재귀분할을 이용한 새로운 점진적 인스턴스 기반 학습기법

한 진 철[†] · 김 상 귀[††] · 윤 충 화[†††]

요 약

인스턴스 기반 학습의 대표적인 알고리즘인 k-NN(K-Nearest Neighbors)은 단순히 전체 학습패턴을 메모리에 저장한 다음, 분류할 때 학습패턴들과의 거리를 계산하여 가장 가까운 학습패턴의 클래스로 테스트 패턴을 분류한다. K-NN 기법은 만족할 만한 분류성능을 보여주지만, 학습패턴의 개수가 늘어나면 메모리와 분류 시간이 증가하는 문제점을 가지고 있다. 그러므로, 메모리의 효율적 사용과 분류 시간을 단축시키기 위한 다양한 연구들이 발표되었으며, 그 대표적인 예로 NGE(Nested Generalized Exemplar) 이론을 들 수 있다. 본 논문에서는 학습패턴의 집합으로부터 대표패턴을 생성하는 RPA(Recursive Partition Averaging)기법과 점진적으로 대표패턴을 추출하는 IRPA(Incremental RPA)기법을 제안하였다. RPA기법은 전체 학습패턴의 공간을 재귀적으로 분할하면서 대표패턴을 생성하며, IRPA 기법은 RPA 기법의 특성상 패턴의 특징 개수가 많은 경우, 과도한 분할로 인하여 생성되는 많은 개수의 대표패턴을 줄이기 위하여 점진적으로 대표패턴을 추출하는 알고리즘이다. 본 논문에서 제안한 기법은 기존의 k-NN 기법과 비교하여 현저하게 줄어든 대표패턴을 이용하여 유사한 분류 성능을 보여주며, NGE 이론을 구현한 EACH 시스템과 비교하여 탁월한 분류 성능을 보여준다.

키워드 : 메모리기반 추론, 인스턴스 기반 학습, 점진적 학습 알고리즘

A New Incremental Instance-Based Learning Using Recursive Partitioning

Jin-Chul Han[†] · Sang-Kwi Kim[††] · Chung-Hwa Yoon[†††]

ABSTRACT

K-NN (k-Nearest Neighbors), which is a well-known instance-based learning algorithm, simply stores entire training patterns in memory, and uses a distance function to classify a test pattern. K-NN is proven to show satisfactory performance, but it is notorious formemory usage and lengthy computation. Various studies have been found in the literature in order to minimize memory usage and computation time, and NGE (Nested Generalized Exemplar) theory is one of them. In this paper, we propose RPA (Recursive Partition Averaging) and IRPA (Incremental RPA) which is an incremental version of RPA. RPA partitions the entire pattern space recursively, and generates representatives from each partition. Also, due to the fact that RPA is prone to produce excessive number of partitions as the number of features in a pattern increases, we present IRPA which reduces the number of representative patterns by processing the training set in an incremental manner. Our proposed methods have been successfully shown to exhibit comparable performance to k-NN with a lot less number of patterns and better result than EACH system which implements the NGE theory.

Key Words : Memory-Based Reasoning, Instance-Based Learning, Incremental Learning Algorithm

1. 서 론

메모리 기반 추론(Memory-Based Reasoning)은 단순히 학습패턴 전체를 메모리에 저장한 다음, 테스트 패턴과의 거리를 계산하여 분류하므로, 거리기반 학습(Distance Based Learning) 기법이라고도 한다[1, 2]. 메모리 기반 추론의 대표적인 학습기법인 k-NN(k-Nearest Neighbors)은 학습패턴들과 테스트 패턴 사이의 거리를 계산하여 가장 가까운 k개의 학습패턴을 선택하고, 가장 많은 학습패턴이 소속된 클래스로 테스트 패턴을 분류한다[2, 3]. K-NN 기법은 성능면에서 만족할 만한 결과를 보여주지만, 전체 학습패턴을 모두 메모리에 저장하기 때문에 다른 기계학습 방법에 비하여 많은 메모리를 필요로 하며, 데이터셋마다 최적의 k값을 사전에 계산하여야 한다는 문제점이 있다. 그러므로, 학습패턴 개수가 증가할 수록 분류에 필요한 시간도 많이 소요된다는 단점을 갖는다[4, 5]. 또한 k-NN 기법은 학습 시에는 단순히 학습패턴 전체를 메모리에 저장한 다음, 테스트 패턴을 분류할 때 모든 계산이 수행되므로, "Lazy Learning Algorithm" 범주에 속하는 반면에, NGE이론이나 본 논문에

† 준 회 원 : 명지대학교 산업기술연구소 전임연구원
†† 준 회 원 : 명지대학교 컴퓨터공학과 겸임교수
††† 정 회 원 : 명지대학교 컴퓨터공학과 교수
논문접수 : 2005년 11월 15일, 심사완료 : 2006년 2월 27일

서 제안한 기법은 학습시간에 분류에 사용할 초월평면이나 대표패턴을 생성하는 작업을 수행하므로 "Eager Learning Algorithm" 범주에 속한다. 이러한 메모리 기반 학습의 문제점을 해결하기 위한 연구가 활발히 진행되고 있으며, 대표적인 연구로 NGE(Nested Generalized Exemplar)[6, 7] 이론을 들 수 있다.

본 논문에서는 IBL(Instance-Based Learning) 기법을 기반으로 한 새로운 학습 방법인 RPA(Recursive Partition Averaging) 기법과, 점진적으로 대표패턴을 추출하는 IRPA (Incremental RPA) 기법을 제안하였으며, UCI Machine Learning Repository에서 벤치마크 데이터를 발췌하여 제안한 기법과 k-NN 기법, EACH 시스템의 분류 성능과 메모리 사용 효율을 실험적으로 검증하였다.

2. 관련 연구

2.1 k-NN 기법

K-NN 기법은 메모리 기반 추론을 사용한 최초의 분류기로, 알고리즘은 다음의 <표 1>과 같다.

<표 1> k-NN 기법

① 전체 학습패턴을 메모리에 저장한다.
② 테스트 패턴과 학습패턴들과의 거리를 수식 (1)을 이용하여 계산한다.
③ 위에서 계산한 거리를 기준으로 테스트 패턴과 근접한 k개의 학습패턴을 선정한다.
④ 이 k개 중에서 가장 많은 개수의 학습패턴을 포함하는 클래스로 테스트 패턴을 분류한다.

$$D = \sqrt{\sum_{i=1}^{n} (E_i - Q_i)^2} \quad (1)$$

E_i와 Q_i는 학습패턴과 테스트 패턴의 i번째 특징 값이며, n은 패턴의 특징 개수이다. 이때, k값은 분류기의 성능을 최적화하기 위하여 Leave-one-out Cross-validation기법을 사용하여 사전에 결정한다[2, 3]. 여기에서 k값은 학습패턴의 개수와 데이터 셋에 의존되는 특성을 갖는다.

k-NN 기법의 단계 ④에서, 테스트 패턴과 가까운 학습패턴에 대해 큰 가중치-거리의 역(1/D)-를 부여하는 방법을 WeightVote k-NN 기법이라고 하며, 클래스별로 가중치의 합을 구한 다음, 합이 가장 큰 클래스로 테스트 패턴을 분류한다[3].

2.2 EACH 시스템

NGE(Nested Generalized Exemplar) 이론에 기반한 학습기법인 EACH시스템은 학습패턴을 그대로 저장하는 것이 아니라, 인접한 학습패턴들을 포함하는 초월평면(Hyperrectangle)의 형태로 저장하며, 그 결과 k-NN 기법보다 적은 메모리를 사용한다[6, 7, 9]. 다음의 <표 2>는 EACH 시스템의 알고리즘을 보여준다.

<표 2> EACH 시스템

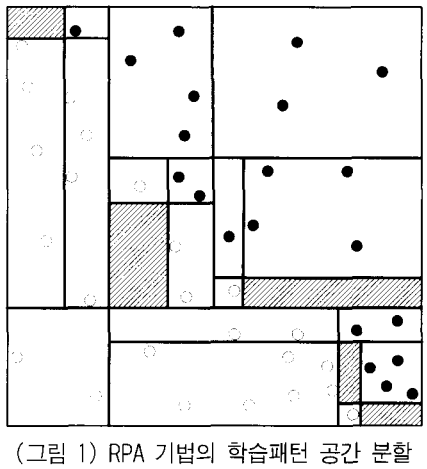
① 무작위로 몇 개의 학습패턴을 시드(seed)로 선택하여 예제(Exemplar)로 저장한다.
② 학습패턴을 선택하고, 가장 가까운 예제를 검색한다.
③ 학습패턴의 클래스와 가장 가까운 예제의 클래스가 동일하면, 학습패턴을 이용하여 그 예제를 확장하고 예제의 가중치를 수정한 다음, 단계 ⑥을 수행한다.
④ 클래스가 다를 경우, 가중치를 수정하고 두 번째로 가까운 예제를 선택한다.
⑤ 학습패턴의 클래스와 두 번째로 가까운 예제의 클래스가 동일하면, 예제를 확장하고 가중치를 수정하며, 다를 경우, 학습패턴을 별도의 새로운 예제로 저장한다.
⑥ 학습패턴 집합이 공집합이 될 때까지 단계 ②-⑤를 반복한다.

EACH 시스템의 학습이 종료되면, 학습패턴들은 예제의 집합으로 표현되는데, 예제는 점 또는 초월평면의 형태를 취하게 되며, 테스트 패턴은 가장 가까운 예제의 클래스로 분류한다. 예제가 점(point)일 경우에는 점과의 거리를 계산하며, 초월평면일 경우에는 가까운 면과의 거리를 계산한다.

3. RPA(Recursive Partition Averaging) 기법

본 논문에서 제안하는 RPA기법은 전체 학습패턴 공간을 (그림 1)과 같이 재귀적으로 분할하면서 대표패턴을 생성하며, 대표패턴은 인스턴스 평균(Instance Averaging)법을 이용하여 계산한다[10].

(그림 1)은 패턴공간이 RPA 기법에 의해 재귀적으로 분할된 예제이며, 이 예제에서는 총 17개의 대표패턴이 생성된다. 이때, 빗금친 분할영역은 학습패턴이 존재하지 않으므로 대표패턴을 생성하지 않는다. 또한, RPA 기법은 특징간의 영향력을 평준화하기 위하여 학습 개시 이전에 모든 특



(그림 1) RPA 기법의 학습패턴 공간 분할

징 값을 정규화하며, 테스트 패턴에 대한 분류 정확도를 높이기 위하여 특징 가중치를 이용한다.

3.1 특징의 정규화

메모리 기반 추론은 테스트 패턴과 메모리에 저장된 학습패턴들 사이의 거리를 계산하여, 가장 가까운 학습패턴의 클래스로 테스트 패턴을 분류한다. 따라서, 패턴을 구성하는 특징 값의 범위가 확연히 다를 경우에 문제가 발생한다. 예를 들어 (0.9, 400, 0.0004), (0.8, 410, 0.02)와 같은 특징 값을 가지는 패턴에서, 두 번째 특징은 다른 두 개의 특징에 비하여 상대적으로 큰 값을 가지므로 두 번째 특징이 조금만 차이가 나더라도 나머지 특징들에 비해 출력 클래스를 결정하는데 많은 영향을 준다. 이러한 문제점을 해결하기 위하여 다음의 수식 (2)를 이용하여 모든 특징 값을 0과 1사이의 값으로 정규화한다.

$$f_{i_{new}} = \frac{f_i - f_{i_{\min}}}{f_{i_{\max}} - f_{i_{\min}}} \tag{2}$$

f_i는 패턴의 i번째 특징 값이며, $f_{i_{\max}}$와 $f_{i_{\min}}$은 특징 i의 최대 값과 최소 값이다.

3.2 패턴공간의 분할과 대표패턴 생성

RPA 기법에서는 분할이 필요한 경우, 모든 특징에 대한 분할점을 결정해야 하며, 특징의 분할점을 선택하기 위하여 (그림 2)와 같이 특징 값을 오름차순으로 정렬하고 특징 값이 변화하는 위치를 경계 값으로 선정한다. 예를 들어, (그림 2)에서 70과 72 사이의 경계값은 두 특징 값의 평균인 71이 된다.

(그림 2) 경계 값 선정

구한 경계값들 중, 결정트리 알고리즘의 결정 노드(Decision Node)에서 특징의 비교기준을 선정할 때 사용하는 IG (Information Gain) 값을 이용하여 가장 변별력이 좋은 경계값을 분할점으로 선택한다[11]. IG값은 수식 (3), (4)을 이용하여 계산한다.

$$I = -\sum_{i=1}^{C} p_i \log_2 p_i \tag{3}$$

p_i는 학습패턴 집합에서 클래스 i에 소속되는 패턴의 비율이며, C는 클래스의 개수를 의미한다.

$$IG(f) = I - \sum_{i=1}^{2} P_i I_i \tag{4}$$

I는 분할 이전의 정보량이며, P_i는 분할 이전의 학습 패턴중, 분할된 각 영역에 포함된 학습패턴의 비율이다. I_i는 특정 경계값 f를 기준으로 분할했을 때 분할된 각 공간의 정보량을 의미하며, 수식 (3)을 이용하여 계산한다. 이때 I값이 크다는 사실은 올바르게 분류하기 위하여 많은 양의 정보가 필요하다는 것을 의미하며, IG값은 분할 이전의 정보량과 경계값을 기준으로 분할했을 경우 정보량의 차이를 의미한다. 즉, IG값은 분할 이후의 정보량이 작아질 경우에 큰 값을 가지게 되며, 결국 IG값이 큰 경계값을 분할점으로 선택할 때 효율적인 분할이 가능하다.

다음 <표 3>은 RPA 기법의 알고리즘을 보여준다.

〈표 3〉 RPA 기법

초기화 단계

① 전체 패턴 집합을 정규화한다.
② 패턴 집합을 학습패턴과 테스트 패턴 집합으로 분리한다.
③ 전체 학습패턴 집합을 포함하는 영역을 초기 분할영역으로 정의한 후, 다음의 학습 알고리즘을 적용한다.

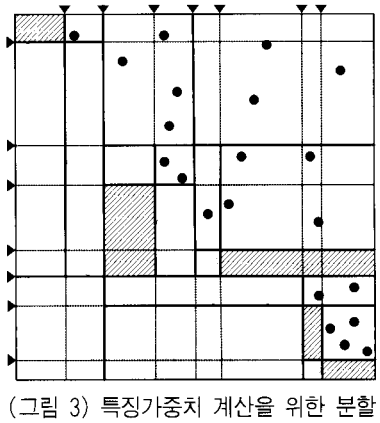
학습 알고리즘

① 현재 분할영역에 포함된 모든 학습패턴의 클래스를 검사한다.
② 만약 모든 학습패턴의 클래스가 동일하면, 인스턴스 평균법으로 대표패턴을 추출하고 종료한다.
③ 만약 클래스가 다른 학습패턴이 존재하면, 현재 분할영역의 특징별로 새로운 경계값을 구하고, 이 중에서 가장 효율적인 경계값을 분할점으로 선정한다.
④ 단계 ③에서 선정된 분할점을 이용하여 새로운 영역들로 분할한다.
⑤ 단계 ④의 분할영역중, 한 개 이상의 학습패턴을 포함하는 모든 분할영역에 대하여 위의 학습 알고리즘을 재귀 호출한다.

단계 ②의 인스턴스 평균법은 여러 개의 학습패턴의 특징 값들을 평균하여 하나의 대표패턴으로 대체하는 방법을 의미한다.

3.3 특징 가중치의 계산

본 논문에서는 테스트 패턴의 정확한 분류를 위하여 패턴간의 거리를 계산할 때, 특징 가중치를 이용하며, 특징 가중치는 학습이 완료되면, 분할된 패턴공간의 학습패턴 분포를 이용하여 계산한다.



(그림 3) 특징가중치 계산을 위한 분할

우선, 특징 가중치를 구하기 위하여 각 특징의 분할영역의 개수를 검사한다. (그림 3)은 RPA 기법으로 학습했을 경우, 분할된 패턴 공간의 예제이며, 이때 실제 분할된 분할영역의 경계선에서 가상의 분할선을 연장하여 각 특징에 대한 분할 개수를 결정한다.

(그림 3)에서 굵은 실선으로 표시된 부분은 RPA 기법에 의해 실제로 분할된 영역을 나타내며, 가는 점선으로 표시된 부분은 특징 가중치 계산을 위하여 패턴공간을 가상으로 분할한 선을 나타낸다. 이 경우 가로 특징축 8개, 세로 특징축 8개로 분할 된 것을 볼 수 있으며, 특징 가중치 값은 수식 (5)를 이용하여 계산한다.

$$W_i = I - \sum_{i=1}^{N} P_i I_i \tag{5}$$

P_i는 분할 이전의 학습패턴 중 분할된 영역에 포함된 학습패턴의 비율이다. I는 분할 이전의 정보량, I_i는 각 분할점들을 기준으로 분할했을 때 각 공간의 정보량이며, 이들은 수식 (3)을 이용하여 계산된다. 또한, N은 특징 i의 최종 분할영역 개수이다.

4. IRPA(Incremental RPA) 기법

RPA 기법은 패턴공간을 재귀적으로 분할하여 대표패턴을 생성한다. 하지만, RPA기법의 특성상 패턴을 구성하는 특징의 개수가 많은 경우, 과도한 분할로 인하여 생성되는 대표패턴 개수가 많아지게 된다. 본 논문에서는 불필요한 대표패턴의 생성을 방지하여 메모리 사용 효율을 높이고 분류 시간을 단축시키기 위해서 점진적으로 대표패턴을 추출하는 IRPA 기법을 제안하며, 다음의 <표 4>는 IRPA 기법의 알고리즘을 보여주고 있다.

〈표 4〉 IRPA 기법

① RPA 기법을 수행하여 대표패턴을 생성한다.
② 가장 많은 학습패턴을 이용하여 생성된 대표패턴 하나를 선택하고, 이때 사용된 학습패턴을 학습패턴 집합으로부터 제거한다.
③ 더 이상 학습할 패턴이 없을 때까지, ①-② 단계를 반복 수행한다.

IRPA 기법은 대표패턴을 추출하기 위하여 RPA 기법을 여러 번 수행하며, 분류를 위한 특징 가중치 값은 최초로 RPA기법을 적용하여 분할된 패턴공간을 이용하여 수식 (5)로 계산한다.

5. 분류 알고리즘

본 논문에서 제안한 RPA, IRPA 기법은 테스트 패턴을 분류하기 위하여 대표패턴들과 수식 (6)으로 거리 계산을 하며, 가장 가까운 대표패턴의 클래스를 출력으로 결정한다. 따라서 기존의 k-NN 기법과는 달리 사전에 최적의 k 값을 구할 필요가 없다.

$$D = \sqrt{\sum_{i=1}^{n} W_i (E_i - Q_i)^2} \tag{6}$$

W_i는 특징 i의 가중치 값이며, E_i와 Q_i는 대표패턴과 테스트 패턴의 i번째 특징값이다. n은 패턴의 특징 개수를 의미한다.

6. 실험 및 분석

본 논문에서 제안한 RPA, IRPA 기법의 성능을 k-NN 기법, WeightVote k-NN 기법, 그리고 EACH 시스템과 비교 검증하였으며, 실험 방법은 Stratified 10-fold Cross-validation 기법을 사용하였다.

6.1 실험 데이터

본 논문에서는 기계 학습의 벤치마크 자료로 많이 사용되는 UCI Machine Learning Database Repository 에서 6개의 데이터셋을 발췌하여 사용하였다[12]. 이들 데이터셋의 모든 특징은 실수 값으로 구성되며, 다음의 <표 5>는 실험자료의 분포를 나타낸다.

〈표 5〉 데이터셋의 패턴 분포

데이터 셋	패턴 개수	특징 개수	클래스 별 패턴 개수					
			1	2	3	4	5	6
Breast-Cancer	699	10	458	241	-	-	-	-
Glass	214	10	70	76	17	13	9	29
Ionosphere	351	34	225	126	-	-	-	-
Iris	150	4	50	50	50	-	-	-
New-Thyroid	215	5	150	35	30	-	-	-
Wine	178	13	59	71	48	-	-	-

Breast-Cancer 데이터셋은 Wisconsin 대학병원의 William H. Wolberg 박사가 정리한 유방암 진단 자료이며[13], Glass 데이터셋은 범죄 수사 연구에 사용하기 위해서 유리를 분석한 자료이다. Ionosphere 데이터셋은 Goose Bay에서 수집된 레이더 데이터이며, Iris 데이터셋은 패턴인식 분야에서 가장 많이 사용되는 꽃잎과 꽃받침의 길이와 너비 수치를 기반으로 식물의 종류를 판별하는 데이터셋이다. New-Thyroid 데이터셋은 갑상선 진단 자료이며, Wine 데이터셋은 이탈리아의 동일 지역에서 세가지 다른 품종으로 재배된 와인의 화학적 분석 결과이다.

6.2 분류 성능

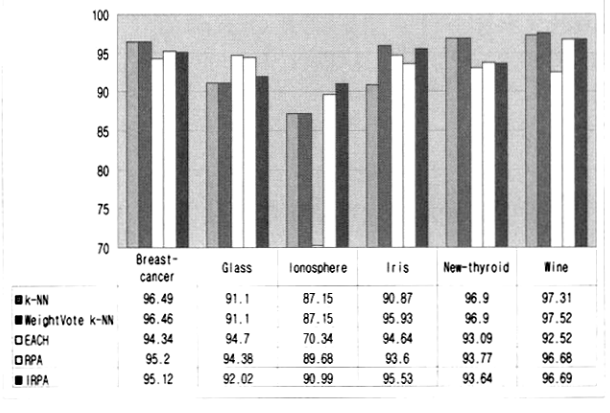
분류 성능 실험에서 k-NN 기법은 Leave-one-out Cross-validation기법으로 계산한 최적의 k값을 사용하였으며[10],

EACH 시스템은 시드 개수 5, 가중치 변화량 0.2를 초기값으로 설정하여 실험하였다. 다음 <표 6>은 각 데이터셋에서 사용된 k-NN 기법의 k값과 k값을 계산하기 위하여 사용된 시간을 나타낸다.

〈표 6〉 분류성능 최적화를 위한 k값 및 계산 시간(Hour)

데이터셋	Breast-Cancer	Glass	Ionosphere	Iris	New-Thyroid	Wine
k값	21	1	1	51	1	19
시간	261	2.26	40.56	0.33	1.61	1.29

다음의 (그림 4)는 분류 성능을 보여주고 있다. 본 논문에서 제안한 RPA, IRPA 기법은 k-NN 기법, EACH 시스템과 비교하여 유사한 성능 또는 향상된 분류 성능을 보여주고 있으며, Ionosphere의 경우, EACH 시스템보다 높은 분류 성능을 보여주고 있다. EACH 시스템이 Ionosphere에서 저조한 성능을 보이는 것은 무작위로 설정한 초기 시드(seed)의 영향으로 볼 수 있으며[9], <표 7>에 나타난 바와 같이 본 논문에서 제안한 기법이 EACH 시스템보다 모든 데이터셋에서 안정적인 성능을 보여준다.



(그림 4) 분류 성능

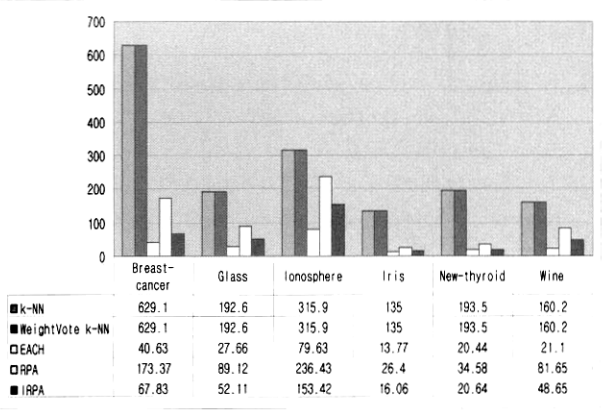
<표 7>은 분류 성능에 대한 표준편차를 보여준다.

〈표 7〉 분류 성능에 대한 표준편차

	Breast-cancer	Glass	Ionosphere	Iris	New-thyroid	Wine
k-NN	2.24	5.37	5.08	7.16	3.66	3.57
WeightVote k-NN	2.26	5.37	5.08	4.21	3.66	3.4
EACH	3.66	5.19	18.13	5.58	4.84	6.29
RPA	2.2	4.94	4.65	4.45	5.03	4.32
IRPA	2.5	5.33	4.73	5.85	5.24	4.25

6.3 메모리 사용량

(그림 5)는 메모리에 저장되는 패턴의 개수를 보여주고 있다. RPA 기법은 k-NN 기법보다 작은 개수의 예제로 구성되며, IRPA 기법의 경우에는 k-NN 기법보다 Breast-cancer 90%, Glass 77%, Ionosphere 53%, Iris 89%, New-Thyroid 91%, Wine 69% 정도 줄어드는 것을 볼 수 있다. EACH 시스템과 비교할 때, RPA, IRPA 기법의 대표패턴 개수가 전반적으로 많이 생성되며, 특히 다른 데이터셋에 비하여 특징의 개수가 많은 Ionosphere(34)의 경우 두 배 이상 많은 것을 볼 수 있다. 그 이유는 RPA기법의 특성상 특징의 개수가 많은 데이터셋의 경우 과다한 분할이 발생하기 때문이다. 그 결과, 학습패턴이 1개만 존재하는 분할영역이 많게 되며, 이로 인하여 대표패턴 생성의 효과를 무의미하게 만드는 결과를 (그림 5)에서 확인할 수 있다.



(그림 5) 메모리 사용량

7. 결 론

본 논문에서 제안한 RPA, IRPA 기법은 k-NN 기법보다 적은 개수의 대표패턴을 이용하여 유사한 성능을 보여주고 있으며, 데이터셋이 달라질 때마다 최적의 k값을 따로 구할 필요가 없다는 장점을 가진다. 또한, IRPA 기법은 학습패턴 집합으로부터 점진적으로 대표패턴을 추출하는 방법을 이용하여 보다 적은 패턴개수로 k-NN, RPA 기법과 유사하거나 향상된 분류 성능을 보장한다. 한편, EACH 시스템은 초기 시드에 따라서 분류 성능의 편차가 심하지만, 본 논문에서 제안한 기법은 모든 데이터 셋에서 안정적인 분류 성능을 보장한다.

8. 향후 연구

Ionosphere와 같이 특징의 개수가 많은 경우에는 과도한 분할이 발생하여 생성되는 대표패턴이 많아지게 된다. 그러므로, 향후 연구로 이러한 데이터셋에서 불필요한 분할을 방지할 수 있는 방법과 학습에 필요한 특징의 개수를 줄일 수 있는 방법에 대해서 연구할 예정이며, 또한, RPA나 IRPA기법을 기반으로 IF-THEN형태의 규칙을 생성하는 방법을 현재 연구중에 있다.

참 고 문 헌

[1] T. Dietterich, "A Study of Distance-Based Machine Learning Algorithms", Ph. D. Thesis, computer Science Dept., Oregon State University, 1995.

[2] D. Wettschereck and T. Dietterich, "Locally Adaptive Nearest Neighbor Algorithms", Advances in Neural Information Processing Systems 6, pp.184-191, Morgan Kaufmann, San Mateo, CA. 1994.

[3] D. Wettschereck, "Weighted k-NN versus Majority k-NN A Recommendation". German National Research Center for Information Technology, 1995.

[4] D. Aha, "A Study of Instance-Based Algorithms for Supervised Learning Tasks: Mathematical, Empirical, and Psychological Evaluations", Ph. D. Thesis, Information and Computer Science Dept., University of California, Irvine, 1990.

[5] D. Aha, "Instance-Based Learning Algorithms, Machine Learning", Vol. 6, No. 1, pp. 37-66, 1991.

[6] D. Wettschereck and T. Dietterich, "An Experimental Comparison of the Nearest-Neighbor and Nearest-Hyperrectangle Algorithms", Machine Learning, Vol.19, No. 1, pp.1-25, 1995.

[7] S. Salzberg, "A Nearest hyperrectangle learning method, Machine Learning", No.1, pp.251-276, 1991.

[8] D. Wettschereck, et al., "A Review and Empirical Evaluation of Feature Weighting Methods for a Class of Lazy Learning Algorithms", Artificial Intelligence Review Journal, 1996.

[9] 심범식, 정태선, 윤충화, "최근접 초월평면 학습법에서 시드개수의 영향에 대한 분석", 한국정보처리학회 '98 춘계학술대회, 1998.

[10] 이형일, 정태선, 윤충화, 강경식, "재귀 분할 평균법을 이용한 새로운 메모리기반 추론 알고리즘", 한국정보처리학회논문지, Vol.006, No.007, pp.1849-1857, 1999.

[11] Ian H. Witten, Eibe Frank, "Data Mining", Morgan Kaufmann, pp.89~94, 1999.

[12] http://www.ics.uci.edu/~mlearn

[13] O. L. Mangasarian and W. H. Wolberg: "Cancer diagnosis via linear programming", SIAM News, Vol.23, No.5, pp.1 & 18, September, 1990.

한 진 철

e-mail : jchan0415@mju.ac.kr

1998년 명지대학교 컴퓨터공학과(공학사)

2000년 명지대학교 컴퓨터공학과(공학석사)

2004년 명지대학교 컴퓨터공학과 박사과정 수료

2002년~현재 명지대학교 교양 시간강사

2006년~현재 명지대학교 산업기술연구소 전임연구원

관심분야 : 인공지능, 지능형 소프트웨어, 데이터마이닝, 시멘틱웹

김 상 귀

e-mail : kimsk98@mju.ac.kr

1990년 명지대학교 전자계산학과(학사)

1993년 명지대학교 대학원 전자계산학과 (공학석사)

1995~1998년 명지대학교 컴퓨터공학과 (박사수료)

1998~2002년 세경대학 컴퓨터소프트웨어과 전임강사

2002~현재 디지털 C&P 정보관리부 차장

2002~현재 명지대학교 컴퓨터공학과 겸임교수

관심분야 : 인공지능, 지능형 소프트웨어, 데이터마이닝, 패턴인식

윤 충 화

e-mail : yoonch@mju.ac.kr

1979년 9월 서울대학교 자연과학대학 수학과(학사)

1984년 9월 University of Texas at Austin 전산학과(석사)

1989년 7월 Louisiana State University 전산학과(박사)

1990년 3월~현재 명지대학교 컴퓨터공학과 교수

현 재 데이터마이닝 학회 이사

관심분야 : 인공지능, 지능형 소프트웨어, 데이터마이닝