k-최근접 템플릿기반 다중 분류기 결합방법
(Multiple Classifier Fusion Method based on k-Nearest Templates)

민 준 기 † 조 성 배 ††
(Jun-Ki Min) (Sung-Bae Cho)

요 약 본 논문에서는 다중 분류기를 효과적으로 결합하기 위하여 k-최근접 템플릿방법을 제안한다. 이는 하나의 클래스를 여러개의 템플릿으로 모델링하기 위하여 분류기의 출력값을 기반으로 각 클래별 학습 샘플들을 여러개의 하위 클래스로 분해하고, 각 하위클래스별 분류기 출력값의 평균을 계산하여 지역화된 템플릿을 생성한다. 그 뒤 평가샘플과 각 템플릿간의 거리를 계산하고, k개의 최근접 템플릿들 중 가장 많은 비율을 차지하는 클래스로 평가샘플을 분류한다. 본 논문에서는 클래스 분해를 위해 C-means 클러스터링 알고리즘을 이용하였으며, k값은 주어진 데이타 셋의 클래스 내 밀집도와 클래스 간 분리도에 따라 자동으로 결정하였다. 제안하는 방법은 각 클래스별로 여러 개의 모델을 사용하며, 이들 중 가장 유사한 하나의 모델과 매칭하는 대신 k개의 모델을 참조하기 때문에 안정적이고 높은 분류성능을 획득할 수 있다. 본 논문에서는 UCI와 ELENA데이타베이스를 이용한 실험을 통해 제안하는 방법이 기존의 결합방법들에 비해 우수한 분류성능을 보임을 확인하였다.

키워드 : 분류기 결합, 지역화된 템플릿, C-Means 클러스터링

Abstract In this paper, the k-nearest templates method is proposed to combine multiple classifiers effectively. First, the method decomposes training samples of each class into several subclasses based on the outputs of classifiers to represent a class as multiple models, and estimates a localized template by averaging the outputs for each subclass. The distances between a test sample and templates are then calculated. Lastly, the test sample is assigned to the class that is most frequently represented among the k most similar templates. In this paper, C-means clustering algorithm is used as the decomposition method, and k is automatically chosen according to the intra-class compactness and inter-class separation of a given data set. Since the proposed method uses multiple models per class and refers to k models rather than matches with the most similar one, it could obtain stable and high accuracy. In this paper, experiments on UCI and ELENA database showed that the proposed method performed better than conventional fusion methods.

Key words : Classifier Fusion, Localized Template, C-Means Clustering

1. 서 론

다중 분류기의 결합은 높고 안정적인 분류성능을 얻기 위한 방법으로 패턴인식 분야에서 많이 연구되어 왔다[1]. 분류기 결합 방법은 크게 추가 학습이 필요한 것과 추가 학습이 필요 없는 것으로 나눌 수 있다. 추가 학습이 필요 없는 방법 중 가장 널리 사용되는 것으로는 투표기반(MAJ), 최대값 선택(MAX), 최소값 선택(MIN), 평균 선택(AVG) 등이 있으며[2], 추가 학습이 필요한 방법 중 대표적인 것으로는 결정템플릿(DT, Decision Templates)과 BKS(Behavior Knowledge Space)가 있다[3]. 이 외에도 여러 개의 분류기를 생성한 뒤 예측된 분류결과에 따라 분류기를 선택하는 방법도 연구되고 있다[4].

앞에서 소개한 결합방법들 중 결정템플릿은 클래스별 학습샘플에 대한 분류기의 출력 벡터들의 중심을 해당 클래스의 분류모형으로 사용하는 방법으로, 높은 분류성능을 보이며 분류기 선택방법과 혼합되어 사용되기도 하였다[4]. 그러나 이 방법은 클래스를 하나의 템플릿으로 모델링하기 때문에 데이타의 특징을 정교하게 표현하는데 어려움이 있다. 이를 해결하기 위해 클러스터링 알고리즘을 이용하여 여러 개의 국부화된 템플릿을 생성하는 다중결정템플릿(MuDTs) 방법이 제안되었지만, 이는 하나의 템플릿만을 참조함으로써 클러스터링 결과에 민감할 수 있다[5].

본 논문에서는 다중결정템플릿의 분류성능과 안정성을 높인 k-최근접 템플릿방법을 제안한다. 이 방법은 데이타 셋의 클래스 내 밀집도와 클래스 간 분리도에 따라 다중결정템플릿에서 참조할 템플릿의 수인 k값을 자동으로 결정하며, 이를 통해 오류 클러스터의 영향을

· 이 논문은 지식경제부를 통해 IITA에서 지원받았음(IITA-2008-(C1090-0801-0011))
· 이 논문은 제34회 추계학술대회에서 'k-최근접 템플릿기반 다중 분류기 결합방법'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 연세대학교 컴퓨터과학과
loomlike@sclab.yonsei.ac.kr
†† 종신회원 : 연세대학교 컴퓨터과학과 교수
sbcho@cs.yonsei.ac.kr

논문접수 : 2008년 1월 10일
심사완료 : 2008년 3월 24일

정보과학회논문지: 컴퓨팅의 실제 및 레터 제14권 제4호(2008.6)

줄인다. 본 논문에서는 다양한 데이타 셋을 이용한 실험을 통해 제안하는 방법의 성능을 검증하였다.

2. 관련연구

2.1 C-Means 알고리즘

C-Means(또는 k-Means)알고리즘은 샘플과 클러스터중심 간의 거리 I를 최소화하는 C개의 클러스터를 탐색하는 반복적 알고리즘이다[6]. 데이타 셋의 샘플 수를 n이라 하고, c번째 클러스터의 중심을 z_c라 하였을 때 I는 일반적으로 다음과 같이 계산한다.

$$I = \sum_{i=1}^{n}\sum_{c=1}^{C} u_{c,i}\|x_i - z_c\|^2 \quad (1)$$

식 (1)에서 $u_{c,i}$는 분할행렬(Partition matrix)의 (c,i)번째 원소를 나타내는 것으로, 샘플 x_i가 클러스터가 c에 속한 경우 1이고 그 외에는 0의 값을 갖는다. C-Means알고리즘은 데이타공간상에서 임의로 C개의 점을 선택한 뒤 이를 중심으로 클러스터를 분할한다. 각 클러스터의 중심은 소속 샘플들의 중심점으로 갱신되며, 중심의 이동이 거의 없어질 때까지 이 과정을 반복한다. 이 방법은 지역 해에 빠질 수 있다는 단점이 있지만 알고리즘이 간단하면서 좋은 성능을 보이고, 또한 클러스터 수의 선택이 용이해 널리 사용된다.

2.2 결정템플릿

결정템플릿은 Kuncheva[3]가 제안한 분류기 결합방법으로, 학습데이타에 대한 분류기의 출력 값을 행렬 형식의 결정프로파일로 구성한다. M클래스 문제에 대해 L개의 분류기를 사용할 때, i번째 학습샘플 x_i $(i=1,\cdots,n)$의 결정프로파일 $DP(x_i)$는 다음과 같다.

$$DP(x_i) = \begin{bmatrix} d_{1,1}(x_i) & \cdots & d_{1,M}(x_i) \\ \vdots & d_{y,z}(x_i) & \vdots \\ d_{L,1}(x_i) & \cdots & d_{L,M}(x_i) \end{bmatrix} \quad (2)$$

식 (2)의 $d_{y,z}(x_i)$는 y번째 분류기의 클래스 z에 대한 출력 값을 의미한다. 결정프로파일이 생성되면, 식 (3)과 식 (4)를 이용하여 클래스 m에 대한 템플릿 DT_m을 계산한다.

$$DT_m = \begin{bmatrix} dt_m(1,1) & \cdots & dt_m(1,M) \\ \vdots & dt_m(y,z) & \vdots \\ dt_m(L,1) & \cdots & dt_m(L,M) \end{bmatrix} \quad (3)$$

$$dt_m(y,z) = \sum_{i=1}^{n} u_{m,i} d_{y,z}(x_i) \Big/ \sum_{i=1}^{n} u_{m,i} \quad (4)$$

분류 시에는 평가 샘플의 결정프로파일과 각 클래스의 템플릿을 유클리드거리 등과 같은 유사도 계산방법을 사용하여 비교한 뒤, 가장 유사한 템플릿의 레이블로 샘플의 클래스를 결정한다.

3. k-최근접 템플릿기반 분류기 결합

결정템플릿방법은 클래스를 하나의 템플릿으로 축약하여 모델링하기 때문에 다양한 특징 정보가 손실된다. 본 장에서는 클래스를 여러 개의 하위클래스로 분해한 뒤 k개의 템플릿을 참조하여 분류성능과 안정성을 높인 k-최근접 템플릿기반 분류기 결합방법에 대해 설명한다. 그림 1은 제안하는 방법의 전체 흐름도를 나타낸다.

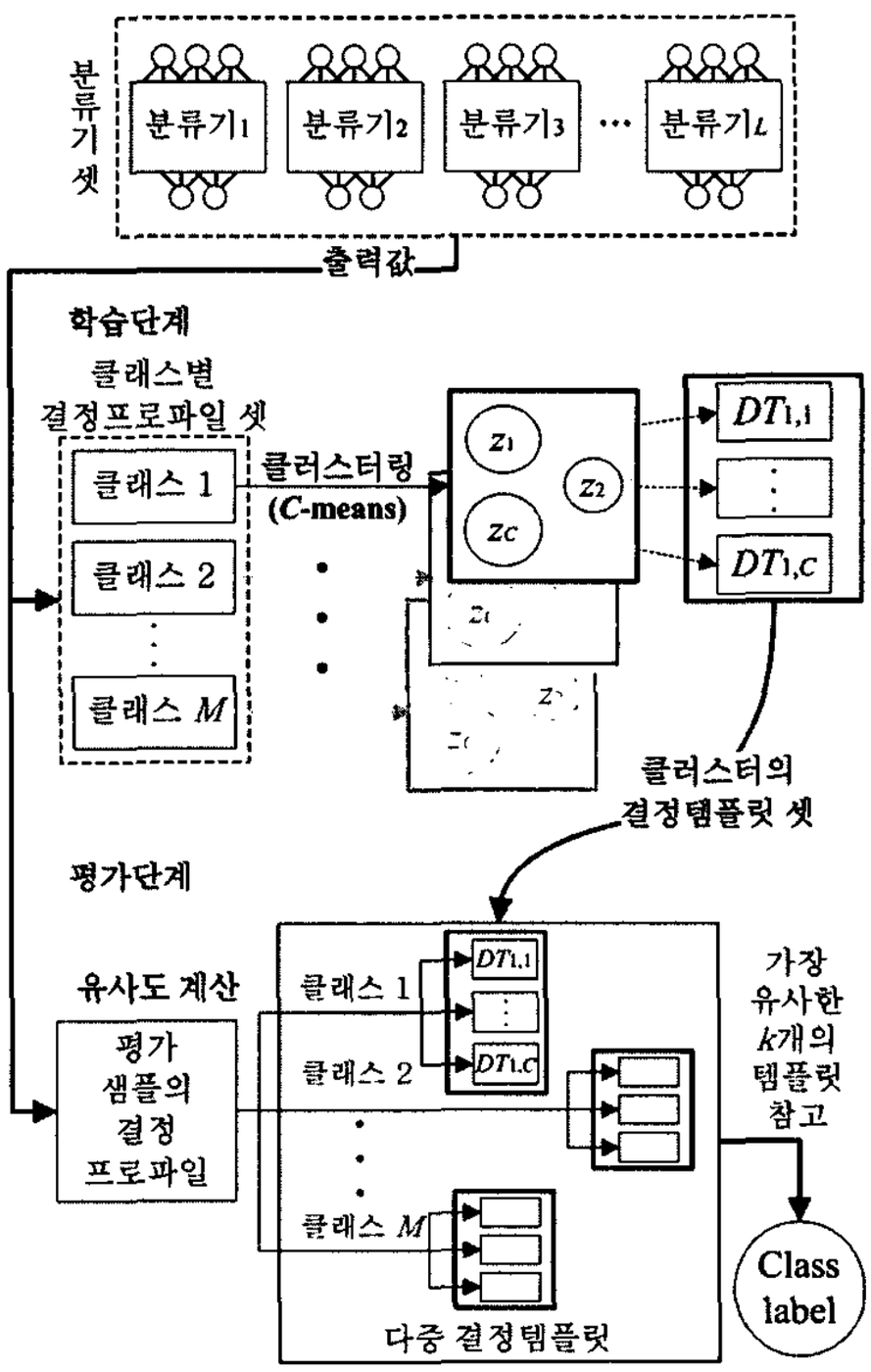

그림 1 다중결정템플릿의 생성 및 k-최근접 템플릿기반 분류

3.1 다중결정템플릿의 생성

먼저 개별 분류기를 학습하고, 동일한 학습 데이타로부터 얻은 분류기들의 출력 값을 2.2절의 식 (2)의 결정프로파일로 구성한다. 각 클래스의 결정프로파일을 C-Means 알고리즘을 이용하여 클러스터링하고, 다음 식을 이용하여 c번째 클러스터의 지역화된 템플릿 $DT_{m,c}$를 계산한다[5].

$$DT_{m,c} = \begin{bmatrix} dt_{m,c}(1,1) & \cdots & dt_{m,c}(1,M) \\ \vdots & dt_{m,c}(y,z) & \vdots \\ dt_{m,c}(L,1) & \cdots & dt_{m,c}(L,M) \end{bmatrix} \quad (5)$$

$$dt_{m,c}(y,z) = \sum_{i=1}^{n} u_{m,c,i} d_{y,z}(x_i) \Big/ \sum_{i=1}^{n} u_{m,c,i} \quad (6)$$

식 (6)에서 $u_{m,c,i}$는 샘플 x_i가 클래스 m의 c번째 클러스터에 소속되어있는 경우 1이고 그 외에는 0의 값을 갖는다. 본 논문에서는 클러스터의 수를 데이타 셋에 상관없이 C=20으로 고정시켰다.

3.2 k-최근접 템플릿 기반 분류

평가샘플의 결정프로파일과 템플릿들 간의 유사도를 계산한 뒤, 가장 유사한 k개의 템플릿들 중 가장 많은 비율을 차지하는 레이블로 샘플을 분류한다. 본 논문에서는 유클리드거리식을 이용하여 유사도를 계산하였다. 이 방법은 k-최근접 이웃(k-Nearest Neighbor)분류방법과 마찬가지로 k값에 영향을 많이 받는다. 최적의 k값은 데이타 셋에 의존적이기 때문에 제안하는 방법에서는 클래스 내 밀집도(Intra-class compactness)와 클래스 간 분리도(Inter-class separation)를 분석하여 k값을 결정한다. 데이타 셋의 밀도 분석은 주로 클러스터링 알고리즘의 정당성 지표(Validity index)에 사용되는 방법으로, 본 논문에서는 식 (7)과 식 (8)을 이용하여 클래스 내 밀집도 IC와 클래스 간 분리도 IS를 계산하였다[7].

$$IC = E_1 / E_M,\ E_M = \sum_{i=1}^{n}\sum_{m=1}^{M} u_{m,i}\|x_i - z_m\| \tag{7}$$

$$IS = \max_{i,j=1,\ldots,c}\|z_i - z_j\| \tag{8}$$

k값은 식 (9)와 같이 간단한 규칙에 의해 결정되며, 임계값인 t_{IC}와 t_{IS}는 실험을 통해 각각 1.5와 2로 결정하였다.

$$k = \begin{cases} 1 & \text{if } IC \le t_{IC} \text{ and } IS \le t_{IS} \\ C/2 & \text{if } IC > t_{IC} \text{ and } IS > t_{IS} \end{cases} \tag{9}$$

4. 실험 및 결과

4.1 실험 환경

본 논문에서는 패턴인식 분야에서 널리 사용되는 UCI와 ELENA 데이타베이스 중 10개의 데이타 셋을 대상으로 실험을 수행하였다(표 1).

본 논문에서는 결합을 위한 개별 분류기로 신경망을 사용하였으며, 학습율과 모멘텀은 각각 0.15와 0.9로 설정하였다. 신경망 내부 노드들 간의 초기 연결강도는 -0.5~0.5 사이의 임의 값으로 초기화하였으며, 각 신경망은 Bagging을 이용하여 학습하였다. 신경망의 은닉노드(Hidden node) 수와 세대 수는 [1]의 연구와 동일한 기준으로 다음과 같이 결정하였다. 우선 은닉노드는 최소 5개가 되도록 하였으며, 하나의 클래스 당 혹은 10개의 특징 당 최소 하나의 은닉노드를 갖도록 하였다. 세대 수는 데이타 셋의 샘플수가 250개 이하인 경우 60~80세대, 샘플수가 250~500개인 경우 40세대, 샘플수가 500개 이상인 경우 20~40세대로 설정하였다(표 2).

표 1 실험에 사용된 데이타 셋

데이타 셋	샘플 수	특징 수	클래스 수	출처
Breast-cancer (Br)	683	9	2	UCI
Ionosphere (Io)	351	34	2	UCI
Iris (Ir)	150	4	3	UCI
Satellite (Sa)	6435	36	6	UCI
Segmentation (Se)	2310	19	7	UCI
Sonar (So)	208	60	2	UCI
Phoneme (Ph)	5404	5	2	ELENA
Texture (Te)	5500	40	11	ELENA
Clouds (Cl)	5000	2	2	ELENA
Concentric (Co)	2500	2	2	ELENA

표 2 신경망 파라메터

데이타 셋	은닉노드 수	세대 수
Breast-cancer (Br)	5	20
Ionosphere (Io)	10	40
Iris (Ir)	5	80
Satellite (Sa)	15	30
Segmentation (Se)	15	20
Sonar (So)	10	60
Phoneme (Ph)	5	30
Texture (Te)	20	40
Clouds (Cl)	5	20
Concentric (Co)	5	20

본 논문에서 사용할 신경망의 수는 기존의 분류기 결합방법인 MAJ, MIN, MAX, AVG를 이용한 실험을 통해 결정하였다. 그림 2와 같이 사용하는 신경망의 수가 10개 이상이 되면 결합을 통해 얻을 수 있는 성능의 향상 정도가 줄어드는 것을 확인하였으며, 따라서 본 논문의 이후 실험에서는 [1]의 연구와 마찬가지로 신경망의 수를 25개로 고정하였다.

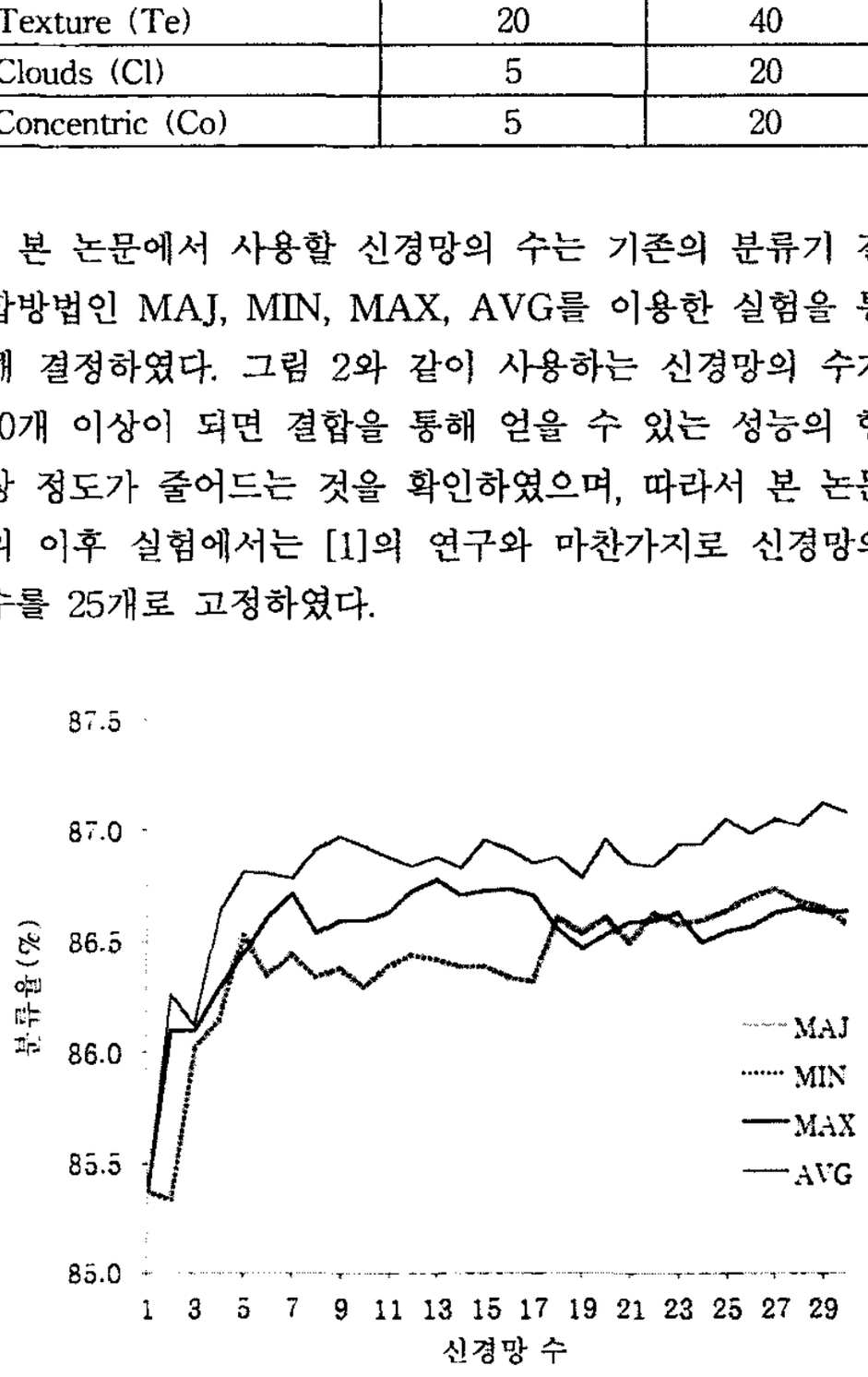

그림 2 결합하는 신경망 수에 따른 분류성능(모든 데이타 셋에 대한 실험결과의 평균)

4.2 분류성능 평가

결합 방법별 성능을 비교하기 위하여 각 데이타에 대해 10-fold cross validation실험을 수행하였다. 표 3과 표 4는 각각 실험결과의 오분류율과 표준편차를 나타낸다.

결합방법 중 오라클(ORA)은 분류기들 중 하나라도 맞게 분류하면 전체가 맞았다고 보는 방법으로, 데이타 셋의 최대 가능 분류율을 의미한다. 제안하는 방법인 k-최근접 템플릿방법의 경우 3.2절의 식 (9)에 의해 Ionosphere, Sonar, Phoneme, Clouds, Concentric의 5가지 데이타 셋의 경우 k=1, 나머지의 경우는 $k=C/2$=10으로 k값이 선택되었다. 표 3에서 굵게 표시된 숫자는 각 데이타 셋의 최소 오분류율을 나타내는 것으로(오라클 제외), 실험결과 제안하는 방법이 가장 많은 수인 6개의 데이타 셋에서 최고 분류율을 보였으며, 그 외의 데이타 셋에서도 높은 성능을 나타냈다. MuDTs(다중결정템플릿)은 두 번째로 좋은 성능을 보였으며, 기존 결합방법들 중에서는 평균 선택(AVG)방법이 좋은 성능을 나타냈다.

그림 3은 전체 데이타 셋에 대한 평균 오분류율과 표준편차를 보여준다. 그림과 같이 제안하는 방법이 다른 방법에 비해 가장 높은 성능을 보였으며, DT(결정템플릿)과 AVG(평균 선택 방법)은 비슷한 성능을 보였다. MuDTs의 경우 분류성능은 DT와 비슷하였으나, 클러스터링 결과에 영향을 받기 때문에 표준편차가 높게 나타났다. 표 5는 제안하는 방법과 AVG, DT, MuDTs간의 대응 t검정결과를 보여준다.

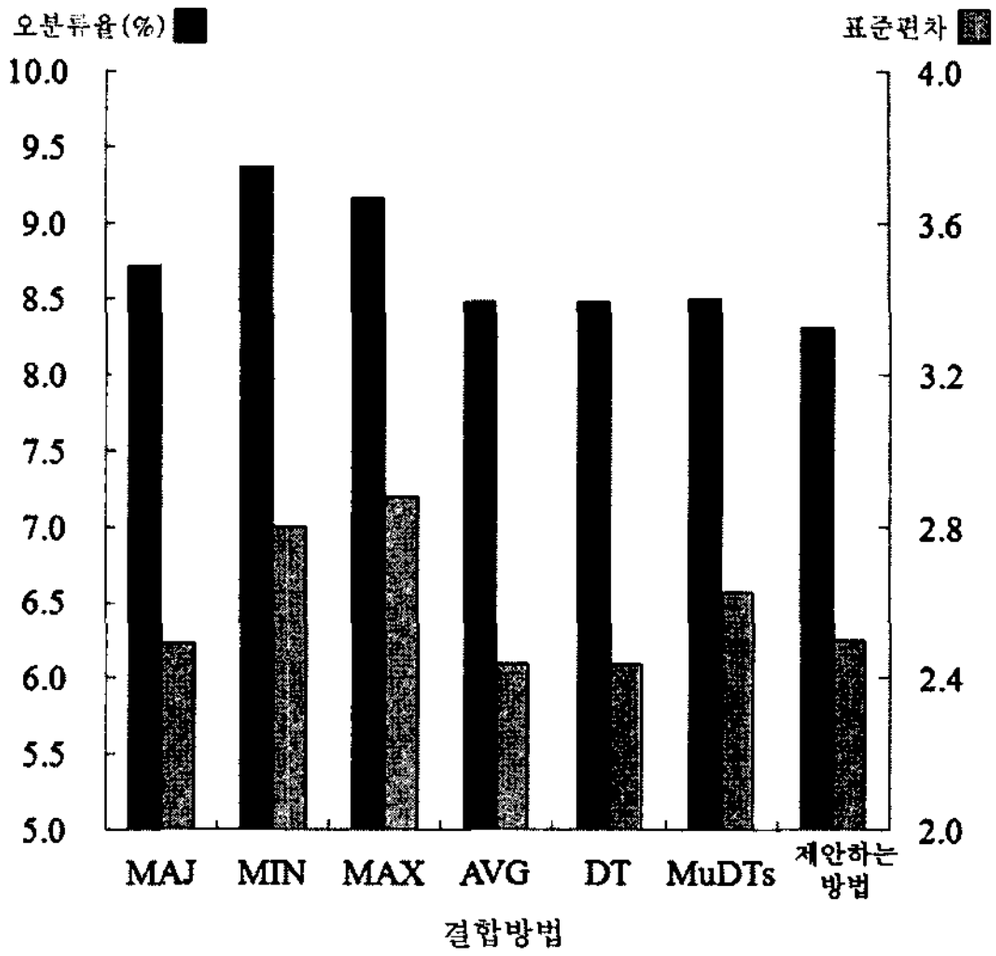

그림 3 결합방법별 모든 데이타 셋에 대한 평균 오분류율과 표준편차

표 3 모든 데이타 셋에 대한 결합방법별 오분류율

	MAJ	MIN	MAX	AVG	DT	Mu DTs	제안하는 방법	ORA
Br	3.09	2.94	2.94	2.94	**2.79**	4.56	**2.79**	1.32
Io	10.00	9.15	8.86	10.00	10.00	**7.72**	**7.72**	1.72
Ir	3.33	3.33	4.00	**2.67**	**2.67**	4.67	3.33	1.33
Sa	10.56	11.20	11.25	10.42	10.64	11.38	**10.37**	2.44
Se	5.89	6.36	5.58	5.54	5.54	**3.77**	5.41	1.17
So	14.50	19.00	17.50	**14.00**	14.50	16.00	16.00	2.00
Ph	19.82	19.74	19.74	19.67	19.61	**19.35**	**19.35**	6.89
Te	**0.31**	0.44	0.42	0.33	0.33	0.36	0.33	0.04
Cl	20.54	20.70	20.70	20.64	20.42	**18.10**	**18.10**	15.30
Co	2.28	2.44	2.44	2.16	2.04	**1.24**	**1.24**	0.00

표 4 모든 데이타 셋에 대한 결합방법별 표준편차

	MAJ	MIN	MAX	AVG	DT	Mu DTs	제안하는 방법	ORA
Br	1.62	1.55	1.70	1.83	1.76	2.10	1.83	1.76
Io	5.43	5.18	5.30	5.43	5.43	4.48	4.48	2.76
Ir	4.71	4.71	4.66	4.66	4.66	5.49	4.71	2.81
Sa	1.24	0.91	0.99	0.98	1.14	1.34	0.99	0.51
Se	1.85	2.16	1.91	1.71	1.67	1.42	1.48	0.54
So	6.43	8.43	9.20	6.58	6.43	7.75	7.75	3.50
Ph	1.53	1.62	1.62	1.35	1.50	1.76	1.76	1.20
Te	0.15	0.23	0.29	0.15	0.15	0.21	0.15	0.08
Cl	2.47	2.51	2.50	2.45	2.52	1.66	1.66	3.66
Co	1.16	1.15	1.15	0.76	0.81	0.61	0.65	0.00

표 5 제안하는 방법과의 t검정을 통한 성능 통계적 유의수준 분석결과

결합방법	p값
AVG	$p<0.02$
DT	$p<0.002$
MuDTs	$p<0.007$

4.3 성능 안정성 평가

분류기 결합방법의 성능 안정성을 평가하기 위하여 본 논문에서는 결합방법별로 모든 데이타셋에 대한 10-fold cross validation실험 분류결과의 CV(Coefficient of Variance)값을 측정하였다. CV값은 상호분산정도를 측정하기위해 많이 사용되는 통계적 방법으로, 데이타 값의 집중경향을 나타낸다. σ와 μ를 각각 분류율의 표준편차와 평균이라고 하였을 때, CV값은 다음과 같이 계산된다.

$$CV = \frac{\sigma}{\mu} \times 100\% \tag{10}$$

이때 CV값은 작을수록 분류성능이 안정적임을 나타낸다. 분석결과 그림 4와 같이 제안하는 방법이 기존의 결합방법에 비해 가장 안정적인 성능을 보임을 확인하였다.

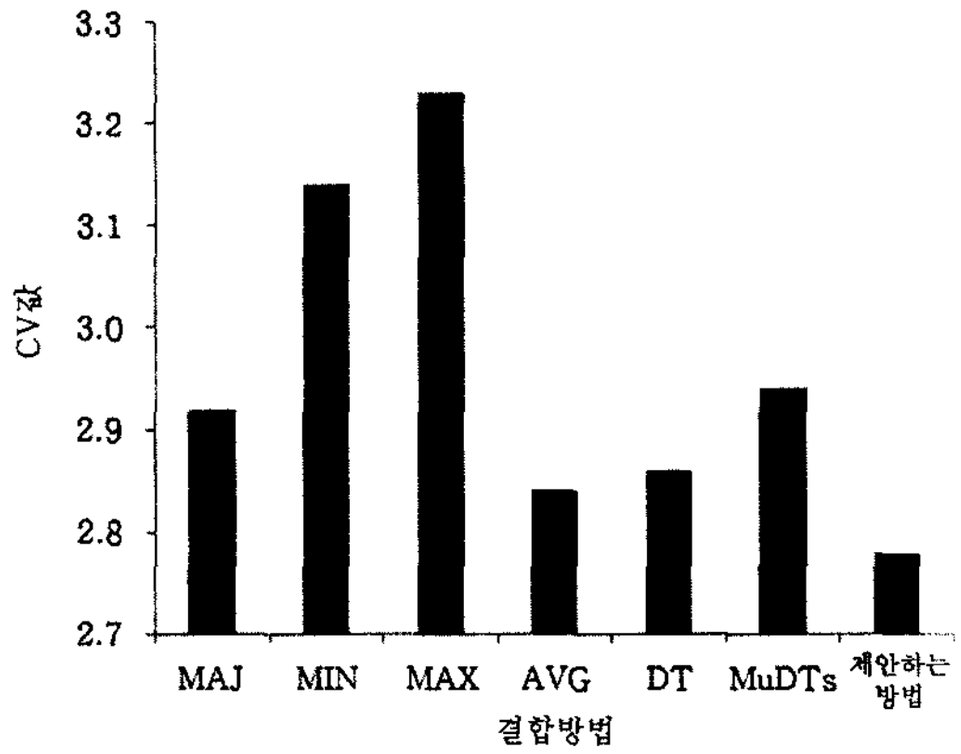

그림 4 분류기 결합방법별 성능 안정성(CV값이 작을수록 안정적임을 나타냄)

5. 결 론

본 논문에서는 다중 분류기를 효과적으로 결합하기 위하여 다중결정템플릿방법에 k-최근접 이웃 분류기법을 적용한 k-최근접 템플릿을 제안하였다. 이는 각 클래스를 여러 개의 템플릿으로 모델링하며, 이들 중 가장 유사한 하나의 템플릿과 매칭하는 대신 k개의 모델을 참조하기 때문에 기존의 분류기 결합방법에 비해 안정적이고 높은 분류성능을 획득할 수 있다. 이때 k값은 데이타 셋의 클래스 내 밀집도와 클래스 간 분리도에 따라 적합한 k값을 자동으로 선택한다. 기존의 분류기 결합방법인 투표기반, 최대값 선택, 최소값 선택, 평균 선택, 결정템플릿방법 등을 이용하여 UCI와 ELENA의 10가지 데이타 셋에 대한 비교실험을 수행한 결과 제안하는 방법이 안정적이면서 높은 성능을 보임을 확인하였다.

추후연구로 보다 다양한 데이타 셋에 대한 성능평가를 통해 방법의 일반성을 검증하고, 데이타 셋의 특징과 k값과의 상관관계를 정량적으로 분석하여 k값을 보다 정교하게 선택해주는 규칙을 생성할 계획이다.

참 고 문 헌

[1] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," J. Artificial Intelligence Research, Vol.11, pp. 169-198, 1999.

[2] L.I. Kuncheva, "A theoretical study on six classifier fusion strategies," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.24, No.2, pp. 281-286, 2002.

[3] L.I. Kuncheva, J.C. Bezdek, and R.P.W. Duin, "Decision templates for multiple classifier fusion: An experimental comparison," Pattern Recognition, Vol.34, No.2, pp. 299-314, 2001.

[4] L.I. Kuncheva, "Switching between selection and fusion in combining classifiers: An experiment," IEEE Trans. Systems, Man, and Cybernetics, Part B-Cybernetics, Vol.32, No.2, pp. 146-156, 2002.

[5] J.-K. Min, J.-H. Hong, and S.-B. Cho, "Fingerprint classification using multiple decision template with SVM," J. Korea Information Science Society, Vol.32, No.11, pp. 1136-1146, 2005.

[6] A.K. Jain and R.C. Dubes, Algorithms for Clustering Data, Prentice Hall, 1988.

[7] U. Maulik and S. Bandyopadhyay, "Performance evaluation of some clustering algorithms and validity indices," IEEE Trans. Pattern Analysis and Machine Intelligence, Vol.24, No.12, pp. 1650-1654, 2002.