IE Interfaces
Vol. 18, No. 1, pp. 10-21, March 2005.

# 전자메일 분류를 위한 나이브 베이지안 학습과 중심점 기반 분류의 성능 비교

김 국 표 · 권 영 식†

동국대학교 산업시스템공학부

# Performance Comparison of Naive Bayesian Learning and Centroid-Based Classification for e-Mail Classification

Kuk Pyo Kim · Young S. Kwon

Department of Industrial & Systems Engineering, Dongguk University, Seoul, 100-715

With the increasing proliferation of World Wide Web, electronic mail systems have become very widely used communication tools. Researches on e-mail classification have been very important in that e-mail classification system is a major engine for e-mail response management systems which mine unstructured e-mail messages and automatically categorize them.

In this research we compare the performance of Naive Bayesian learning and Centroid-Based Classification using the different data set of an on-line shopping mall and a credit card company. We analyze which method performs better under which conditions. We compared classification accuracy of them which depends on structure and size of train set and increasing numbers of class.

The experimental results indicate that Naive Bayesian learning performs better, while Centroid-Based Classification is more robust in terms of classification accuracy.

***Keywords***: classification, text mining, naive bayesian classifier, centroid-based classifier

## 1. 서 론

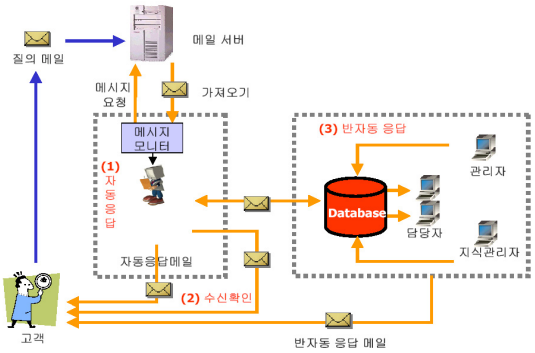
### 1.1 연구 배경

인터넷의 발전과 더불어 전자메일 사용자가 증가하게 되고, 기업의 고객접촉 채널로서 전자메일에 대한 중요성 또한 증가되고 있다. 그러나 Jupiter Communication의 연구에 의하면 55%의 고객들이 기업에 전자메일을 보낸 후 6시간 이내에 그에 대한 정확한 응답을 기대하지만 20%의 기업들만이 고객의 기대를 충족시켜 준다고 한다(LG Economic Research Institute, 2000). 이렇게 고객의 요구에 대해 적시에 적절하게 응답하지 못하면 고객의 불만족이 증가하게 되고, 그와 더불어 고객의 충성도(loyalty)를 감소시켜 결국 장기적 매출 및 수익성 악화를 초래하게 된다. 따라서 고객의 전자메일에 신속, 정확하게 응답할 수 있는 전자메일 자동관리 시스템의 필요성이 증가되고 있다.

<그림 1>은 전자메일 자동관리 시스템의 한 가지 예를 나타낸 것이다. 고객의 요구사항을 담은 전자메일이 메일 서버로 실시간으로 들어오게 되고, 이렇게 서버로 들어온 메일을 시스템이 불러 들여서 우선 자동응답이 가능한지를 판단하게 된다. 만약 자동응답이 가능한 메일일 경우 바로 고객에게 응답 메일을 보내게 된다. 그러나 자동응답이 가능한 메일이 아닌 경우 고객에게 메일이 처리중이라는 수신확인 메일을 보내어 고객의 요구사항이 현재 처리 진행중이라는 것을 알리고, 다음 단계로 넘어가게 된다. 본 연구에서 구현된 부분인 이 단계는 고객의 문의메일을 업무영역에 따라 분류하여 문의내용을 처리할 수 있는 담당자에게 보내주는 역할을 한다. 각 담당

† 연락저자 : 권영식 교수, 100-715 서울 중구 필동 3가 26 동국대학교 산업시스템공학과, Fax 02-2269-2212, E-mail : yskwon@dongguk.edu
2003년 10월 8일 접수, 2회 수정 후 2004년 12월 10일 게재 확정.

자는 본인이 맡은 업무와 관련 있는 문의 메일에 대해서 신속한 응답을 함으로써 모든 시스템의 처리과정을 마치게 된다. 이러한 전자메일 자동관리 시스템은 고객의 요구사항에 적합한 담당자에게 신속한 응답을 가능하게 함으로써 고객과 기업의 관계 유지에 많은 도움이 된다.




출처:(주)다우기술

**그림 1.** 전자메일 자동분류 시스템.

본 연구에서는 국내 기업환경에 적합한 분류기를 구현하여 전자메일 분류를 자동화한다. 자동분류를 통하여, 고객의 요구에 신속하게 대응할 수 있으며, 더불어 기업에 대한 만족도를 최대화하여 궁극적으로 이익을 극대화할 수 있다.

### 1.2 연구 목적 및 방법

최근에 텍스트 마이닝 분야의 연구가 활발히 진행되면서 많은 연구결과들이 나오고 있다. 텍스트 마이닝에 대한 연구들은 웹 문서와 같은 정형적인 형식을 갖춘 텍스트를 이용하는 연구가 주를 이룬다. 또한, 전자메일 분류에 관한연구도 있지만 대부분 스팸메일 분류나 개인 사용자의 관심 여부에 따른 분류가 주를 이루고 있다. 따라서 본 연구의 목적은 국내 기업의 전자메일을 업무영역에 따라 효과적으로 분류할 수 있는 전자메일 분류기를 구현하는 것이다.

우선, 전자메일과 일반문서의 차이점을 살펴보면, 전자메일은 일반문서와 달리 형식이 없고, 주로 개인적인 내용을 다루며, 문서길이가 짧다는 특징을 가지고 있다. 또한 표준어가 아닌 신조어, 인터넷 용어를 많이 사용하며 맞춤법이 틀린 경우가 많다는 것이다.

본 연구에서는 나이브 베이지안 분류기와 중심점 기반 분류기를 구현함으로써 실제 기업의 전자메일 분류실험을 통해 두 분류기의 성능을 비교 연구한다. 나이브 베이지안 분류기가 학습집합의 구성이나 속성들의 특징에 관계없이 의사결정나무 분류기보다 안정적인 성능을 보인다는 것은 기존 연구를 통해서 알 수 있다(Diao *et al.*, 2000). 그러나 최근에 개발된 중심점 기반 분류 알고리듬은 성능이 나이브 베이지안 분류기보다 우수하다고 하는데(Han(Sam) *et al.*, 2000), 아직까지 전자메일을 이용한 두 분류기의 성능비교 연구는 이루어지지 않았다. 따라서 본 연구에서 구현하게 될 분류기로 중심점 기반 분류기와 나이브 베이지안 분류기를 선택하게 되었다.

성능비교 시 학습집합의 구성을 달리하여 학습집합 내의 특정 단어에 의한 성능향상을 피함으로써, 얼마나 안정적인 성능을 유지하는가에 대한 실험과 학습집합의 크기를 줄이면서 성능의 변화를 실험한다. 전자메일 분류란 사전에 정해진 클래스에 전자메일을 할당하는 것으로, 사람이 사전에 학습을 위해서 전체의 전자메일을 수작업으로 분류해야 한다. 이러한 작업은 많은 시간과 비용을 요구하는 과정으로, 보다 적은 학습집합으로 뛰어난 성능을 가지는 분류기가 더 우수하다고 할 수 있다. 또한, 비즈니스 환경이 변화함에 따라 클래스 수의 변화도 불가피하다. 따라서, 클래스 수를 증가시키면서 성능의 변화를 실험해야한다. 이렇게 여러 가지 조건을 변화시켜 다각도로 두 분류기의 정확도 변화를 실험해 봄으로써 한글과 국내 기업에 적합한 전자메일 분류기를 구현하고자 한다.

### 1.3 연구의 구성

본 연구는 모두 5개의 장으로 구성되어 있으며, 각 장의 내용은 다음과 같다.

제 1장에서는 본 연구의 배경과 목적 및 방법, 그리고 연구의 구성 등 일반적인 사항을 기술하였다. 제 2장은 문서 자동분류에 관한 선행 연구와 본 연구에서 구현하게 되는 분류기인 나이브 베이지안 분류기와 중심점 기반 분류기를 자세하게 소개하였다. 또한 각 분류기의 개념과 특징을 기존 연구자료를 토대로 정리하였다.

제 3장에서는 데이터 정제와 전처리 과정 및 본 연구에서 직접 구현된 두 가지 분류기, 그리고 실험자료에 대한 내용을 포함하고 있으며, 제 4장에서는 실험결과를, 그리고 제 5장에서는 결론 및 향후 연구과제를 제시하였다.

## 2. 선행연구

### 2.1 문서 자동분류

문서 자동분류란 특정 문서의 내용에 기반하여 컴퓨터가 자동으로 이 문서를 미리 정의되어 있는 분류목록에 할당하는 작업을 말하며, 문서 자동분류를 위한 학습 프로그램은 특정 문서가 적합한 분류목록에 들어갈 수 있도록 규칙을 배우게 된다(Han *et al.*, 2000; Liere and Tadepalli, 1996).

문서의 자동분류는 문서 영역의 특성상 그 작업이 매우 어렵다. 문서는 백터의 형태로 표현되어 있지 않으며, 수많은 특징이 존재하고, 각 문서별로 많은 변화를 일으킬 가능성을 내포하고 있다. 우리가 관심을 가지는 것은 문서 내에 포함된 단어이다. 그렇지만 문서는 자연언어로 쓰여져 있으므로 많은

중의성을 포함하고 있다. 이는 문서영역에 에러 공간이 상당히 넓게 분포하고 있음을 의미하며 숫자, 생략기호, 약어 등이 이러한 에러 공간에 해당한다. 이렇게 볼 때 고차원적이고 에러 공간이 넓은 영역의 특성상 문서 분류를 기계학습을 통해 수행하는 것은 그만큼 어려움이 존재한다(Han *et al.*, 2000).

문서의 분류를 기계학습에 적용한 Apte와 Damerau에 의해 한 가지 주목할 만한 결과가 나왔는데, 이들은 규칙기반 증명을 통하여 Reuters-22173 집합에서 문서분류를 실시한 결과 80.5%의 정확률(precision)과 재현율(recall)을 얻어냈다(Apte. C. and Damerau, 1994). 이 결과를 얻기 위하여 10,000개의 정리된 학습 데이터가 사용되었다. 일반적으로 감독학습(supervised learning)을 통해서 좋은 결과를 얻기 위해서는 수천 개 정도의 많은 정리된 데이터가 필요하다. Castelli와 Cover는 정리된 데이터와 정리되지 않은 데이터의 상대적 가치를 베이지안 정리를 통해서 계산해 보았을 때, 정리된 데이터가 정리되지 않은 데이터보다 지수적으로 가치가 증가한다는 결론을 얻었다(Castelli. V. and Cover. M. T., 1995). 이는 정리된 데이터를 가지고 학습을 수행하는 것이 학습에 소요되는 계산적 노력을 줄여준다는 사실을 나타내고 있다. 그러나 문서는 그 자체로 정리되어 있지 않으며 이 문서의 정리작업은 사람의 몫으로 남게 된다. "지수적으로 더욱 가치가 있다"라는 측면에서 보면 효율성을 위해 전처리과정을 거쳐야 됨을 알 수 있으며, 이것은 돈과 시간의 투자 그리고 사람의 노력으로 이루어져야 한다(Han *et al.*, 2000).

현재 대부분의 문서 자동분류에 대한 연구는 규칙기반, 확률기반, 통계 및 학습 기반으로 이루어지고 있으며(Ko and Lee, 2001), 주로 이용되고 있는 알고리듬은 신경망 (Neural Networks), 베이지안 확률, K-최근점 (K-Nearest Neighbor), 의사결정나무 (Decision Tree), 서포트 벡터 머신 (Support Vector Machine) 등이 있다. 이들 중에서 베이지안 확률을 사용한 문서분류는 각 언어권에서 일반적으로 높은 분류효율을 나타내는 방법이다(Ko and Lee, 2001).

그러나 최근 Han과 Karypis에 의해 연구된 결과에서 중심점을 이용한 문서분류 알고리듬이 나이브 베이지안 분류기보다 더 우수한 정확도를 보였다(Han(Sam) and Karypis, 2000).

전자메일 분류와 관련된 연구로는 개인의 흥미 유무에 관한 분류, 스팸메일 필터링에 관한 것이 대부분이다. 그 예로, Cohen은 RIPPER를 이용하여 개인 전자메일 필터를 구현하였으며(Cohen, 1996), Sahami *et al*은 나이브 베이지안 분류기를 이용하여 스팸메일 필터를 구현하였다(Dumais *et al.,* 1998). 개인의 흥미유무에 관한 필터링으로써, Dial *et al*(2000)이 나이브 베이지안 분류기와 의사결정나무 분류기를 이용하여 분류한 결과 나이브 베이지안 분류기가 더 안정적이고, 우수한 성능을 보였다(Diao *et al.*, 2000).

국내 연구로서, 황호순은 속성추출방법을 이용하여 정보력이 뛰어난 속성들만을 추출하고, 나이브 베이지안 분류기로 스팸메일 필터링이 아닌 온라인 쇼핑몰의 고객문의 메일을 분류하였다(Hwang, H. S., 2001). 또한 윤종식은 약학습자를 결합하는 방식인 배깅(bagging)과 부스팅(boosting)을 이용하여 나이브 베이지안 분류기의 전자메일 분류성능을 향상시켰다(Yoon, 2001).

전자메일을 포함한 문서의 자동분류에 대한 연구 역시, 다른 정보검색에 관련된 연구들처럼 오래 전부터 진행되어 왔지만 이는 거의 모두가 영어권의 나라에서 이루어진 것이 대부분이다. 각 나라의 언어적인 특성을 고려해 볼 때, 외국의 연구들을 그대로 한국어 문서자료 집합에 적용하는 것은 문제가 있기 때문에 한국어의 특성에 맞는 자동분류 시스템에 대한 연구가 필요하다. 그러나 정보검색의 효율향상을 위한 다른 연구들과는 달리 문서 자동분류에 대한 연구는 아직까지 미흡한 상태이다(Hur *et al.*, 2001).

따라서 본 연구에서는 전자메일 분류를 위해서 전자메일을 포함한 다른 모든 문서분류에서 우수한 성능을 보이는 나이브 베이지안 분류기와 최근 연구에서 나이브 베이지안 분류기보다 더 우수하다는 중심점 기반 분류기를 이용하여 전자메일을 분류하고 성능을 비교해 보고자 한다.

### 2.2 나이브 베이지안 분류기

나이브 베이지안 분류기는 다양한 베이지안 분류기들 중 가장 간단한 모델이다. 그것은 "나이브 베이즈 가정" 이라고 불리는, 문서 내 모든 속성들이 주어진 클래스 내에서 서로 독립이라는 가정 때문이다. 이 가정이 대부분의 현실 세계의 문제를 해결함에 있어서 명백한 거짓임에도 불구하고, 나이브 베이지안 분류기는 종종 매우 좋은 분류성능을 보여 준다(McCallum and Nigam, 1998). 이러한 모순은 이진분류의 경우, 분류추정(classification estimation)이 함수 추정(function estimation)에서 단지 함수의 부호(sign)를 추정하는 것과 같다는 사실로 설명될 수 있다. 다시 말해서, 분류정확도는 높은 반면에 함수근사(function approximation)는 여전히 미흡할 수 있다(Yang. Y., 1994). 이러한 독립 가정 때문에, 속성의 수가 많을 때, 각 속성의 모수들은 분리해서 학습할 수 있는데, 이것이 학습을 간단하게 한다(McCallum and Nigam, 1998; Mitchell, 1997).

문서분류는 많은 수의 속성을 가진 도메인과 같다. 여기서의 속성은 단어가 되며, 서로 다른 단어의 수는 매우 클 수 있다. 단어 크기가 100개 이하인 몇몇 간단한 문서분류 작업은 정확하게 수행될 수 있는 반면에, 수천 개의 단어로 구성된 웹, 유즈넷, 전자문서의 실제 데이터에 관한 작업은 매우 복잡하다. 나이브 베이지안 분류기는 많은 연구결과에서 문서분류에 성공적으로 적용되고 있다(Yoon, 2001; Hwang, 2001; Cohen, 1996; Dumais *et al.,* 1998; Lewis and Ringuette, 1998; McCallum and Nigam, 1998).

많은 연구에서 나이브 베이지안 분류기가 이용되고 있지만, 일반적으로 서로 다른 생성모델을 가지는 두 가지의 분류기 때문에 혼동이 있다. 또한 두 가지 분류기 모두 나이브 베이지

안 분류기라고 부른다(McCallum and Nigam, 1998).

첫 번째 모델은 문서에서 단어가 존재하는지, 그렇지 않은지만을 가리키는 이진속성 벡터에 의해서 문서를 표현한다. 문서에서 단어의 발생빈도는 고려되지 않고 문서의 확률을 계산할 때, 문서에서 발생하지 않은 단어의 확률을 포함해서 모든 속성값들의 확률을 곱한다. 여기서, 우리는 문서를 "사건(event)"으로 이해할 수 있고, 단어의 존재 유무는 그 사건의 속성으로 고려할 수 있다. 이것은 다변량 베르누이 이벤트(multivariate Bernoulli event) 모델에 근거한 분포를 설명하고 있다. 이것은 베이지안 네트워크 영역에서 보다 더 전통적인 접근 기법이며, 속성의 수가 고정된 분야에 적절하다(McCallum and Nigam, 1998).

두 번째 모델은 문서에서 단어가 발생한 빈도에 의해 문서를 표현한다. 위와 같이, 단어의 순서는 고려하지 못하지만, 문서에서 각 단어의 발생빈도는 문서의 확률을 계산할 때, 발생한 단어의 확률값을 곱하는 방법으로 고려할 수 있다. 여기서 우리는 개별적인 단어의 발생을 "사건"으로 이해할 수 있고, 문서는 사건의 집합으로 생각할 수 있다. 이것은 다항식 이벤트(multinomial event) 모델이라고 부른다. 이것은 음성인식과 같은 모델링보다 일반적인 기법이다(McCallum and Nigam, 1998).

이처럼 베이지안 프레임워크 안에서는 서로 다른 두 가지 생성모델을 가진 나이브 베이지안 분류기가 존재한다. 그러나, 연구결과에 따르면 단어의 크기가 크거나 단어들이 적절하게(optimally) 선택되었을 경우 다항식 모델이 다변량 베르누이 모델보다 우수한 성능을 보인다고 알려져 있다(Diao *et al.*, 2000; McCallum and Nigam, 1998). 따라서 우리는 나이브 베이지안 분류기 구현 시 다항식 모델을 이용하여 전자메일 분류를 수행한다.

이 논문에서 사용될 기호로 클래스는 $C=\{c_1, c_2, \ldots, c_k\}$와 같으며, $k$는 클래스의 수를 나타내고, 전자메일과 같은 문서는 $D=\{d_1, d_2, \ldots, d_m\}$로 표현한다. 여기서 $m$은 문서의 개수를 나타내고, 각 문서를 구성하는 단어, 즉 속성들은 $W=\{w_1, w_2, \ldots, w_n\}$로 표현하며, $n$은 속성의 개수를 나타낸다. 따라서 문서 $d'$의 클래스는 다음 식 (1)에 의해서 결정될 수 있다.

$$C=\arg\max_k P(c_k|d')=\arg\max_k P(d'|c_k)P(c_k) \quad (1)$$

문서가 속성들의 집합 $\{w_1, \ldots, w_n\}$으로 표현 가능하고, 문서에서 발생하는 각 속성들은 서로 독립이라는 나이브 베이즈 가정을 이용하여 식 (1)을 다음 식 (2)와 같이 표현할 수 있다.

$$C=\arg\max_k P(w_1|c_k)P(w_2|c_k)\ldots P(w_n|c_k)P(c_k) \quad (2)$$

주어진 학습집합을 이용하여, 학습과정에서는 모든 모수들의 베이즈-최적 추정값(bayes-optimal estimates)을 계산하게 된다. 여기서 클래스 $k$가 주어졌을 때, 속성 $w_j$의 확률값과 각 클래스의 사전 확률값의 추정은 식 (3)과 식 (4)로부터 계산될 수 있다.

$$P(w_j|c_k)=\frac{1+\sum_{i=1}^{|D|}N(w_j, d_i)P(c_k|d_i)}{|V|+\sum_{t=1}^{|V|}\sum_{i=1}^{|D|}N(w_t, d_i)P(c_k|d_i)} \quad (3)$$

$$P(c_k)=\frac{\sum_{t=1}^{|V|}\sum_{i=1}^{|D|}N(w_t, d_i)P(c_k|d_i)}{\sum_{k=1}^{K}\sum_{t=1}^{|V|}\sum_{i=1}^{|D|}N(w_t, d_i)P(c_k|d_i)} \quad (4)$$

여기서 $N(w_j, d_i)$는 문서 $d_i$에서 속성 $w_j$의 발생빈도이며, $P(c_k|d_i)=\{0,1\}$은 각 문서의 클래스에 의해서 결정된다. $|D|$는 학습문서의 수를 나타내고, $\sum_{t=1}^{|V|}P(w_t|c_k)=1$이다. 학습 데이터에서 나타나지 않은 속성의 확률을 교정하기 위해서, 1을 더하는 평활(smoothing)기법이 사용되었다. 여기서 $|V|$는 단어의 개수를 나타낸다(Diao and Wu, 2000).

### 2.3 중심점 기반 분류기

중심점 기반 분류 알고리듬에서, 문서들은 벡터-공간 모델을 이용하여 표현된다. 이 모델에서, 각 문서 $d$는 단어-공간(term-space)에서 벡터로 간주된다. 이러한 단순한 형태에서, 각 문서는 단어-빈도(TF) 벡터 $\overrightarrow{d_{ft}}=(tf_1, tf_2, \ldots, tf_n)$으로 표현되며, 여기서 $tf_i$는 문서에서 $i$번째 단어의 빈도를 나타낸다. 이 모델에서 주로 이용되는 방법으로 각 단어에 문서의 역빈도(inverse document frequency)를 곱하는 것이다. 이와 같은 가중치를 주는 목적은 모든 문서에서 자주 나타나는 단어는 문서를 분류할 수 있는 능력인 분별력(discrimination power)이 제한되어 있기 때문이다. 따라서 이러한 단어들의 중요도를 감소시키는 것이다. 이것은 주로 단어 $i$의 빈도에 $\log(N/df_i)$를 곱한다. 여기서 $N$은 전체 문서의 수를 말하고, $df_i$는 $i$번째 단어를 포함하고 있는 문서의 수(document frequency)를 말한다.

$$\vec{d}_{tfidf}=(tf_1\log(N/df_1), tf_2\log(N/df_2), \ldots, tf_n\log(N/df_n)) \quad (5)$$

따라서 문서는 식 (5)와 같은 $tf-idf$의 형태로 표현될 수 있다. 마지막으로, 서로 다른 문서의 길이를 고려하기 위해서 각 문서 벡터의 길이가 1이 되도록 각 문서 벡터를 정규화시킨다. 즉, $\|\vec{d}_{tfidf}\|_2=1$이다[13].

벡터-공간 모델에서, 두 문서 $d_i$와 $d_j$ 간의 유사도는 다음 식 (6)과 같은 코사인(cosine) 함수를 이용하여 주로 계산된다(Salton, 1989).

$$\cos(\vec{d}_i, \vec{d}_j)=\frac{\vec{d}_i\cdot\vec{d}_j}{\|\vec{d}_i\|_2\times\|\vec{d}_j\|_2} \quad (6)$$

여기서, 문서 벡터들은 길이가 1인 단위 벡터이므로, 위의 식

(6)은 $\cos(\vec{d}_i, \vec{d}_j) = \vec{d}_i \cdot \vec{d}_j$와 같이 간단하게 정리할 수 있다.

문서들의 집합 $S$가 벡터의 형태로 표현되어 있다면, 우리는 중심점 벡터를 식 (7)과 같이 정의한다.

$$\vec{C} = \frac{1}{|S|} \sum_{\vec{d} \in S} \vec{d} \tag{7}$$

식 (7)에서 $S$를 중심점 벡터 $\vec{C}$에 대한 지지 집합(supporting set)이라고 부른다. 두 개의 중심점 벡터 간의 유사도와 문서 벡터와 중심점 벡터 간의 유사도는 문서 벡터 간의 유사도 계산과 비슷하게 코사인 함수를 이용하여 계산될 수 있다. 중심점 벡터 간의 유사도는 식 (8)과 같고, 중심점 벡터와 문서 벡터 간의 유사도는 식 (9)와 같다.

$$\cos(\vec{C}_i, \vec{C}_j) = \frac{\vec{C}_i \cdot \vec{C}_j}{\| \vec{C}_i \|_2 \times \| \vec{C}_j \|_2} \tag{8}$$

$$\cos(\vec{d}, \vec{C}) = \frac{\vec{d} \cdot \vec{C}}{\| \vec{d} \|_2 \times \| \vec{C} \|_2} = \frac{\vec{d} \cdot \vec{C}}{\| \vec{C} \|_2} \tag{9}$$

중심점 기반 분류기의 아이디어는 매우 간단하다. 동일한 클래스에 속하는 각 문서들의 집합에 대해서, 중심점 벡터를 계산한다. 만약 k개의 클래스가 있다면, k개의 중심점 벡터 $\{ \vec{C}_1, \vec{C}_2, \ldots, \vec{C}_k \}$를 계산하게 된다. 여기서 $\vec{C}_i$는 $i$번째 클래스의 중심점 벡터를 말한다. 먼저, 테스트 문서 $x$를 단어들의 빈도와 역빈도를 이용하여 $tf-idf$ 벡터로 표현하고, 문서 벡터의 길이가 1이 되도록 정규화한다. 그런 후, 코사인 함수를 이용하여 테스트 문서와 인접한 k개의 중심점 벡터들 간의 유사도를 계산하여, 가장 큰 유사도를 가지는 중심점의 클래스를 문서에 할당하게 된다. 즉, 식 (10)과 같다.

$$\underset{j=1,\ldots,k}{\arg\max} (\cos(\vec{x}, \vec{C}_j)) \tag{10}$$

중심점 기반 분류기의 학습단계에서의 계산상의 복잡도는 문서의 수와 학습집합에 포함된 단어의 수에 선형이다. 게다가 새로운 문서 $x$를 분류하는 데 필요한 시간은 거의 $O(km)$이다. 여기서 $m$은 $x$에 나타난 단어의 수를 나타낸다. 따라서, 이 알고리듬의 전반적인 계산상의 복잡도는 매우 낮고, 나이브 베이지안 분류기와 같은 문서분류기만큼 빠르다[13].

이 분류모델을 좀더 잘 이해하기 위해서 다음과 같은 간단한 이진 분류 알고리듬을 생각해 보자. A, B 두 개의 클래스가 있다고 가정하고, $\overline{S}_A$는 A에 있는 아이템들 간의 평균 유사도를 말하고, $\overline{S}_B$는 B에 있는 아이템들 간의 평균 유사도를 말하며, $\overline{S}_{A,B}$는 $a \in A$이고, $b \in B$인 모든 아이템들(a,b) 간의 평균 유사도를 말한다고 하자. 테스트 아이템을 $x$라 하고, A와 B내의 모든 아이템들과 $x$ 간의 평균 유사도를 각각 $\overline{S}_{x,A}$와 $\overline{S}_{x,B}$라고 하자. 이러한 설정은 아래 <그림 2>에 나타나 있다. 이 분류기에서 $x$는 평균 유사도에 의해 측정된 특성과 클래스 A, B의 아이템들의 특성이 얼마나 유사한가에 기반하여 A나 B로 분류되어질 것이다.

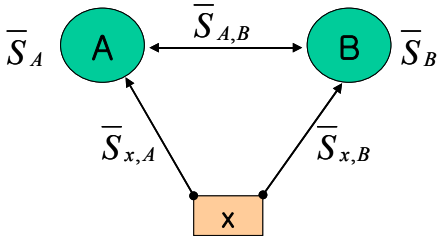



**그림 2.** 간단한 이진분류기.

이러한 작용은 비율 $\overline{S}_A / \overline{S}_{A,B}$와 $\overline{S}_B / \overline{S}_{A,B}$를 조사해서, $\overline{S}_{x,A} / \overline{S}_{x,B}$와 $\overline{S}_{x,B} / \overline{S}_{x,A}$와 비교해 봄으로써 모형화시킬 수 있다. 첫 번째 비율 $\overline{S}_A / \overline{S}_{A,B}$는 클래스 A에 속하는 아이템들 간의 내부 유사도가 클래스 A와 B에 속하는 아이템들 간의 유사도에 비해 상대적으로 얼마나 강한가를 측정한다. 비슷하게, 두 번째 비율 $\overline{S}_B / \overline{S}_{A,B}$는 클래스 B에 속하는 아이템들 간의 내부 유사도가 클래스 A와 B에 속하는 아이템들 간의 유사도에 비해 상대적으로 얼마나 강한가를 측정한다. 그리고, 마지막 2개 비율은 $x$가 클래스 B의 아이템들에 비해 클래스 A의 아이템들과의 유사도가 얼마나 강한가를 측정한다. 그리고 그 역도 마찬가지다. 위와 같은 비율들이 주어졌다면, 분류 알고리듬은 다음 식 (11)의 조건을 만족하는 $x$를 클래스 A로 분류하게 된다. 만약 그렇지 않으면, $x$를 클래스 B로 분류하게 될 것이다(Han(Sam) and Karypis, 2000).

$$\frac{\overline{S}_{x,A} / \overline{S}_{x,B}}{\overline{S}_A / \overline{S}_{A,B}} \geq \frac{\overline{S}_{x,B} / \overline{S}_{x,A}}{\overline{S}_B / \overline{S}_{A,B}} \tag{11}$$

위 식 (11)에서 부등식의 양변에 나타난 $\overline{S}_{A,B}$를 상쇄시키고, 간단한 대수조작을 취하면 다음과 같은 관계를 얻을 수 있다.

$$\frac{\overline{S}_{x,A} / \overline{S}_{x,B}}{\overline{S}_A / \overline{S}_{A,B}} \geq \frac{\overline{S}_{x,B} / \overline{S}_{x,A}}{\overline{S}_B / \overline{S}_{A,B}} \Rightarrow \frac{\overline{S}^2_{x,A}}{\overline{S}_A} \geq \frac{\overline{S}^2_{x,B}}{\overline{S}_B}$$

$$\Rightarrow \frac{\overline{S}_{x,A}}{\sqrt{\overline{S}_A}} \geq \frac{\overline{S}_{x,B}}{\sqrt{\overline{S}_B}} \tag{12}$$

토너먼트 방법을 사용하면 2개 이상의 클래스 문제에도 확장시켜 적용시킬 수 있다. 따라서, $x$에는 $\overline{S}_{x,j} / \sqrt{\overline{S}_j}$의 값이 가장 큰 클래스 $j$를 할당하게 된다. 앞에서 언급했듯이, 위의 문제에서 데이터 아이템들은 길이가 1인 문서 벡터들이다. 그리고 유사도는 코사인 함수에 의해서 계산된다. 따라서 식 (12)로부터, 다음을 만족하는 경우 $x$를 클래스 A로 분류하게 된다.

$$\cos(\vec{x}, \vec{C}_A) \geq \cos(\vec{x}, \vec{C}_B) \tag{13}$$

그렇지 않으면 $x$는 클래스 B로 분류하게 된다. 여기서 $\vec{C}_A$와 $\vec{C}_B$는 각각 클래스 A, B의 중심점 벡터이다(Han(Sam) and Karypis, 2000).

## 3. 실험방법

본 연구에서는 나이브 베이지안 분류기와 중심점 기반 분류기를 구현함으로써 온라인 쇼핑몰 업체와 카드 회사의 고객문의 메일을 분류할 시스템을 구현하고, 그 성능을 평가해 보고자 한다.

### 3.1 실험 데이터

3.1.1 온라인 쇼핑몰 전자메일

본 자료는 실제 온라인 쇼핑몰 업체로 들어오는 고객문의메일로서 메일 성격에 따라 시스템 문의사항, 상품 문의사항, 사업 문의사항의 3가지 클래스로 분류된다. 시스템 문의는 패스워드 분실이나 메일에서 글씨가 깨지는 문제 등이며, 상품 문의는 상품의 재고 여부나 색깔, 사이즈에 대한 내용과 주문 확인 등이나 주문 취소에 관련된 내용이다. 또한 사업문의는 배너광고 교환, 제휴, 업체입주 등에 관련된 내용이 포함된다. 전체 399개의 메일 중 카테고리별로 시스템문의 사항에 관련된 메일이 189개, 상품 문의사항에 관련된 메일이 149개, 사업 문의사항에 관련된 메일이 61개로 구성되어 있다. 이 자료는 학습집합의 구성에 따른 분류기의 성능 변화를 실험하기 위해서 10번 반복적으로 학습집합과 테스트집합을 70 : 30으로 랜덤하게 분리하였다. 또한 학습집합의 크기가 변화되면서 성능의 변화를 알아보기 위해서 학습집합과 테스트집합을 50 : 50의 비율, 30 : 70의 비율로 각각 10번 반복적으로 분리하여 실험 데이터를 구성하였다.

**표 1.** 학습집합의 비율이 70%인 경우의 실험데이터

(단위 : 개)

| 분 류 | 시스템문의 | 상품문의 | 사업문의 | 합 계 |
|---|---|---|---|---|
| 학습집합 | 130 | 100 | 40 | 270 |
| 테스트집합 | 59 | 49 | 21 | 129 |
| 합 계 | 189 | 149 | 61 | 399 |

**표 2.** 학습집합의 비율이 50%인 경우의 실험데이터

(단위 : 개)

| 분 류 | 시스템문의 | 상품문의 | 사업문의 | 합 계 |
|---|---|---|---|---|
| 학습집합 | 94 | 75 | 31 | 200 |
| 테스트집합 | 95 | 74 | 30 | 199 |
| 합 계 | 189 | 149 | 61 | 398 |

**표 3.** 학습집합의 비율이 30%인 경우의 실험데이터

(단위 : 개)

| 분 류 | 시스템문의 | 상품문의 | 사업문의 | 합 계 |
|---|---|---|---|---|
| 학습집합 | 59 | 49 | 21 | 129 |
| 테스트집합 | 130 | 100 | 40 | 270 |
| 합 계 | 189 | 149 | 61 | 399 |

3.1.2 카드회사 전자메일

본 자료는 카드회사로 들어오는 고객들의 문의메일로서 실제 카드회사의 업무영역을 고려하여 7개의 클래스로 분리하였다. 전체 1415개의 메일 중 결제, 결제일 관련 메일(class1)이 227개, 승인내역, 거래내역, 승인취소와 관련된 메일(class2)이 189개, 연체, 할부, 대출, 현금서비스, 수수료 관련 메일(class3)이 242개, 제휴카드, 제휴서비스, 부가서비스 관련 메일(class4)이 212개, 홈페이지 이용과 관련된 문의메일(class5)이 135개, 카드의 종류변경과 해지에 관련된 메일(class6)이 205개, 카드 신규발급, 분실/훼손 재발급과 관련된 메일(class7)이 205개로 구성되어 있다. 이 자료는 클래스 수를 증가시키면서 분류기의 성능 변화를 실험하는 데 사용된다. 전체 데이터의 70%는 학습을 위해서, 30%는 평가를 위해서 사용하였다.

**표 4.** 카드회사 전자메일 실험데이터

(단위 : 개)

| 분류 | class1 | class2 | class3 | class4 | class5 | class6 | class7 | 합계 |
|---|---|---|---|---|---|---|---|---|
| 학습집합 | 160 | 132 | 170 | 150 | 95 | 145 | 145 | 997 |
| 테스트집합 | 67 | 57 | 72 | 62 | 40 | 60 | 60 | 418 |
| 합계 | 227 | 189 | 242 | 212 | 135 | 205 | 205 | 1,415 |

### 3.2 실험방법

분류실험을 하기 전에 데이터에 대해서 전처리과정을 수행하게 된다. 전처리과정이란 영어에서 어간추출(stemming) 방법을 사용함으로써 데이터를 정제하는 과정을 말한다. 예를 들면 'school', 'schools'는 동일한 단어로 간주하게 된다. 본 논문에서도 '학교들', '학교'를 동일한 단어로 인식하도록 하는 방법을 취함으로써 정제작업을 수행하였다. 또한 특별한 의미를 가지고 있지 않는 단어를 제거하였다. 이런 단어를 불용어라고 부르는데, 영어에서는 'a', 'the', 'in', 'of', 'and', 'or', 'that' 등을 말하며, 한글에서는 '예를 들면', '그래서', '그리고', '도대체', '수많은', '빨리' 등과 같은 단어를 의미한다. 그리고, &, ?, #, !, ^, -, ; 와 같은 특수문자와 이들의 조합을 또한 제거하였다. 본 연구에서 사용하는 실험데이터인 전자메일은 유즈넷, 웹, 전자저널과 달리 은어, 비어, 속어, 줄임말, 특히 맞춤법에 맞지 않는 단어들을 포함하고 있어 실험을 수행하기에 앞서

정제하는 데 많은 노력과 시간이 요구되었다(Hwang, 2001).

본 연구의 실험을 위해서 사용된 나이브 베이지안 분류기와 중심점 기반 분류기는 C 언어를 이용하여 직접 구현하였다.

### 3.2.1 나이브 베이지안 분류기

본 연구에서 구현된 나이브 베이지안 분류기는 <그림 3>과 같이 학습단계와 분류단계로 구성되어 있다.

먼저 학습단계에서는 어간추출, 불용어 제거, 특수문자 제거와 같은 전처리과정을 거친 다음, 각각의 클래스별로 학습집합을 구성한 후 유일한(unique) 단어들의 문서에서 발생빈도를 계산함으로써, 각 단어들과 클래스의 확률을 계산하게 된다.

분류단계에서는 학습단계와 같은 전처리과정을 거친 후, 학습단계에서 계산된 확률값을 이용하여 전자메일이 클래스에 속할 확률을 계산하고, 그 확률값이 가장 큰 클래스로 분류하게 된다(Hwang, 2001).

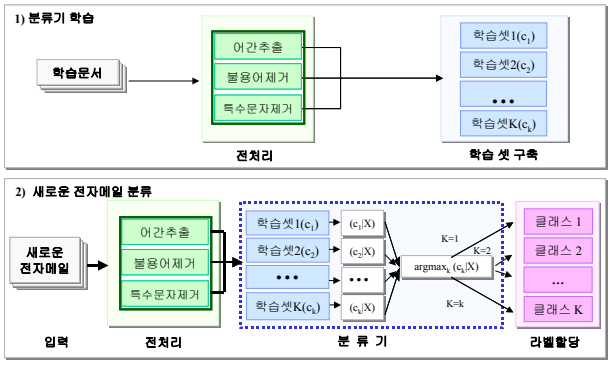



**그림 3.** 나이브 베이지안 분류기를 이용한 전자메일 분류.

### 3.2.2 중심점 기반 분류기

본 연구에서 구현된 중심점 기반 분류기는 <그림 4>를 통해서 알 수 있듯이 나이브 베이지안 분류기와 같은 학습단계와 분류단계로 구성되어 있다.

학습단계에서는 나이브 베이지안 분류기와 동일한 전처리과정을 거친다. 이 단계에서 얻은 유일한(unique) 단어들의 빈도(TF)와 역빈도(IDF)를 이용하여 TFIDF 형태의 벡터로 표현한 후, 벡터의 크기를 1로 정규화시킨다. 이렇게 정규화된 벡터로 표현된 학습집합을 이용하여 각 클래스의 중심점을 계산한다.

분류단계에서는 학습단계와 같이 전처리과정, TFIDF 변환과 정규화를 거친 후, 코사인 함수를 이용하여 테스트 전자메일과 각 클래스의 중심점과의 유사도를 계산하게 된다. 최종적으로 가장 큰 유사도를 가지는 클래스로 메일을 분류하게 된다.

텍스트 문서분류 시스템의 성능평가에 관한 다양한 방법이 연구되고 있다. 이런 연구들 중에서 가장 많이 사용되고 있는 측정방법은 시스템의 효과성(effectiveness)을 측정하는 정확도(accuracy), 정확률(precision), 재현율(recall), 그리고 에러(error)이다. 클래스가 k개일 때, 정확도, 정확률, 재현율, 그리고 에러는 다음과 같다(Hwang, 2001; Diao *et al.*, 2000).

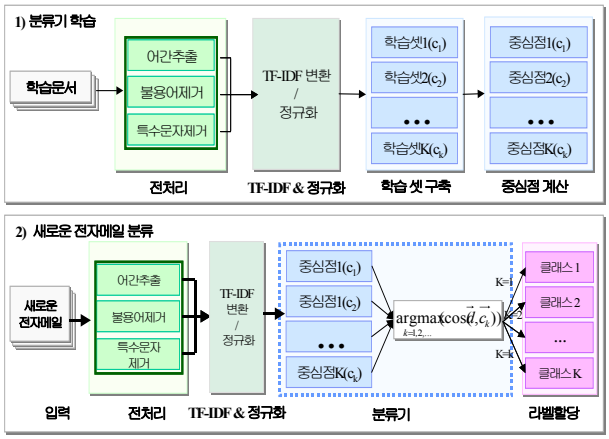



**그림 4.** 중심점 기반 분류기를 이용한 전자메일 분류.

$$\text{에러(error)} = \frac{\text{오분류 수}}{\text{분류된 메시지 수}}$$

정확도(accuracy) = 1 - 에러

$$C_k\text{의 정확률(precision)} = \frac{C_k\text{를 } C_k\text{으로 분류한 메시지}}{C_k\text{으로 분류된 } total \text{ 메시지 수}}$$

$$C_k\text{의 재현율(recall)} = \frac{C_k\text{를 } C_k\text{으로 분류한 메시지}}{C_k\text{에 속한 } total \text{ 메시지 수}}$$

본 연구에서는 정확도 관점에서 두 시스템의 성능을 평가하였는데, 그 이유는 현업에서 여러 종류의 전자메일 분류기를 구현하여 적용하고 있지만, 이러한 분류기가 시간이 지남에 따라 정확도가 떨어진다는 것이 문제가 되고 있다. 따라서 본 연구에서는 정확도에 입각해 다각도로 실험하면서 두 분류기의 성능을 비교 / 평가하였다.

## 4. 실험결과 및 고찰

### 4.1 온라인 쇼핑몰 전자메일 실험결과

(클래스 수 : 3개)

#### 4.1.1 학습집합의 비율이 70%인 경우의 실험결과

나이브 베이지안 분류기(NBC)와 중심점 기반 분류기(CBC)를 이용하여 온라인 쇼핑몰 업체의 전자메일 분류실험을 하였다. 전자메일 데이터를 정제하는 과정이 비교적 시간과 노력이 많이 드는 힘든 과정이었지만 두 분류기 모두 신뢰할 만한 좋은 결과를 얻은 것을 알 수 있다. <표 5>에서 최상과 최하의 정확도를 볼드체로 표시하였다.

**표 5.** 학습집합의 비율이 70%인 경우의 실험결과

| 반복 횟수 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 평균 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CBC | 0.91 | 0.91 | 0.94 | **0.90** | **0.93** | 0.92 | **0.93** | **0.90** | 0.92 | 0.91 | 0.92 |
| NBC | **0.87** | 0.91 | **0.92** | 0.90 | 0.90 | **0.92** | **0.92** | 0.88 | 0.91 | 0.91 | 0.89 |
| 에러 차이(개) | -5 | 0 | -3 | -1 | -4 | 0 | -2 | -2 | -2 | 0 | -19 |

그러나 <표 5>를 통해서 두 분류기의 10회 반복실험의 정확도를 살펴보면 중심점 기반 분류기는 전반적으로 0.9 이상의 정확도를 유지하면서 평균적인 정확도가 0.92이다. 그런 반면에 나이브 베이지안 분류기의 경우 가장 나쁜 결과가 0.87, 가장 좋은 결과가 0.92로 학습집합의 구성에 따른 정확도의 차이가 크다는 것으로 알 수 있다. 또한 나이브 베이지안 분류의 평균적인 정확도는 0.89로, 중심점 기반 분류기보다 정확도 측면에서 0.3 정도 떨어지는 결과를 보이고 있다. <표 5>에서 에러 차이는 중심점 기반 분류기의 에러에서 나이브 베이지안 분류기의 에러를 뺀 값을 말한다.

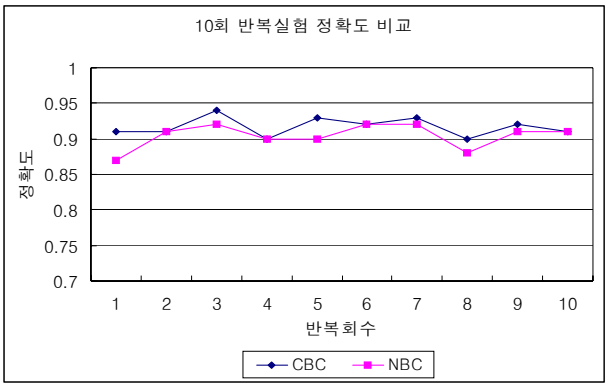



**그림 5.** 학습집합의 비율이 70%인 경우의 실험결과.

<그림 5>를 보면 위에서 설명한 것처럼 중심점 기반 분류기가 나이브 베이지안 분류기보다 10번의 실험에서 4번의 실험에서만 동일한 정확도를 가지는 반면 전반적으로 중심점 기반 분류기가 우수한 결과를 보여 주고 있다.

$H_0 : p=0$ v.s $H_1 : p\neq 0, \quad p=p_{CBC}-p_{NBC}$

$$t=\frac{\bar{p}}{\sqrt{S^2/n}} \sim t_{n-1,\alpha/2}$$

P : 에러율
n : 실험 반복 회수(10)
α : 유의수준(0.05)

$|t|>t_{9,0.975}=2.262$이면 $H_0$를기각

**그림 6.** 두 분류기 성능 가설검정 (유의수준 0.05).

본 연구에서는 두 분류기의 정확도 차이가 통계적으로 유의한지 T-검정을 실시하였다(Dietterich, 1998). 우선 에러율은 전체 테스트 문서와 두 분류기의 에러 차이의 비로 정의하였다. 그리고 귀무가설은 두 분류기의 에러율 차이는 없다는 것이다. 검정통계량은 $t=\frac{\bar{p}}{\sqrt{S^2/n}}$ 이며 이 값은 자유도가 $n-1$인 $t$ 분포를 따른다. 여기서 $\bar{p}$는 에러율의 평균을 나타내고, $S^2$은 에러율의 분산, 그리고 $n$은 실험 반복횟수를 나타낸다. 검정통계량이 $|t|=3.4716$으로 $t_{9,0.975}=2.262$보다 더 크므로 유의수준 0.05에서 귀무가설을 기각한다. 따라서 학습집합과 테스트집합을 70:30으로 분류한 실험에서 중심점 기반 분류기가 나이브 베이지안 분류기보다 더 우수한 성능을 보인다고 검정되었다.

4.1.2 학습집합의 비율이 50%인 경우의 실험결과

이번 실험은 학습집합과 테스트집합을 50 : 50 비율로 나누어서 10회 반복 실험하였다. 결과는 다음 <표 6>과 같다.

**표 6.** 학습집합의 비율이 50%인 경우의 실험결과

| 반복 횟수 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 평균 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CBC | 0.90 | 0.90 | 0.91 | 0.88 | 0.89 | 0.89 | 0.89 | 0.89 | 0.90 | 0.89 | 0.90 |
| NBC | 0.90 | 0.90 | 0.88 | 0.91 | 0.88 | 0.88 | 0.89 | 0.90 | 0.88 | 0.88 | 0.89 |
| 에러 차이(개) | 0 | -1 | -5 | 5 | -1 | -2 | 0 | 2 | -2 | -2 | -6 |

평균 정확도는 중심점 기반 분류기가 0.90이고 나이브 베이지안 분류기가 0.89로 거의 비슷한 성능을 보이며, 전반적으로 학습집합의 구성에 관계없이 안정적인 정확도를 보여주는 것을 알 수 있다.

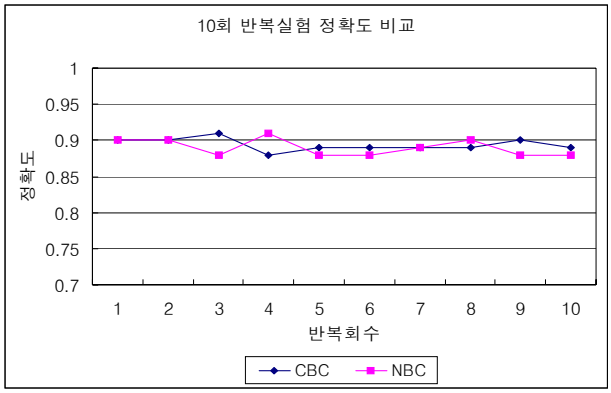



**그림 7.** 학습집합의 비율이 50%인 경우의 실험결과.

앞의 70:30으로 분류한 실험결과와 달리 <그림 7>을 보면 중심점 기반 분류기와 나이브 베이지안 분류기의 정확도가 거의 비슷한 결과를 보인다는 것을 알 수 있다. 또한 두 분류기의 정확

도 변화폭도 거의 차이가 없이 안정적인 결과를 보여주고 있다.

두 분류기의 정확도 차이가 통계적으로 유의한지 앞에서와 같은 방법으로 T-검정을 실시하였다. 검정통계량 값이 $|t| = 0.7099$로 $t_{9,0.975} = 2.262$보다 더 작으므로 유의수준 0.05에서 귀무가설을 기각할 수 없다. 따라서 학습집합과 테스트집합을 50:50으로 분류한 실험에서는 중심점 기반 분류기와 나이브 베이지안 분류기의 성능은 차이가 없다고 검정되었다.

4.1.3 학습집합의 비율이 30%인 경우의 실험결과

이번 실험은 학습집합과 테스트집합을 30 : 70 비율로 나누어서 10회 반복 실험하였다. 결과는 다음 <표 7>과 같으며, 최상과 최하의 정확도를 볼드체로 표시하였다.

**표 7.** 학습집합의 비율이 30%인 경우의 실험결과

| 반복 횟수 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 평균 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| CBC | **0.84** | 0.86 | **0.88** | **0.88** | 0.86 | 0.87 | 0.86 | 0.86 | 0.87 | 0.84 | 0.86 |
| NBC | 0.84 | 0.84 | **0.92** | 0.86 | 0.85 | 0.91 | 0.84 | 0.86 | 0.87 | **0.84** | 0.86 |
| 에러 차이(개) | 2 | -5 | 10 | -4 | -1 | 10 | -7 | -1 | 0 | 0 | 4 |

반복실험의 평균 정확도는 두 분류기 모두 0.86으로 동일한 정확도를 보인 것을 알 수 있다. 그러나 중심점 기반 분류기의 정확도는 0.85를 기준으로 거의 고른 성능을 보이는 반면 나이브 베이지안 분류기의 정확도는 0.84에서 최고 0.91까지 큰 폭의 차이를 가지면 변화가 심하다는 것을 알 수 있다. 따라서 나이브 베이지안 분류기는 중심점 기반 분류기보다 학습집합의 크기가 작아질 경우, 학습집합의 구성에 따라 정확도의 변화가 심하다는 것을 알 수 있었다.

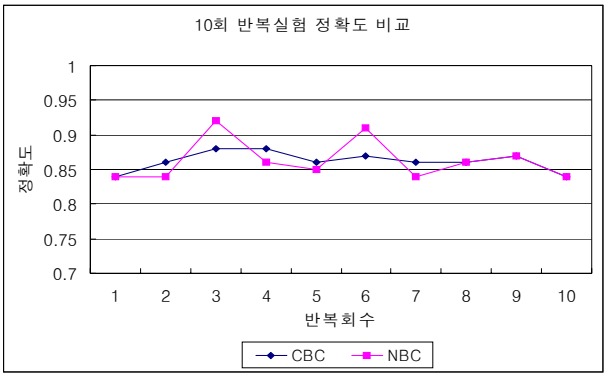



**그림 8.** 학습집합의 비율이 30%인 경우의 실험결과.

두 분류기의 정확도 차이가 통계적으로 유의한지 앞에서와 같은 방법으로 T-검정을 실시하였다. 검정통계량 값이 $|t| = 0.2210$로 $t_{9,0.975} = 2.262$보다 더 작으므로 유의수준 0.05에서 학습집합과 테스트집합을 30:70으로 분류한 실험에서는 중심점 기반 분류기와 나이브 베이지안 분류기의 성능은 차이가 없다고 검정되었다.

### 4.2 카드회사 전자메일 실험결과

실제 카드회사로 들어오는 고객의 전자메일을 나이브 베이지안 분류기와 중심점 기반 분류기를 이용하여 클래스 수를 변화시켜가면서 10회 반복실험한 후 <표 8>과 같은 결과를 얻었으며, 최상과 최하의 정확도를 볼드체로 표시하였다. <표 8>을 보면 대체적으로 두 분류기 모두 좋은 정확도를 보여주고 있다는 것을 알 수 있다. 그러나, 정확도의 평균을 보면 클래스 수가 3개인 경우는 두 분류기 모두 같은 결과를 얻지만 클래스 수가 4개, 5개인 경우에는 중심점 기반 분류기가 나이브 베이지안 분류기보다 조금 우수한 결과를 보여준다. 그러나 클래스 수가 6개, 7개인 경우에는 나이브 베이지안 분류기가 중심점 기반 분류기보다 좋은 결과를 보여주고 있다.

**표 8.** 카드회사 자료를 이용한 두 분류기의 정확도 비교

| 클래스 \ 실험 횟수 | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 평균 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 3 | CBC | **0.85** | 0.89 | 0.89 | 0.88 | 0.89 | 0.86 | 0.89 | 0.86 | 0.89 | **0.91** | 0.883 |
| | NBC | **0.84** | 0.89 | 0.89 | 0.89 | **0.91** | 0.87 | 0.88 | 0.86 | **0.91** | 0.90 | 0.883 |
| 4 | CBC | 0.87 | **0.91** | **0.85** | 0.89 | 0.88 | 0.89 | 0.86 | 0.87 | 0.86 | 0.88 | 0.875 |
| | NBC | 0.86 | **0.90** | **0.85** | 0.88 | **0.90** | 0.87 | 0.86 | 0.86 | 0.88 | 0.89 | 0.874 |
| 5 | CBC | 0.83 | **0.86** | **0.81** | 0.84 | 0.84 | 0.85 | 0.83 | 0.84 | 0.83 | **0.86** | 0.838 |
| | NBC | 0.82 | 0.85 | **0.80** | 0.84 | **0.86** | 0.81 | 0.82 | 0.81 | 0.84 | 0.85 | 0.828 |
| 6 | CBC | 0.80 | 0.80 | 0.78 | 0.80 | 0.80 | 0.80 | 0.80 | 0.78 | **0.77** | **0.81** | 0.791 |
| | NBC | 0.82 | 0.83 | 0.79 | 0.82 | **0.85** | **0.79** | 0.82 | 0.80 | 0.82 | **0.85** | 0.818 |
| 7 | CBC | 0.75 | 0.76 | 0.74 | 0.76 | 0.76 | 0.75 | 0.76 | **0.73** | 0.74 | **0.78** | 0.753 |
| | NBC | 0.78 | 0.78 | 0.76 | 0.79 | **0.80** | 0.76 | 0.79 | **0.75** | 0.78 | 0.79 | 0.777 |

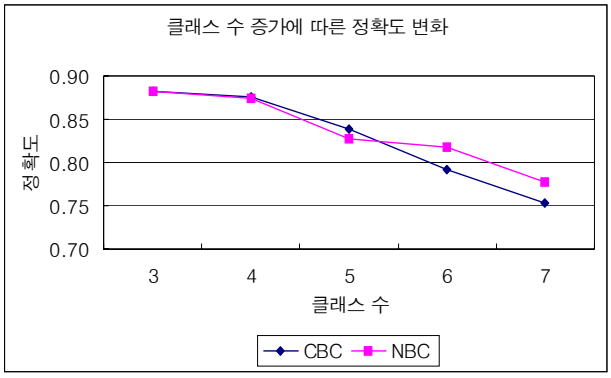



**그림 9.** 클래스 수 증가에 따른 정확도 변화 그래프.

<그림 9>는 두 분류기의 평균 정확도를 그래프로 나타낸 것이다. 두 분류기 모두 클래스 수가 증가하면서 정확도가 떨어지는 것을 볼 수 있는데, 이것은 모든 알고리듬에서 나타나는 일반적인 현상이다. 그러나 중심점 기반 분류기는 클래스

수가 증가하면서 그 성능의 변화가 나이브 베이지안 분류기보다 더 급격한 것을 볼 수 있다. 이것으로부터 나이브 베이지안 분류기가 중심점 기반 분류기보다 클래스 수의 증가에 따른 정확도의 변화폭이 크지 않다는 것을 알 수 있다.

<그림 10>에서 <그림 14>까지는 클래스가 3~7개인 각각의 경우에 10회 반복실험으로부터 얻은 두 분류기의 정확도 변화를 그래프로 본 것이다. 클래스 수가 6개와 7개인 경우에는 나이브 베이지안 분류기가 중심점 기반 분류기보다 매우 우수한 결과를 보인 것을 확인할 수 있다. 그러나 전반적으로 나이브 베이지안 분류기의 성능 변화가 심하다는 것을 알 수 있다. 특히 클래스 수가 증가하면서 그런 경향이 더 확실하게 나타나는 것으로 보아 나이브 베이지안 분류기는 중심점 기반 분류기보다 학습집합의 구성에 따른 성능 변화가 심하다는 것을 알 수 있다.

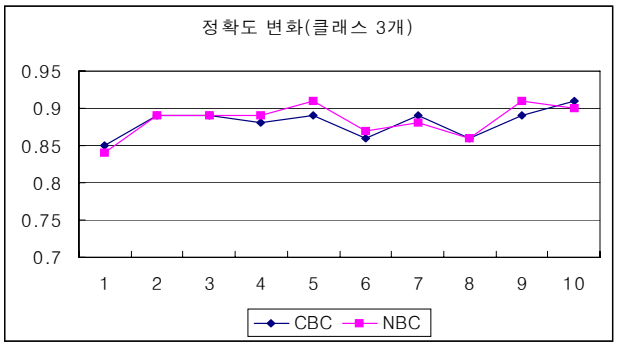



**그림 10.** 정확도 변화 그래프(클래스 3개).

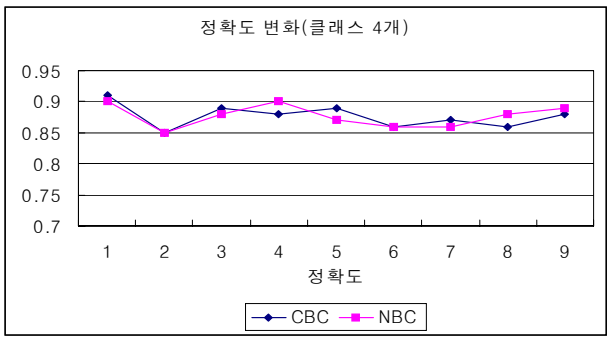



**그림 11.** 정확도 변화 그래프(클래스 4개).

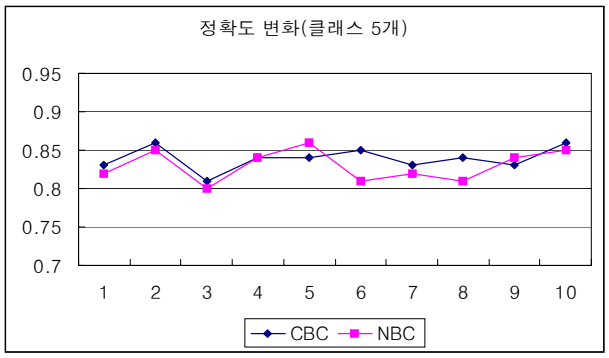



**그림 12.** 정확도 변화 그래프(클래스 5개).

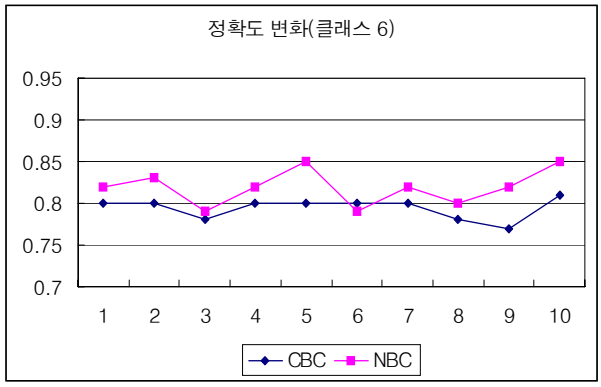



**그림 13.** 정확도 변화 그래프(클래스 6개).

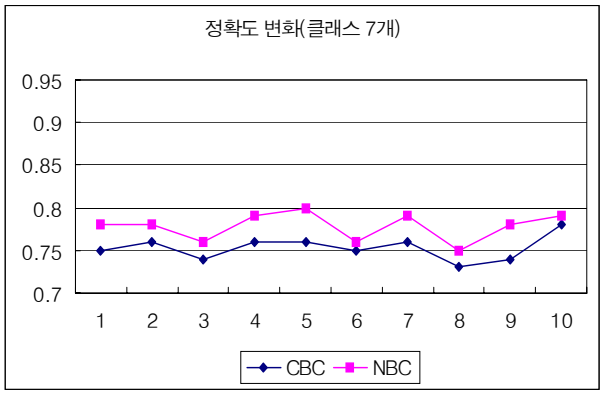



**그림 14.** 정확도 변화 그래프(클래스 7개).

**표 9.** 클래스 수 증가에 따른 두 분류기 성능 차이 검정

| 구 분 | 클래스 수 | | | | |
|---|---|---|---|---|---|
| | 3개 | 4개 | 5개 | 6개 | 7개 |
| 평 균 | 0 | -0.00078 | -0.01007 | 0.026536 | 0.024402 |
| 표준편차 | 0.012264 | 0.012365 | 0.01644 | 0.017181 | 0.010446 |
| 실험횟수 | 10 | 10 | 10 | 10 | 10 |
| T-값 | 0 | -0.62691 | -6.12372 | 15.44494 | 23.35946 |
| 기각역 | t9,0.975 = 2.262 (유의수준 0.05) | | | | |

각 클래스 수별 두 알고리듬의 성능 차이를 유의수준 0.05로 검정한 결과는 <표 9>와 같다. 클래스 수가 3개, 4개인 경우의 두 분류기의 성능 차이는 없는 것으로 검정되었으며, 클래스 수가 5개인 경우의 중심점 기반 분류기가 나이브 베이지안 분류기보다 더 좋은 결과를 보인다. 그러나 클래스 수가 6개, 7개인 경우는 나이브 베이지안 분류기가 중심점 기반 분류기보다 더 우수한 것으로 검정되었다.

## 5. 결론 및 향후 연구과제

본 논문에서는 나이브 베이지안 분류기와 중심점 기반 분류기

를 구현하여 한글 전자메일 분류성능을 정확도를 기준으로 비교하였다. 연구결과 두 분류기 모두 외국의 연구사례와 비슷하게 비교적 우수한 분류정확도를 보여 주었다. 중심점 기반 분류기는 학습집합의 크기가 클 때 나이브 베이지안 분류기보다 더 우수한 정확도를 보였으며, 학습집합의 구성에 상관없이 비교적 안정적인 정확도를 보이는 것을 알 수 있었다. 그리고, 나이브 베이지안 분류기는 학습집합의 구성에 따라 민감한 정확도 변화를 보인다는 것을 알 수 있었다.

그러나 클래스의 수가 증가할 경우 나이브 베이지안 분류기가 중심점 기반 분류기보다 더 우수한 결과를 보인다는 것을 알 수 있었다.

이것은 웹 문서분류에서 중심점 기반 분류기가 나이브 베이지안 분류기보다 더 좋은 성능을 보인다는 연구결과와 상반되는 결과이다. 이러한 결과가 나온 이유는 전자메일이 일반문서와는 다른 성격을 가졌기 때문이라고 해석될 수 있다. 그 대표적인 것으로 전자메일은 개인적인 내용을 다루면서 특별한 형식이 없다는 것과 신조어나 인터넷 용어를 많이 사용하고 맞춤법에 어긋나는 용어를 많이 사용한다는 것인데, 이러한 요인들이 중심점 기반 분류기의 오분류율을 높였다고 해석될 수 있다.

현업에서 전자메일 분류기를 구현하고자 할 경우, 본 연구보다 더 많은 분류 클래스를 가지고, 학습을 위한 데이터 수를 증가시키는 것에 대한 많은 애로사항을 가진다. 따라서 나이브 베이지안 분류기를 적용하여 전자메일 분류기를 구현하는 것이 적합하다고 생각된다.

본 연구는 온라인 쇼핑몰과 카드회사 전자메일에 대해서 실험을 하였다. 향후 여러 분야의 데이터를 이용한 계속적인 실험을 통해서, 본 연구의 결과를 검증해 보아야 될 것이다.

또한, 나이브 베이지안 분류기가 학습집합의 구성에 따라서 정확도의 변화가 심하게 나타나는 단점은 앞으로 개선되어야 할 과제라고 생각된다. 본 연구에서는 문서분류를 위해서 정보력이 뛰어난 속성들만을 추출하는 부분을 배제하고 연구를 수행하였다. 속성추출방법을 이용하여 정보력이 뛰어난 속성들만을 이용하여 나이브 베이지안 분류기를 구현한다면 학습집합의 구성에 따른 심한 정확도 변화를 개선할 수 있을 것이다. 그러나 본 연구에서는 모든 속성들을 이용할 경우 나이브 베이지안 분류기의 정확도가 가장 뛰어나다는 결론을 이용하여 연구를 수행하였으며, 중심점 기반 분류 알고리듬은 속성들 간의 독립을 가정하지 않은 알고리듬인 관계로 속성추출방법을 적용하지 못했다.

본 연구에서 실험에 허용된 데이터는 온라인 쇼핑몰과 카드회사 전자메일의 2가지 종류이며, 온라인 쇼핑몰은 3개의 클래스, 카드회사 전자메일은 최소 3개 최대 7개의 클래스를 이용하여 실험을 하였기에 현실의 전자메일 분류 전체를 설명하기에는 한계가 있다. 따라서 향후 더 많은 데이터에 대한 실험을 통해서 본 연구결과에 대한 검증이 요구된다.

또한, 향후 연구에서는 중심점 기반 분류기처럼 학습집합의 구성이나 크기에 관계없이 일정한 범위의 높은 정확도를 유지하면서 나이브 베이지안 분류기처럼 클래스 수가 증가하더라도 정확도의 하락폭이 크지 않은 전자메일 분류기 개발에 관한 연구가 진행되어야 할 것이다.

## 참고문헌


Apte. C. and Damerau. F.(1994), Automated Learning of Decision Rules for Text Categorization, *ACMTOIS*, 12(3), 233-251.

Castelli. V. and Cover. M. T.(1995), On the Exponential Value of Labeled Samples, *Pattern Recognition Letters*, 16(1), 105-111.

Cohen. W. W.(1996), Learning Rule that Classify E-Mail, *AAAI spring symposium*.

Diao. Y., Lu. H. and Wu. D.(2000), A Comparative Study of Classification Based Personal E-mail Filtering, *PAKDD*.

Dietterich. T. G.(1998), Approximate statistical tests for comparing supervised classification learning algorithms, *Neural Computation*, 10(7).

Dumais. S. S., Heckerman. D. and Horvitz. E.(1998), A Bayesian Approach to Filtering Junk e-mail, *AAAI Technical Report WS- 98-05*.

Han(Sam). E. H. and Karypis. G.(2000), Centroid-Based Document Classification : Analysis & Experimental Results, *PAKDD*.

Han, K. R., Sun, B. K., Han, S. T. and Rim, K. W.(2000), A Study on Development of Automatic Categorization System for Internet Documents, *Korea Information Processing Society*, 79(9), 2867-2875.

Hur, J. H., Choi, J. H., Lee, J. H., Kim, J. B. and Rim, K. W.(2001), An Automatic Classification System of Korean Documents Using Weight for Keywords of Document and Word Cluster, *Korea Information Processing Society*, 8(5), 447-454.

Hwang, H. S.(2001), Developing e-Mail Classifier for Front end e-CRM, *Masters Thesis, Dongguk University*.

Ko, S. J. and Lee, J. H.(2001), Bayesian Automatic Document Categorization Using Apriori-Genetic Algorithm, *Korea Information Processing Society*, 8(3), 251-260.

Lewis. D. and Ringuette. M.(1998), Comparison of two learning algorithms for text categorization, *In Tenth European Conference on Machine Learning*.

LG Economic Research Institute.(2000), E-mail Marketing Strategy, *Weekly Economy* No. 593.

Liere. R. and Tadepalli. P.(1996), The Use of Active Learning in Text Categorization, *Working notes of the AAAI Spring Symposium on Machine Learning, Stanford*.

McCallum. A. and Nigam. K.(1998), A comparison of event models for naive bayes text classification, *In AAAI-98 Workshop on Learning for Text Categorization*.

Mitchell. T. M.(1997), *Machine Learning*, The McGraw-Hill Company.

Salton. G.(1989), *Automatic Text Processing : The Transformation, Analysis, and Retrieval of Information by Computer*, Addison Wesley.

Yang. Y.(1994), Expert network: Effective and efficient learning from human decisions in text categorization and retrieval, *In SIGIR-94*.

Yoon, J. S.(2001), Improving Naive Bayesian e-Mail Classifier Accuracy by Bagging and Boosting, *Masters Thesis, Dongguk University*.

**김 국 표**
동국대학교 정보통계학 학사
동국대학교 산업공학 석사
현재: 트랙원 테크놀로지스 컨설팅 팀
관심분야: 데이터마이닝, 정보시스템, 신용 평가시스템

**권 영 식**
서울대학교 산업공학 학사
한국과학기술원 산업공학 석사
한국과학기술원 산업공학 박사
현재: 동국대학교 산업시스템공학과 교수
관심분야: 데이터마이닝, Machine learning, IT 전략