산업기술연구(강원대학교 산업기술연구소 논문집), 제28권 A호, 2008.
Journal of Industrial Technology, Kangwon Natl. Univ., Korea, No. 28 A, 2008.

모델 축소를 위한 그룹 모델 클러스터링 방법에 대한 연구

Group Model Clustering Method for Model Downsizing

박 미 나* 하 진 영**
Park, Mi-Na Ha, Jin-Young

Abstract

Practical pattern recognition systems should overcome very large class problem. Sometimes it is almost impossible to build every model for every class due to memory and time constraints. For this case, grouping similar models will be helpful. In this paper, we propose GMC(Group Model Clustering) to build a large class Chinese character recognition system. We built hidden Markov models for 10% of total classes, then classify the rest of classes into already trained group classes. Finally group models are trained using group model clustered data. Recognition is performed using only group models, in order to achieve reduced model size and improved recognition speed.*

키워드 : *그룹모델, 그룹모델 클러스터링*
Keywords : *Group Model, GMC , Group Model Clustering*

1. 서 론

대부분의 통계적 패턴 인식 시스템은 클래스마다 모델을 미리 훈련하여 저장하고 인식할 데이터가 들어오면 데이터의 특징과 모델과의 유사도가 가장 큰 모델을 선택하여 최종 인식 결과를 보여준다. 클래스의 수가 방대할 경우, 데이터를 모델과 비교하는 과정에서 많은 수의 모델과 비교하게 되므로 인식 시간이 늘어나고, 모델을 저장하기 위한 메모리의 사용도 클 수밖에 없다. 또한 데이터와 서로 유사한 클래스가 여러 개 결정될 수가 있다. 결정된 여러 개의 클래스 중에 어느 클래스에 더 가깝다고 결정하기가 어렵게 된다. 이런 경우에는 가중치 등을 고려하여 그 유사도가 크다고 판단되는 모델로 결정하게 된다. 이 때 가중치에 따라서 그 판단에 오류가 생길 가능성이 있다. 또한 클러스터링과 관련되는 임의의 특징은 모델의 복잡도를 증가시켜 모델이 어느 쪽으로 클러스터링 될지를 혼돈스럽게 한다[1][2]. 이는 혼동을 유발하는 특징이 임의의 클래스 간에 잠재하고 있기 때문이다.

이러한 문제점을 해결하기 위해 본 논문에서는 유사 클래스 간의 별도의 모델을 두지 않고 이를 하나의 그룹 모델로 사용함으로써 모델의 크기를 줄이고 속도를 향상시키며, 여러 개의 인식 후보를 고려하여 그룹 모델 안에서 한 번 더 인식 결과를 찾고 최종 인식결과를 얻는 GMC(Group Model

* 강원대학교 컴퓨터정보통신공학과 박사과정
** 강원대학교 IT 특성화 학부대학 컴퓨터학부 교수, 공학박사

Clustering)를 제안한다.

GMC는 유사 클래스들의 대표 모델을 핵심 특징을 기반으로 휴리스틱하게 정하고 이를 훈련하여 대표 모델을 만든다. 훈련된 모델을 이용하여 데이터를 인식하면 대표 모델로 인식되는 임의의 데이터가 발생한다. 이렇게 인식된 데이터와 대표 모델을 하나의 모델로 그룹화 하는 방법이다. 모델의 수를 축소하고 하나의 그룹모델에는 여러 개의 클래스가 포함되어 있으므로 데이터 인식 후 다수개의 인식 후보를 고려하여 인식률을 측정하고 최종 결과를 보여준다. 본 논문에서는 후보 50순위까지를 고려한다.

이 방법은 모델의 수가 적기 때문에 메모리를 적게 사용하고 인식 속도가 빨라져 시간적 비용을 줄 일 수 있다. 그리고 그룹모델에는 유사한 클래스가 포함되어 있어 가중치 등에 따라 발생 할 수 있는 오류를 최소화 할 수 있다. 본 논문에서는 온라인 한자 데이터를 대상으로 실험하였다.

2. K-Means 클러스터링

클러스터링에서 데이터를 서로 비교하거나 통합을 할 때 가장 먼저 고려해야 하는 것은 데이터의 핵심 특징을 유지하면서 data set의 크기를 줄이는 것이다. 클러스터링은 겹쳐지는 그룹이나 클러스터가 없도록 데이터의 경계를 나누는 것이다. 데이터 집합이 클러스터링 될 때, 모든 경계는 어떤 하나의 클러스터에 속해야 한다. 모든 클러스터는 하나의 데이터로 표현되는데, 이러한 대표적인 데이터를 중심이라 하며, 보통 클러스터내의 데이터 평균으로 구해진다[7].

대표적인 클러스터링 방법인 K-means 클러스터링은 주어진 데이터를 특별한 특징을 기반으로 k개의 클러스터로 나누는 방법이다[3]. 이것은 생물학에서 컴퓨터 그래픽에 걸쳐 넓은 영역에 걸쳐 적용되고 있다[4][5][6].

먼저, 구성하고자하는 클러스터의 개수 k를 결정한다. k 만큼의 데이터를 선택한 후 그 데이터가 중심이 되도록 클러스터를 만든 후 클러스터안의 데이터를 중심으로 다시 중심을 결정한다. 이 데이터가 벡터 공간을 이룬다고 가정하면, 이 알고리즘은 각 클러스터의 분산을 최소화한다.

$$E = \sum_{i=1}^{m} \sum_{j \in S_i}^{k} |x_{ij} - \mu_i|^2 \qquad (1)$$

여기서 k개의 집합은 S_i, i = 1, 2,.....,k 이고 μ_i은 각 집합에 속한 데이터의 중심이다. 이 작업을 반복하면 데이터들이 소속된 집합을 바꾸지 않거나, 중심데이터가 변하지 않는 상태가 될 때까지 반복한다.

이 알고리즘은 연산속도가 빠르기 때문에 실제 널리 쓰이고 있으나 최근 연구에 따르면 이 알고리즘이 n개의 데이터에 대해 최악의 경우 $2^{\Omega(\sqrt{n})}$ 시간이 걸리는 경우가 있음을 보였다[8]. 따라서 최종 결과의 유용 측면에서는 이 알고리즘은 최적 값을 기대 할 수 없다. 그리고 맨 처음 나눈 방법에 대부분 의존한 결과가 나오므로 변별력 있는 결과가 나오기 힘들다. 데이터가 골고루 분포되어 있지 않다면 이 또한 나쁜 결과를 얻을 수도 있다. 하지만, 이 알고리즘은 빠르기 때문에 다른 초기 값으로 여러 번 시도하여 좋은 집합을 얻어낼 수 있으나 좋은 집합을 얻기 위해 많은 시도를 해야 한다.

3. GMC(Group Model Clustering)

통계적 패턴 인식 시스템은 클래스마다 모델을 미리 만들어 저장하고 임의의 입력 데이터가 있을 때 입력된 데이터의 특징을 추출하여 모델집합과의 유사도가 최대인 것을 선택하여 최종 인식 결과를 보여준다. 따라서 많은 시간적 비용이 들며, 유사 클래스 간의 가중치로 인한 오류가 생길 가능성이 있다. 클러스터링과 관련되는 임의의 특징은 모델의 복잡도를 증가시켜 모델이 어느 쪽으로 클러스터링 될지를 혼돈스럽게 한다. 이는 혼동을 유발하는 특징이 임의의 클래스 간에 잠재하고 있기 때문이다. 본 논문에서는 유사 클래스 간의 별도의 모델을 두지 않고 이를 하나의 그룹 모델로 사용함으로써 모델의 크기를 줄이고 속도를 향상시키며, 여러 개의 인식 후보를 고려하여 최종 인식결과를 얻는 GMC(Group Model Clustering)를 제안한다.

3.1 대표모델선택

실험에서 사용될 데이터는 한자데이터이다. 이 데이터의 주된 특징은 한자의 획수를 기본으로 한다. 각각의 클래스마다 같은 획수를 가진 데이터를 기준으로 10%의 임의의 데이터를 추출한다. 추출된 데이터의 수가 일정 기준 값에 미치지 않은 경우 그 클래스에 대해서는 기준 값을 조정하여 데이터를 추출한다. 이렇게 하는 이유는 클래스 별로 데이터가 너무 적은 경우 충분히 모델 훈련이 되지 않아 실험결과에 좋은 영향을 미치지 않기 때문이다.

3.2 모델 그룹화

추출된 데이터를 모델 훈련한다. 이렇게 훈련된 모델로 데이터를 인식하여 해당모델로 인식된 데이터를 분석한다. 아래 그림은 추출된 모델로 훈련 된 모델을 이용하여 인식된 결과를 보이고 있다.

袈	11	袈	11
모델	획수	인식된 데이터	획수

그림 1 해당 모델로 인식된 결과

그림 1은 한자 '가사 가'에 해당하는 모델이 있는 경우 데이터가 인식된 결과이고, 그림 2은 모델이 없는 한자 데이터가 획수가 비슷한 모델로 인식 된 결과이다.

袈	11	葛	11
모델	획수	인식된 데이터	획수

그림 2 유사 모델로 인식된 결과

인식결과를 분석하여 같은 모델로 인식되어진 데이터를 모아 그 모델과 그룹화 한다. 만일 같은 데이터가 다른 모델로 분산되어 인식되어졌을 때는 인식 되어진 횟수를 분석하여 많이 인식된 모델로 데이터를 그룹화 한다.

巧 5 司 5
示 5 司 5
好 4 司 5
叮 5 司 5

그림 3 '司'가 다양한 모델로 인식된 경우

데이터 '司(맡을 사)'를 인식하였을 때 이 데이터는 위의 그림 3에서와 같이 다양한 유사 모델로 인식이 된다. 그러나 그 결과를 살펴보면 '可(옳을 가)'로 가장 많이 인식된다는 것을 알 수 있다. 따라서 모델 '可(옳을 가)'의 그룹에 '司(맡을 사)'를 포함시킨다. 이런 과정을 통해 얻어진 모델 그룹에 대한 정보는 아래 표 1에서 그 예를 보이고 있다.

그룹화 된 모델의 가장 앞부분에 있는 모델은 처음 추출되었던 모델로 그룹모델의 대표모델로 한다. 그룹모델을 구성하기 위한 데이터의 수는 최대 15를 넘기지 않도록 한다. 예를 들어 한 그룹모델이 10개의 데이터를 포함한다고 하면 50순위까지 인식률을 고려하였을 때 모든 클래스에 대해 측정하는 것과 같기 때문이다.

표 1 모델 그룹화

이전모델	그룹화 된 모델
司 5	司 5 可 5 巨 5 圭 6 互 4 古 5
比 4	比 4 孔 3 屯 4 氏 4 扎 4 已 3
防 6	防 6 巧 6 辺 5 会 6 朽 6 号 5
汎 6	汎 6 企 6 竹 6 邯 5 讯 5 当 6
合 6	合 6 吉 6 庄 6 全 6 伍 6
兆 5	兆 5 尼 5 匹 4 灯 6 札 5 汝 6
买 5	买 5 末 5 未 5 头 5

3.3 모델의 훈련

모델 훈련은 대표 모델과 그룹에 속한 데이터를 포함하여 훈련한다. 일반적으로 같은 클래스 데이터의 레이블이 같을 때 파라미터를 합하여 평균을 내어 표준 모델의 확률을 결정한다.

$$\text{if}(Data\_label == C(Data2\_label))$$
$$\Pr(Data) = \Pr(Data) \times N(Data) + \frac{\Pr(Data2)}{N(Data)+1};$$

그림 4 일반적인 모델 연산

$Pr(Data)$는 Data의 획수를 의미하며 $N(Data)$는 데이터의 수를 의미한다. C(Data2_label)은 클래스 내 데이터 레이블을 의미한다.

이와 같이 그룹화 된 데이터의 클래스 데이터도 대표모델과 같은 클래스의 데이터로 가정하고 이 데이터들의 파라미터를 이용하여 모델의 확률로 결정한다.

먼저 데이터가 그룹 내에 포함되어있는지를 확인한다. 포함되어 있다면 그 데이터는 모델과 같은 데이터로 취급하고 일반적인 방법과 같이 모델의 확률을 결정한다.

$$\mathrm{if}(Data\_label == G(Data2\_label))$$

$$\Pr(Data) = \Pr(Data) \times N(Data) + \frac{\Pr(Data2)}{N(Data)+1};$$

그림 5 그룹모델 연산

여기서 *G(Data2_label)*는 그룹에 포함되어 있는 데이터의 레이블을 의미한다. 그룹 내에 *Data_label* 이 있는지를 확인하는 작업이다.

모델 훈련이 끝나고 두 단계의 인식과정을 거친다. 일반적인 방법과 같이 대표모델을 이용하여 인식하고 후에 인식된 데이터와 대표모델과 같은 레이블을 갖는지 확인하고 그렇지 않다면 대표모델과 같은 그룹에 속하는지를 확인한다. 그 과정을 통해 최종인식률을 결정한다.

아래 그림 6은 최종적으로 인식했을 때 후보 2순위까지의 예를 보여주고 있다. 아래 예에서 첫 번째 데이터 거짓 가 '假'를 인식하면 위와 같이 인식이 된다. 인식된 결과는 그룹 모델과 그룹모델 사이를 '/' 으로 분리하였다. 첫 번째 '/'를 만나기 전까지가 모델 배 선 '船'을 대표 모델로 하는 그룹이고 후보 1순위를 의미한다. 두 번째 '/'를 만나기 전까지가 심할 극 '劇'을 대표 모델로 하는 그룹이며 후보 2순위를 의미한다.

예) 假 [船 假 脚 距 過 略 象 細 眼 唆 圈 朗 組 / 劇 鄭 慙 飼 飽 嘉 稼 膈 敲 梛 嶋 璉 瘢 膀 /]

이 그룹에 데이터에 해당하는 모델이 속해 있으면 인식된 것으로 간주한다.

假 [船 假 脚 距 過 略 象 細 眼
唆 圈 朗 組 / 劇 鄭 慙 飼 飽 嘉
稼 膈 敲 梛 嶋 璉 瘢 膀 /]

瑕 [暇 闇 閑 嘩 閉 関 辑 間 暎
蹕 / 號 膠 輿 樗 槻 幔 幀 膘 瞋
惆 憬 獗 懊 /]

價 [瞑 網 價 镇 穗 練 賞 億 緞 /
儐 櫃 礒 犠 礁 穉 戴 藉 穢 /]

柯 [蚌 沿 球 柯 垦 剋 春 悧 律
枳 峠 残 项 / 栂 档 贼 捆 挠 砾
柞 砥 枷 桎 挖 /]

그림 6 후보 2순위까지의 인식 결과

4. 실험 및 결과 분석

4.1 실험과정 및 데이터베이스

실험에서 사용할 데이터 집합을 훈련을 위한 훈련 데이터 집합과 테스트를 위한 테스트 데이터 집합으로 분할하였다. 첫 번째로 훈련 데이터 집합에서 획수를 기준으로 각 클래스 별로 랜덤하게 10%의 데이터를 선택한다. 선택한 데이터를 훈련하여 모델을 만든다. 훈련데이터 집합을 훈련된 모델을 사용하여 인식결과를 얻는다. 인식 결과를 분석하여 임의의 모델과 데이터를 그룹화 한다. 그룹화 된 데이터를 모두 모델 훈련한다. 이때 그룹 내의 데이터를 포함하는 클래스를 모두 이용한다. 이렇게 훈련된 모델은 테스트집합으로 인식 실험을 한다. 이때 최종결과는 두 단계를 거치는데 먼저 인식 과정을 거친 후 그 다음 인식된 데이터가 그 모델을 포함하는 그룹 내에 존재하는지 확인하는 과정을 가진다. 이 과정을 통해 최종 인식률을 결정한다.

본 논문에서는 한자 데이터를 실험 대상으로 삼았다. 훈련데이터의 집합은 총 6821개 클래스인데, 약 3만개의 데이터로 구성되어 있다. 테스트 집합은 총 3000개의 클래스로 약 1만개의 데이터로 구성되어 있다.

4.2 실험결과

표 2 실험결과

모델의 수	인식률	인식속도	크기
6821	92.43 %	1.214 sec	77 MB
693	90.77 %	0.445 sec	7 MB

6821개의 모델에서 1순위 인식률은 92.43%이며, 모델의 수를 약 10%로 줄여 그룹 모델로 만든 693개의 모델에서 50순위까지의 인식률은 90.77%이다. 모델의 메모리사용은 77MB에서 7MB로 약 90%로 줄었다. 인식속도는 약 63% 향상되었다.

인식률이 향상되지 못한 것은 같은 한자일지라도 필기자에 따라 그 획순과 쓰는 방법 등이 조금씩 다르다. 이 실험에서는 한자의 획수에 따라 유사 클래스를 분류하였으므로 같은 한자일지라도 필기 순서 등이 다르면 다른 글자로 인식되었기 때문이다. 따라서 모델을 그룹화 할 때 이 점을 고려한다면 좀 더 좋은 결과를 얻을 수 있을 것이다.

5. 결론 및 향후과제

본 논문에서는 모델의 크기를 줄이기 위한 방법으로 모델을 그룹화 하는 GMC(Group Model Clustering)방법을 제안하였다. 서로 유사한 클래스

에 대해 별도의 모델을 두지 않고 하나의 모델로 그룹화 하는 방법이다. GMC 방법은 모델의 수나 그 크기에서 약 10% 줄었으나, 모델의 수를 줄이기 전보다 인식률 향상이 저조함을 보였다. 이는 모델을 그룹화 할 때 같은 한자 이지만 획수가 서로 다르면 다른 한자로 분류하였기 때문이다. 따라서 향후 이를 개선하여 인식률을 향상시키는 과제가 남아 있다.

참 고 문 헌

[1] P. Carbonetto, N. de Freitas, P. Gustafson, and N. Thompson, "Bayesian Feature Weighting for Unsupervised learning, with Application to object Recognition," *Proc. Ninth Int'l Conf. Artificial Intelligence and Statistics*, 2003.

[2] M.H. Law, M.A.T. Figueiredo, and A.K. Jain, "Simultaneous Feature Selection and Clustering Using a Mixture Model," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 26, no. 9, pp.1154-1166. Sept. 2004.

[3] Stuart P. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Thery*, Vol. 28, No. 2, pp. 129-136, 1982.

[4] Pankaj K. Agarwal and Nabil H. Mustafa, K-means Projective clustering. In PODS'04: *Proceedings of the twenty-third ACM SIGMOD-SIGACT-SIGAT symposium on Principles of database systems*, New York, NY, USA, pp. 155-165, 2004.

[5] Federic Gibou and Ronald Fedkiw, "A fast hybrid k-means level set algorithm for segmentation," *In 4th Annual Hawaii International Conference on Statistics and Mathematics*, pp. 281-291, 2005.

[6] R. Herwig, A.J. Pouska,C.Muller,C. Bull, H.Lehrach, and J O'Brien, "Large-scale clustering of cdna-fingerprinting data," *Genome Research* 9, pp. 1093-1105, 1999.

[7] Earl Gose, Richard Johnsonbaugh, Steve Jost. *Pattern recognition and Image analysis*, Prentice-Hall, 1996.

[8] D. Arthur, S. Vassilvitskii, "How Slow is the k-means Method?" *Proceedings of the 2006 Symposium on Computational Geometry (SoCG)*, 2006.