학교 전자계산학과 교수