후처리 웹 문서 클러스터링 알고리즘

임 영 희†

요 약

웹 검색 엔진의 검색 결과를 클러스터링하는 후처리 클러스터링 알고리즘은 그 특성상 일반적인 클러스터링 알고리즘과는 다른 요구조건을 갖는다. 본 논문에서는 이러한 후처리 클러스터링 알고리즘의 요구조건들을 최대한 만족하는 새로운 클러스터링 알고리즘을 제안하고자 한다. 제안된 Concept ART는 문서 클러스터링에 있어 여러 가지 장점을 갖는 개념 벡터와 실시간 클러스터링 알고리즘으로 알려진 Fuzzy ART를 결합한 형태로써, 후처리 클러스터링뿐 아니라 범용의 클러스터링 알고리즘으로도 응용이 가능하다.

A Post Web Document Clustering Algorithm

Younghee Im†

ABSTRACT

The post-clustering algorithms, which cluster the results of Web search engine, have several different requirements from conventional clustering algorithms. In this paper, we propose the new post-clustering algorithm satisfying those requirements as many as possible. The proposed Concept ART is the form of combining the concept vector that have several advantages in document clustering with Fuzzy ART known as real-time clustering algorithms. Moreover we show that it is applicable to general-purpose clustering as well as post-clustering

키워드 : 후처리 클러스터링 알고리즘(post-clustering algorithm), 개념 벡터(concept vector), Fuzzy ART, Concept ART

1. 서 론

전통적인 웹 정보 검색 엔진의 경우, 사용자의 질의에 대한 검색 결과를 질의와의 관련 정도에 따라 순위가 매겨진 매우 긴 문서 목록의 형태로 사용자에게 제공한다. 이는 초기 인터넷상의 정보 부재로 인하여 검색엔진의 목표가 되도록 많은 검색 결과를 보여주는데 있었기 때문이다. 그러나 이제는 넘쳐나는 정보들로 인하여, 오히려 검색 결과의 양이 너무 많아 단순한 문서 목록 형태의 검색 결과만으로는 사용자들이 원하는 정보를 찾기가 더욱 힘들어지고 있다. 따라서 검색 엔진에 의해 제공된 일차 검색 결과를 공통된 주제끼리 그룹화하여 사용자에게 가공된 정보의 형태로 제공하고자 하는 연구들이 최근 들어 활발히 진행되고 있으며[1-5], 이는 자연어 검색 기법과 함께 차세대 정보 검색 서비스의 대안으로 여겨지고 있다[6].

문서 클러스터링 기법은 정보 검색 시스템의 전체 문서 코퍼스(corpus)를 오프라인 상에서 미리 클러스터링하여, 질의 요청 시 해당 질의와 가장 유사한 클러스터에 대해서만 검색을 수행하는 "전처리 클러스터링 기법"과 질의 검색 결과를 온라인 상에서 즉각적으로 클러스터링하는 "후처리 클러스터링 기법"으로 나눌 수 있다. 본 연구에서는 문서 클러스터링 기법 중 후처리 클러스터링 기법에 초점을 맞추고자 한다. 후처리 클러스터링 기법은 그 성격상 전처리 클러스터링 기법 혹은 범용의 클러스터링 기법과는 매우 다르므로, 자체 기법의 성격을 잘 반영한 새로운 평가기준이 요구된다. 관련 연구에 의하면, 후처리 클러스터링 기법의 유효성을 평가하는데 사용되는 평가기준(즉, 기본 요건)들은 아래와 같이 정리된다[1] :

1) 관련성(relevance) : 검색된 문서들을 사용자의 질의와 관련된 것과 그렇지 않은 것으로 클러스터링할 수 있어야 한다.
2) 요약(summaries) : 사용자는 클러스터의 내용이 관심이 있는 것인지 없는 것인지 한 눈에 판단할 수 있어야 한다. 따라서 해당 클러스터에 대한 간결하고 정확한 요약을 제공할 수 있어야 한다.
3) 중복(overlap) : 일반적으로 문서들은 하나 이상의 주제(topic)를 내포하고 있을 수 있으므로 하나의 문서가 여러 개의 클러스터에 속할 수 있어야 한다.

† 정 회 원 : 대전대학교 컴퓨터정보통신공학부 교수
논문접수 : 2001년 4월 16일, 심사완료 : 2001년 11월 12일

4) 발췌문-허용오차(snippet-tolerance) : 웹 문서 전체를 입력으로 수행하는 방법 대신, 검색 엔진에 의하여 리턴된 문서의 발췌문만을 가지고 클러스터링을 수행하여도 좋은 결과를 보여야 한다.
5) 속도(speed) : 실제 정보 검색에 응용되기 위해서는 클러스터링의 수행 속도가 빨라야 한다.
6) 점증성(incrementality) : 클러스터링 수행시간을 줄이기 위하여 모든 검색 결과가 전송될 때까지 기다리는 방법이 아닌, 검색된 문서들이 도착하는 즉시 클러스터링을 수행할 수 있어야 한다.

관련 연구[1]에서 제시한 위의 평가기준 외에도, 다음의 평가기준들을 추가적으로 고려해야 한다.

7) 사전지식(priori knowledge) : 클러스터링에 적용되는 문서 집합들이 질의에 따라 각기 다른 특성을 나타내므로 클러스터의 개수와 같은 사전 지식을 요구하지 않는 방법이어야 한다.
8) 메모리 복잡도(memory complexity) : 상업용 검색 엔진의 경우, 하루에도 수 백만 건의 질의를 처리해야 하므로 검색엔진 서버의 과부하 문제 및 메모리 복잡도 문제 등이 고려되어야 한다.

일반적으로, 인공지능 분야에서 개발된 기존의 클러스터링 알고리즘들은 고차원의 대규모 데이터 집합에 적용하기가 어려우며, 후처리 클러스터링 기법을 위하여 개발된 최근의 알고리즘들도 위의 여러 가지 기준들 중 일부 항목에만 초점을 맞추는 방식을 취하고 있다. Dhillon[7]은 문서들이 포함하고 있는 용어들을 특징량(feature)으로 하여, 전체 문서 집합에 대한 벡터 공간을 모델링하였다. 이렇게 모델링된 문서 벡터들은 매우 고차원적이며, sparse한 특성을 나타내므로, 저차원의 dense한 데이터를 마이닝할 때와는 다른 특성을 갖는 Spherical K-Means 클러스터링 알고리즘을 제안하였다. Spherical K-Means 알고리즘에 의해 생성된 각 클러스터는 단위 유클리디안(Euclidean) 노름을 갖도록 정규화된 중점 벡터, 즉 개념 벡터(concept vector)에 의해 대표된다. Spherical K-Means 알고리즘은 각 문서 벡터가 자신을 포함하는 클러스터의 개념 벡터와 가장 큰 코사인 유사도(cosine similarity)를 갖도록 문서 벡터 공간을 분할한다. 상당히 큰 규모의 문서 집합에 대한 실험 결과, 제안된 알고리즘은 상당히 좋은 클러스터링 결과를 보이며, 각 클러스터는 개념 벡터에 의해 잘 요약된 클러스터 레이블(label)을 갖는다. 그러나 Spherical K-Means 알고리즘은 클러스터의 개수를 미리 지정해줘야 하는 문제점이 있다. 실제로 검색 결과는 질의에 따라 매우 동적인 특성을 보이므로, Spherical K-Means 알고리즘을 후처리 클러스터링에 적용하기에는 어려움이 따른다. 또한 K-Means 알고리즘과 마찬가지로 초기 클러스터가 클러스터링의 성능을 좌우한다는 문제점을 안고 있다. 하지만 그들이 제안한 개념 벡터는 위에서 언급한 특성상, 간단한 계산만으로 코사인 유사도나 클러스터의 응집력을 계산할 수 있기 때문에, 수행 속도에 상당히 민감한 후처리 클러스터링의 계산 복잡도를 줄여줄 것으로 기대되며, 개념 벡터들이 각 클러스터에 대해 지역화되므로 본 연구에서 충족시키고자 하는 후처리 클러스터링의 요약 조건을 만족시키기 위한 좋은 출발점이 된다.

반면, 신경망 ART가 가지고 있는 여러 가지 장점에도 불구하고, 현재까지 문서 클러스터링에 ART를 적용한 연구는, 변형된 ART2를 이용하여 웹 페이지를 분류하는 Vlajic[8]의 연구를 제외하고는 발견되지 않고 있다. Vlajic은 문서 집합을 벡터 모델로 변환한 다음, ART2의 변형을 이용하여 문서 벡터들을 분류하였다. 변형된 ART2는 각 클러스터의 가중치 벡터가 해당 클러스터에 소속된 문서 벡터들의 중점값을 갖도록 학습되며, 경계 변수 대신 가중치 벡터와 문서 벡터사이의 불일치 정도에 대한 허용오차(tolerance) 변수를 두어 분류 강도를 조정하였다.

본 논문에서는 후처리 클러스터링 기법이 갖추어야 할 기본 요건들을 최대한 충족시킬 수 있는 새로운 형태의 클러스터링 알고리즘인 Concept ART(Adaptive Resonance Theory)를 제안하고자 한다. 이는 Dhillon[7]의 개념벡터와 실시간 클러스터링 알고리즘으로 알려진 Fuzzy ART 신경망을 결합한 형태로써, 앞서 언급한 기본 요건 중, 1, 2, 4, 5, 7, 8번째 요건을 모두 만족한다. 또한 Concept ART를 신경망 관점에서 바라보면, Fuzzy ART가 갖는 단점들을 극복한, 보다 강건하고 효율적인 새로운 ART 모델의 개발이라는 데, 그 의의가 있다.

본 논문의 구성은 다음과 같다. 2장에서는 클러스터링 알고리즘의 입력으로 사용될 문서들의 벡터화에 대해 설명하고, 3장과 4장에서는 각각 Concept ART의 근간을 이루는 개념 벡터와 Fuzzy ART에 대해 기술한다. 5장에서는 본 논문에서 새롭게 제안된 Concept ART에 대해 자세히 설명하고, 6장에서는 실험결과 및 분석을 기술한다. 마지막으로 7장에서는 결론 및 향후 연구과제에 대해 논한다.

2. 문서의 표현형태

본 논문에서는 후처리 클러스터링을 위한 문서의 표현형태로써 벡터공간 모델을 사용하고자 한다. 따라서 각 문서들은 가중치가 부여된 용어 빈도수의 벡터로써 표현된다. 이때 앞서 언급했듯이, 검색 결과의 클러스터링은 온라인 상에서 실시간으로 수행되어야 하므로 성능 평가시 속도가 매우 중요한 평가기준이 된다. 따라서 문서 전체에 대해 문서 벡터를 구성하는 대신 문서의 일부만으로 문서 벡터를 구성하는 방법 등이 제안되었다[1, 2, 5] :

1) Leouski[2]는 전형적인 잡지 기사(article)의 경우, 처음 두 개의 문단에 기사 내용의 요약이 포함되는 경향이 있다는 점에 착안하여, 문서의 제목과 처음 두 개 문단만을 사용하여 문서 벡터를 구성하였다.
2) Zamir[1,5]는 사용자의 질의를 포함하는 문서 전체를 다운로드받아 문서 벡터를 구성하는 대신, 검색 엔진이 제공하는 발췌문에 대해서만 문서 벡터를 구성하여도 클러스터링 결과가 만족할 만한 성능을 나타냄을 보였다.

위의 연구결과들을 종합해 볼 때, 검색엔진에 의해 제공되는 발췌문만을 가지고 문서 벡터를 구성하는 방법론은 클러스터링의 정확도를 어느 정도 유지하면서 클러스터링의 탐색 공간을 크게 줄일 수 있다. 또한 문서 제목의 경우, 해당 문서의 전체 내용을 대표하는 특성을 가지므로 문서의 제목 역시 문서 벡터 구성 시 매우 중요한 요소가 된다. 따라서 본 논문에서는 발췌문과 해당 문서의 제목만을 입력으로 문서 벡터를 구성함으로써 클러스터링의 속도를 향상하고자 한다(평가기준 4 : 발췌문-허용오차). 또한 발췌문과 제목만을 문서 표현으로 이용할 경우, 해당 클러스터링 툴은 사용자의 클라이언트 머신(client machine)에 탑재되어 실행될 수 있으며, 이는 검색 엔진 서버의 과부하를 줄일 수 있다는 또 다른 장점을 갖는다(평가기준 5 : 속도 및 8 : 메모리 복잡도). 다음은 문서 벡터로의 변환을 위한 일반적인 전처리 과정이다[9].

1) 전체 문서 집합으로부터 모든 용어(term)들을 추출한다.
2) 의미없는 용어, 즉 "stop list"에 등록된 단어들을 제거한다.
3) 각 문서에 대하여 용어의 빈도수를 계산한다.
4) 휴리스틱에 의해, 매우 높은 빈도수(high-frequency)를 갖는 용어와 매우 낮은 빈도수(low-frequency)를 갖는 단어는 기능어(function word)로 간주하여 제거한다.
5) 위 과정의 수행 후, d개의 용어가 남았다고 가정하면, 각 단어에 1부터 d까지의 인덱스를 할당한다. 마찬가지로 각 문서에 1부터 n까지의 인덱스를 할당하면, 전체 문서 집합에 대한 벡터공간 모델인 $d \times n$차원의 행렬 D가 완성된다. 이때 $D_{i,j}$는 j번째 문서에서의 i번째 용어에 대한 tf/idf(term frequency/inverse document frequency) 값이다

단, 각 문서가 검색 엔진으로부터 얻은 웹 문서라면 단계 1의 수행 전에 HTML 태그를 제거해주는 추가 작업이 필요하다. 이상의 과정들을 거쳐 구성된 문서 벡터들은 고차원 벡터의 형태를 띠게 되며, sparsity가 90%이상인 매우 sparse한 벡터가 된다.

본 논문에서는 문서의 벡터화를 위해 MC[10] 프로그램을 사용한다. MC는 대규모의 문서 집합으로부터 아주 빠르게 문서 벡터를 생성해주며, 생성된 문서 벡터는 nonzero 값만을 저장하는 CCS(Compressed Column Storage)[11] 포맷으로 저장된다. 따라서 저장 공간을 절약할 수 있으며, Concept ART에서의 코사인 유사도 계산시 계산량을 크게 줄일 수 있다.

3. 개념 벡터(Concept Vector)

클러스터링 알고리즘에서 두 문서간의 유사도를 결정하는 문제는 클러스터링 알고리즘의 선택 못지 않게 중요한 문제이다. 본 연구에서는 코사인 유사도로 두 문서간의 유사도를 측정하고자 한다. 코사인 유사도는 이해하기 쉽고, sparse 벡터에 대해 계산이 단순하기 때문에 정보 검색이나 텍스트 마이닝에서 널리 사용되는 유사도이다[9]

본 논문에서는 각 문서 벡터 $X_1, X_2, \cdots, X_n$가 단위(unit) L_2 노름(norm)을 갖도록 정규화한다[7]. 이러한 정규화는 문서 벡터들의 방향성(direction)만을 유지하게 해주므로, 문서의 길이가 다르더라도 같은 주제를 다루는 문서들(즉, 유사한 용어들로 구성된 문서들)을 유사한 문서 벡터로 변환해 주는 효과가 있다. 또한 두 문서 벡터 X_i와 X_j 사이의 코사인 유사도는 다음과 같이 두 벡터사이의 내적(inner product)으로 간단히 구할 수 있다.

$$\begin{aligned} S(X_i, X_j) &= X_i^T X_j = \|X_i\| \|X_j\| \cos(\theta(X_i, X_j)) \\ &= \cos(\theta(X_i, X_j)) \end{aligned} \quad (3\text{-}1)$$

여기서 두 벡터사이의 각(angle)은 $0 \le \theta(X_i, X_j) \le \pi/2$이다.

n개의 문서 벡터들이 c개의 클러스터 $\pi_1, \pi_2, \cdots, \pi_c$로 나누어진다고 가정하자. 각 클러스터 π_j의 대표 벡터(representative vector)로 가장 널리 사용되는 평균(mean) 벡터 또는 중점(centroid) 벡터는 다음과 같이 정의된다.

$$M_j = \frac{1}{n_j} \sum_{X \in \pi_j} X \quad (3\text{-}2)$$

여기서 n_j는 클러스터 π_j에 속해 있는 문서의 수이다. 이때 만약 중점 벡터 M_j를 다음과 같이 단위 노름을 갖도록 정규화하면, 중점 벡터의 방향성만을 갖는 개념 벡터 C_j를 정의할 수 있다[7].

$$C_j = \frac{M_j}{\|M_j\|} \quad (3\text{-}3)$$

위와 같이 정의된 개념 벡터 C_j는 다음과 같은 특성을 갖는다. $R^d_{\ge 0}$상의 임의의 단위 벡터 Z에 대해, 다음의 Cauchy-Schwarz 부등식을 유도할 수 있다.

$$\sum_{X \in \pi_j} X^T Z \le \sum_{X \in \pi_j} X^T C_j \quad (3\text{-}4)$$

위의 식 (2-4)에 의해 개념 벡터 C_j는 클러스터 π_j에 속해 있

는 모든 문서 벡터에 대해 가장 근접한 코사인 유사도를 갖는 벡터임을 알 수 있다. 이러한 개념 벡터는 문서 벡터의 sparse한 특성을 그대로 유지하므로, 간단한 계산만으로 코사인 유사도나 클러스터의 응집력 등을 계산할 수 있다. 따라서 수행 속도에 상당히 민감한 후처리 클러스터링의 계산 복잡도를 크게 줄일 수 있다. 또한 개념 벡터들이 각 클러스터에 대해 지역화되므로[7], 본 연구에서 충족시키고자 하는 후처리 클러스터링의 요약 조건을 만족시킬 수 있는 좋은 아이디어를 제공한다. 즉, 클러스터링 수행 결과를 사용자에게 어떤 형태로 서비스 할 것인가(visualization)는 클러스터링 과정 못지 않게 매우 중요한 과제이다. 따라서 클러스터 내 문서들의 내용을 보다 쉽게 이해할 수 있는 레이블을 제공한다면, 사용자는 클러스터에 대한 레이블만을 보고 자신이 원하는 정보들이 포함되어 있는 클러스터를 선택할 수 있을 것이다.

n개의 문서 벡터들이 c개의 클러스터 $\pi_1, \pi_2, \cdots, \pi_c$로 나누어진다고 가정하면, 문서 클러스터 π_j의 키워드(keyword)는 용어 클러스터 Word_j로 표현할 수 있다. 클러스터 π_j의 대표 벡터인 개념 벡터 C_j의 각 용어 중 다른 개념 벡터에서의 가중치보다 큰 가중치를 갖는 용어는 용어 클러스터 Word_j에 속하게 된다[7] :

$$\text{Word}_j = \{ k\text{th word} : 1 \le k \le d, C_{k,j} \ge C_{k,m}, 1 \le m \le c, m \ne j \} \quad (3\text{-}5)$$

이때 d는 전체 용어의 개수이다. 이렇게 구성된 각 클러스터에 대한 용어 클러스터 Word_j는 개념 벡터가 해당 클러스터에 대해 지역화되는 특성을 가지므로, 비교적 좋은 키워드들을 제공해 준다.

또한 각 클러스터에 소속된 문서들 중 해당 클러스터의 개념 벡터와 가장 큰 코사인 유사도를 갖는 문서를 요약(summary)으로 제공함으로써, 단순한 용어의 나열에서는 얻기 어려운 각 클러스터에 대한 직관적인(intuitive) 이해가 가능하다.

$$Summary_j = \arg \max_{x \in \pi_j} \{ \cos(\theta(X, C_j)) \} \quad (3\text{-}6)$$

본 논문에서는 sparse한 문서 벡터의 클러스터링에 있어 여러 가지 장점을 가지고 있는 개념 벡터를 실시간 클러스터링 알고리즘으로 알려진 ART에 적용함으로써 보다 강건하고 효율적인 후처리 클러스터링 알고리즘을 개발하고자 한다.

4. Fuzzy ART(Adaptive Resonance Theory)

Carpenter와 Grossburg에 의해 제안된 ART는 기존에 학습되었던 것이 새로운 학습에 의해 지워지지 않도록 새로운 지식을 전체 데이터베이스에 일관성 있는(self-consistent) 방법으로 통합한다. 실시간 클러스터링 알고리즘으로 알려진 ART는 다른 신경망에 비해 다음과 같은 장점을 갖는다. 첫째, ART는 비교사 학습(unsupervised learning)에 의해 입력 패턴을 클러스터링하므로, 사전에 학습 데이터(training set)를 통한 훈련없이 새로운 입력 패턴을 학습할 수 있다. 둘째, ART는 기존 신경망들의 딜레마인 Stability-Plasticity 문제를 해결한다. 신경망에서 stability란 이전에 학습한 패턴들에 대한 기억을 안정적으로 유지하는 능력을 말하며, plasticity란 이전에 학습한 적이 없는 새로운 패턴을 처리할 수 있는 능력을 말한다. ART는 입력 패턴과 학습된 클러스터 간의 비교를 통해, 이미 학습된 클러스터에 영향을 미치지 않으면서 학습을 수행할 수 있는 Reset 메커니즘을 사용하여 이 딜레마를 해결하였다. 셋째, ART는 경계 변수(vigilance parameter) 값에 따라 클러스터링의 분류 결과를 조정할 수 있다. 즉, 경계 변수의 값을 크게 주면, 좀 더 세분화되고 구체적인 클러스터들을 얻을 수 있다. 또한 학습이 완료된 클러스터에 대한 가중치 값들은 해당 클러스터에 속해 있는 패턴들에 대한 대표 벡터(exemplar vector)로 해석될 수 있다.

ART에는 이진 벡터를 클러스터링하는 ART1과 아날로그 벡터를 클러스터링하는 ART 2가 있으며, Fuzzy ART[12]는 ART1의 교집합(intersection) 연산을 퍼지 집합 이론의 min 연산으로 대치함으로써 ART1이 아날로그 벡터에 대하여 학습할 수 있도록 하였다. Fuzzy ART 알고리즘의 세부적인 수행 과정은 다음과 같다[12]. 먼저, 입력 벡터 $\mathbf{X}_i = (X_{1,i}, X_{2,i}, \cdots, X_{d,i})$, $i = 1, \cdots, n$는 각 콤포넌트가 [0, 1]의 값을 갖으며, $\mathbf{W}_j^{(t)} = (W_{1,j}^{(t)}, W_{2,j}^{(t)}, \cdots, W_{d,j}^{(t)})$, $j = 1, \cdots, c$는 카테고리 j에 대한 가중치 벡터라 하자. 이때 n은 입력 패턴의 수이고, d는 입력 패턴의 차원, c는 카테고리의 개수이다. 또한 Fuzzy ART의 동적 특성을 결정하는 변수들은 선택 변수(choice parameter) $\alpha(>0)$, 학습 변수 $\beta(\in [0,1])$, 경계 변수 $\rho(\in [0,1])$ 등이 있다.

초기화 : 카테고리 개수 c와 변수 α, β, ρ값을 초기화하고, 초기 가중치 벡터는 첫 번째 다음과 같이 초기화한다 :

$$W_{1,j} = W_{2,j} = \cdots = W_{d,j} = 1, \; j = 1, \cdots, c \quad (4\text{-}1)$$

활성화 함수(Activation Fucntion : AF) : 입력 패턴과 가중치 벡터 사이의 매칭 정도를 측정하는 활성화 함수는 퍼지 집합 이론의 min 연산을 이용하여 다음과 같이 계산한다.

$$AF(W_j^{(t)}, X_i) = \frac{\sum_{k=1}^{d} \min(X_{k,i}, W_{k,j}^{(t)})}{\alpha + \sum_{k=1}^{d} W_{k,j}^{(t)}}, \; j = 1, \cdots, c \quad (4\text{-}2)$$

클러스터의 선택 : ART는 경쟁 학습(competitive learning)

의 Winner-Take-All(WTA) 전략을 통해 입력 패턴들을 학습한다. 따라서 Fuzzy ART는 다음과 같이 각 클러스터에 대한 활성화 함수 값에 의해 가장 유사한 클러스터 유닛 $j*$을 선택한다.

$$j* = \arg \max_{j=1,\cdots,c} \{AF(W_j^{(t)}, X_i)\}. \quad (4\text{-}3)$$

Resonance 유닛의 선택 : 식 (4-3)에 의해 선택된 클러스터에 대해 다음의 매칭 함수(Matching Function : MF) 값에 경계 변수 조건을 적용하여 resonance 유닛을 선택한다

$$MF(W_{j*}^{(t)}, X_i) = \frac{\sum_{k=1}^{d} \min(X_{k,i}, W_{k,j*}^{(t)})}{\sum_{k=1}^{d} X_{k,i}} \geq \rho \quad (4\text{-}4)$$

만일 클러스터 $j*$가 위의 경계 변수 조건을 만족하면, 해당 입력 패턴을 학습할 수 있도록 가중치 갱신이 발생하고, 그렇지 않으면 해당 클러스터의 활성화 함수를 reset하고(즉, $AF(W_{j*}^{(t)}, X_i^{(t)}) = -1$), 조건을 만족하는 클러스터를 찾을 때까지, 식 (4-3)에 의해 새로운 $j*$를 탐색한다.

가중치 갱신(또는 학습) : 선택된 카테고리 $j*$에 대해 조건 (4-4)가 만족되면, 다음의 식에 의해 가중치 벡터를 조정한다.

$$W_{j*}^{(t+1)} = \beta \cdot \min(W_{j*}^{(t)}, X_i) + (1-\beta) \cdot W_{j*}^{(t)} \quad (4\text{-}5)$$

Fuzzy ART는 아날로그 입력을 처리할 수 있도록 ART1을 일반화한 것이므로, ART1의 구조적 문제점을 그대로 안고 있다[13]. 즉 입력 패턴의 적용 순서에 매우 민감하며, resonance 유닛을 찾기 위해 최악의 경우 모든 활성화 함수 값을 정렬해야 한다. 따라서 본 논문에서는 위와 같은 Fuzzy ART의 문제점을 해결하는 동시에 정보 검색 결과를 클러스터링하는 문제에 적용이 가능한 새로운 형태의 Concept ART를 제안하고자 한다.

5. Concept ART

Concept ART의 기본 아이디어는 각 클러스터 유닛의 가중치 벡터가 해당 클러스터의 개념 벡터가 되도록 하는 것이다. 이는 Dhillon[7]의 Spherical K-Means 알고리즘과 같이 각 클러스터의 개념 벡터와 입력 패턴의 코사인 유사도를 계산하여 가장 유사한 클러스터로 해당 입력 패턴을 할당하는 작업이지만, Spherical K-Means 알고리즘과는 달리 클러스터의 개수와 같은 사전 지식이 필요없는 비교사 학습이며, 새로운 입력 패턴이 제시되었을 때 전체 시스템을 재학습할 필요없이 점증적 학습(incremental learning)이 가능하다는 장점을 갖는다. 즉, Concept ART는 개념 벡터와 ART가 갖는 장점을 취한 새로운 형태의 클러스터링 알고리즘이다.

초기화 : 카테고리 개수 c는 1로 초기화하고, 입력 패턴들을 단위 L_2 노름을 갖도록 정규화한다. 또한 초기 가중치 벡터는 첫 번째 입력 패턴으로 초기화한다 :

$$W_1^{(0)} = X_1 \quad (5\text{-}1)$$

Concept ART에서 입력 패턴과 가중치 벡터 사이의 매칭 정도는 코사인 유사도에 의해 측정되므로, 식 (5-1)에 의해 첫 번째 입력 패턴과 초기 클러스터 유닛 사이의 코사인 값은 항상 1이 된다. 따라서 사용자가 어떤 값의 경계 변수 ($\rho \in [0,1]$)를 주더라도 첫 번째 패턴이 항상 첫 번째 카테고리에 할당됨을 보장할 수 있다.

활성화 함수 : 입력 패턴과 가중치 벡터 사이의 매칭 정도를 측정하는 활성화 함수는 다음과 같이 두 벡터 사이의 코사인 유사도로 계산된다.

$$AF(W_j^{(t)}, X_i) = \cos(\theta(W_j^{(t)}, X_i)) = X_i \cdot \frac{W_j^{(t)}}{\|W_j^{(t)}\|} \quad (5\text{-}2)$$

이때 가중치 벡터 $W_j^{(t)}$는 클러스터 j로 분류된 입력 패턴들의 합이다 :

$$W_j^{(t)} = \sum_{X_i \in \pi_j} X_i \quad (5\text{-}3)$$

식 (5-3)에서 π_j는 현재까지 클러스터 j에 할당된 문서들의 집합이다. 만약 클러스터 j의 중점 벡터를 $M_j^{(t)}$라 하면(식 (2-2) 참조), 해당 클러스터의 개념 벡터 $C_j^{(t)}$는 다음과 같이 구할 수 있다.

$$C_j^{(t)} = \frac{M_j^{(t)}}{\|M_j^{(t)}\|} = \frac{W_j^{(t)}}{\|W_j^{(t)}\|} \quad (5\text{-}4)$$

즉, 하나의 클러스터의 개념 벡터는 중점 벡터를 따로 계산할 필요없이 해당 클러스터에 할당된 입력 패턴들의 합을 단위 벡터로 정규화함으로써 간단히 구할 수 있다. 따라서 본 논문에서 제안된 Concept ART는 클러스터 유닛의 가중치 벡터를 해당 클러스터에 할당된 입력 패턴들의 합이 되게 하고, 활성화 함수 계산시에만 가중치 벡터를 정규화한 개념 벡터를 사용함으로써 알고리즘을 단순화하였다.

매칭 함수 : 만약 활성화 함수와 매칭 함수가 다음의 조건

$$MF(W_1^{(t)}, X_i) > MF(W_2^{(t)}, X_i) \Leftrightarrow AF(W_1^{(t)}, X_i)) > AF(W_2^{(t)}, X_i) \quad (5\text{-}5)$$

을 만족하도록 선택된다면, 활성화 함수에 의해 선택된, 입력

패턴과 가장 유사한 클러스터의 가중치 벡터가 경계 변수 조건을 만족하지 않을 경우, 그 다음 유사한 클러스터를 탐색할 필요가 없다[13]. 즉, 가장 큰 활성화 함수 값 $AF(W_{j*}^{(t)}, X_i)$에 대한 매칭 함수 값 $MF(W_{j*}^{(t)}, X_i)$이 경계 변수를 만족하지 않는다면, 식 (5-5)에 의해 나머지 클러스터에 대한 매칭 함수 값도 경계 변수 조건을 만족하지 않으므로 별도의 탐색(search) 과정없이, 곧바로 새로운 카테고리를 하나 생성하고, 해당 입력 패턴을 할당하면 된다.

식 (5-5)을 만족하는 가장 단순한 형태의 활성화 함수와 매칭 함수의 선택은 다음과 같다[13].

$$AF(W_j^{(t)}, X_i) \equiv MF(W_j^{(t)}, X_i) \tag{5-6}$$

따라서 본 논문에서는 매칭 함수를 식 (5-2)의 활성화 함수와 일치하도록 정의함으로써 리셋(reset)이 발생할 경우 수행되는 또 다른 resonance 유닛의 선택을 위한 탐색 과정을 제거하였다. 이는 수행속도가 성능 평가의 중요한 척도가 되는 후처리 클러스터링에 Concept ART가 적용될 경우, 별도의 탐색 과정이 필요 없으며 매칭 함수를 따로 계산할 필요가 없으므로 중요한 장점이 될 수 있다.

Resonance 유닛의 선택 : 앞서 설명했듯이 Concept ART에서는 매칭 함수와 활성화 함수가 같으므로 다음의 식에 의해 resonance 유닛을 선택할 수 있다.

$$AF(W_{j*}^{(t)}, X_i) \geq \rho \tag{5-7}$$

이때 $j* = \arg \max_{j=1,\cdots,c} \{AF(W_j^{(t)}, X_i)\}$이다.

입력 패턴과 가장 유사한 클러스터의 가중치 벡터가 식 (5-7)의 경계 변수 조건을 만족하지 않을 경우, 별도의 탐색 과정없이 새로운 카테고리를 생성하고, 해당 입력 벡터를 가중치 벡터로 할당한다.

가중치 갱신(또는 학습) : 선택된 카테고리 $j*$에 대해 식 (5-7)이 만족되면, 입력 패턴을 카테고리 $j*$에 할당하고 다음의 식에 의해 해당 가중치 벡터를 조정한다.

$$W_{j*}^{(t+1)} = W_{j*}^{(t)} + X_i \tag{5-8}$$

Concept ART에서는 개념 벡터의 계산을 간단하게 하기 위하여, 해당 클러스터의 가중치 벡터를 소속된 입력 패턴들의 합이 되도록 한다. 따라서 가중치 갱신시 학습률을 고려할 필요가 없으며, 사용자가 선택해야 하는 유일한 변수는 클러스터링의 형태를 조정할 수 있는 경계 변수뿐이다. 즉, 경계 변수가 클수록 ART의 클러스터들은 좀 더 세분화되고 구체적인 형태를 취하게 된다.

본 논문에서 제안된 Concept ART는 고차원의 sparse 한 문서 벡터의 클러스터링에 효율적인 개념 벡터와 클러스터의 개수와 같은 사전 지식이 필요없는 실시간 클러스터링 알고리즘인 ART를 결합한 새로운 형태의 후처리 클러스터링 알고리즘이다.

Step0. Normalize input pattern with L_2 norm.
Initialize Weights :

$$W_1^{(0)} = X_1$$

Step1. While Stopping Condition is false, do Step 2-7
Step2. For each training input, do Step 3-6
Step3. Set activation of all F_2 to zero
Step4. Compute Activation Function :

$$AF(W_j^{(t)}, X_i) = X_i \cdot \frac{W_j^{(t)}}{\|W_j^{(t)}\|}, \quad j = 1, \cdots, c$$

Step5. Find $j*$ with max activation
Step6. Test for reset :
If $AF(W_{j*}^{(t)}, X_i) \geq \rho$ then

$$W_{j*}^{(t+1)} = W_{j*}^{(t)} + X_i$$

else new processing element allocation :

$$c = c + 1$$
$$W_c^{(t+1)} = X_i$$

Step7. Test for stopping condition

(그림 1) Concept ART 알고리즘

6. 실험 결과 및 분석

본 논문에서는 제안된 Concept ART의 유용성을 보이기 위해 다음의 세 가지 데이터 집합에 대한 실험을 수행하고 한다.

1) **Iris** : Iris 데이터는 많은 연구들에서 클러스터링 알고리즘의 성능 평가를 위한 벤치마킹(bench-marking) 데이터로써 널리 사용된다. 세 개의 클러스터가 각각 50개의 데이터로 구성되며, 각 데이터는 4가지 특징량을 갖는다.
2) **CLASSIC3** : CLASSIC3는 잘 알려진 세 개의 문서 데이터 집합 MEDLINE, CISI, CRANFIELD(ftp : //ftp.cs.cornell.edu/pub/smart)으로부터 각각 50개씩의 문서들을 임의로 추출하여 전체 150개의 문서들로 구성된다. 여기서 MEDLINE은 의학 저널의 요약문들로 구성되며, CISI와 CRANFIELD는 각각 정보 검색 논문들의 요약문과 항공학 관련 논문들로 구성되어 있다. CLASSIC3의 경우 각 문서 집합들의 정확한 클러스터를 알고 있으므로, 문서 클러스터링에 대한 Concept ART의 정확도를 측정할 수 있다.
3) **GUINEA** : 후처리 클러스터링 알고리즘으로써의 Concept ART의 유효성을 검증하기 위하여, 검색어 "Guinea"에 대해 검색 엔진 Google이 리턴해준 185개 문서의 발췌문과 문서 제목으로 구성된 문서 데이터이다.

6.1 Iris에 대한 실험

다음은 Iris에 각각 Fuzzy ART와 Concept ART를 적용한 결과이다. 이때 입력 순서가 서로 다른 두 개의 Iris 데이터 집합(Iris1 & Iris 2)을 생성하여, 입력 패턴의 적용 순서에 따른 성능 변화도 함께 살펴보고자 한다. Fuzzy ART의 경우, 선택 변수 $\alpha = 10^{-5}$, 학습률 변수 $\beta = 1$로 설정하였으며, Concept ART와의 비교를 위해 입력 패턴들을 단위 L_2 노름을 갖도록 정규화하였다. 다음은 Fuzzy ART의 클러스터링 결과이다.

〈표 1〉 Fuzzy ART를 이용한 IRIS의 클러스터링 결과

ρ = 0.84(Iris 1) (86%)

	category 1	category 2	category 3
class 1	50		
class 2		30	20
class 3		1	49

ρ = 0.84(Iris 2) (86%)

	category 1	category 2	category 3
class 1	50		
class 2		30	20
class 3		1	49

ρ = 0.86(Iris 1) (84.7%)

	C 1	C 2	C 3	C 4	C 5
class 1	43	7			
class 2			43	7	
class 3			16	33	1

ρ = 0.86(Iris 2) (90%)

	C 1	C 2	C 3	C 4
class 1	47	3		
class 2			45	5
class 3			10	40

다음은 Concept ART의 클러스터링 결과이다.

〈표 2〉 Concept ART를 이용한 IRIS의 클러스터링 결과

ρ = 0.90(Iris 1) (96.7%)

	C 1	C 2	C 3
class 1	50		
class 2		45	5
class 3			50

ρ = 0.90(Iris 2) (96.7%)

	C 1	C 2	C 3
class 1	50		
class 2		45	5
class 3			50

ρ = 0.97(Iris 1) (96.7%)

	C 1	C 2	C 3	C 4
class 1	50			
class 2		44	3	3
class 3			32	18

ρ = 0.97(Iris 2) (96.7%)

	C 1	C 2	C 3	C 4
class 1	50			
class 2		39	6	5
class 3				50

위의 실험 결과, Concept ART가 Fuzzy ART보다 더 좋은 클러스터링 결과를 보임을 알 수 있으며, Concept ART의 경우, 입력 패턴의 순서가 바뀌더라도 인식률에 큰 차이를 볼 수 없다. 또한 경계 변수 값이 커질수록, 좀더 세분화되고 구체적인 클러스터들이 생성된다. 그러나 Fuzzy ART의 경우에는, 데이터 집합 Iris 1에 대해서 세분화된 클러스터들의 인식률이 오히려 더 나빠지는 경우도 발생한다.

Iris 데이터에 대한 실험 결과를 종합해 볼 때, 클러스터링하고자 하는 데이터 집합이 저차원의 dense한 벡터라 하더라도 Concept ART는 우수한 클러스터링 결과를 보임을 알 수 있다. 이는 Concept ART가 데이터 집합들을 정규화하여 각 데이터의 방향성만을 유지하면서, 해당 클러스터들에 소속된 모든 데이터들과 코사인 유사도가 가장 큰 개념 벡터(식 (2-4))를 ART의 가중치로 저장하기 때문에 가능하다. 따라서 Concept ART는 후처리 클러스터링뿐만 아니라, 범용의 클러스터링 알고리즘으로써도 그 응용이 가능하다.

6.2 CLASSIC3에 대한 실험

후처리 클러스터링에 Concept ART를 적용해 보기에 앞서, 각 문서 데이터의 정확한 클러스터를 이미 알고 있는 CLASSIC3 데이터 집합을 적용해봄으로써, Concept ART가 매우 고차원이면서, sparse한 특성을 갖는 문서 데이터의 클러스터링에 적합한지를 평가하고자 한다. 우선 150개의 문서 데이터를 MC 프로그램을 사용하여 벡터화한다. 이때 불용어와 0.5%이하의 low-frequency, 15%이상의 high-frequency를 갖는 단어를 제거한다[7]. 그 결과 최종적으로 3474개의 용어가 남게 되며, 따라서 각 문서 벡터는 3474차원을 갖는 고차원 벡터가 된다. 이때 각 문서 벡터는 평균 50개 정도만의 nonzero 콤포넌트를 갖는(98% sparsity) 매우 sparse한 벡터이다. 다음은 Concept ART를 이용하여 CLASSIC3를 클러스터링한 결과이다. 마찬가지로 입력 패턴의 적용순서에 따른 성능의 차이를 살펴보기 위해 순서가 다른 데이터 집합을 여러개 생성하여 실험하였다.

실험 결과를 살펴보면, 순서가 다른 각 데이터 집합에 대해, 모두 비교적 좋은 클러스터링 결과를 보임을 알 수 있다.

또한 경계 변수 값이 커짐에 따라, 클러스터가 점점 세분화되므로 인식률이 증가함을 볼 수 있다. 즉, Fuzzy ART에서처럼, 클러스터가 세분화됨에도 불구하고, 인식률은 오히려 낮아지는 구조적인 문제점은 발생하지 않았다. 따라서 사용자는 경계 변수 값을 변화시킴으로써 클러스터의 분류 강도를 조정할 수 있다.

〈표 3〉 Concept ART를 이용한 CLASSIC3의 클러스터링 결과

• **Sequence 1)**

$\rho = 0.008$ (88.67%)

	G1	G2	G3
med	10	5	35
cisi	2	48	
cran	50		

$\rho = 0.010$ (92.67%)

	G1	G2	G4	G5
med	8	4	3	35
cisi	1	49		
cran	3		47	

$\rho = 0.015$ (93.33%)

	G1	G2	G3	G4	G5	G6
med	4		1	3	8	34
cisi	2	6		42		
cran	6		44			

• **Sequence 2)**

$\rho = 0.008$ (62.00%)

	G1	G2	G3	G4
med	1	25	20	4
cisi		24	8	18
cran			49	1

$\rho = 0.015$ (84.00%)

	G1	G2	G3	G4	G5
med	2	13	5		30
cisi		28	3	17	2
cran			49	1	

$\rho = 0.02$ (92.67%)

	G1	G2	G3	G4	G5	G6	G7
med	1		5	13		30	1
cisi		31	3		14	2	
cran			49		1		

• **Sequence 3)**

$\rho = 0.008$ (72.00%)

	G1	G2	G3	G4
med		37	11	2
cisi	46	4		
cran	1	49		

$\rho = 0.01$ (83.33%)

	G1	G2	G3	G4	G5
med		19	19	10	2
cisi	45	5			
cran	1	49			

$\rho = 0.015$ (94.00%)

	G1	G2	G3	G4	G5	G6	G7
med		5	18	9		16	2
cisi	44	3			3		
cran	1	49					

Pentium III 800 CPU를 적재한 PC에서 3474차원의 문서 벡터 150개를 클러스터링하는데 걸리는 시간이 1초 미만이므로 수행속도에 대단히 민감한 후처리 클러스터링 알고리즘에도 적용이 가능함을 알 수 있다. 이는 Concept ART에서 입력 패턴과 가중치 벡터 사이의 유사도 계산이 단순한 벡터의 내적만으로 가능하며, 이때에도 CCS 포맷에 의해 nonzero 콤포넌트에 대해서만 곱을 수행하기 때문이다. 또한 입력 패턴과 가장 유사한 클러스터가 경계 변수 값을 만족하지 못할 경우, 또 다른 resonance 유닛을 찾기 위한 별도의 탐색과정이 필요없이 바로 새로운 카테고리를 생성하므로, 전체적인 알고리즘의 계산 복잡도를 크게 낮추었다.

다음으로 Concept ART에 의해 생성된 각 클러스터의 가중치 벡터가 해당 클러스터의 개념 벡터로써, 해당 클러스터들의 내용을 얼마나 잘 요약하는 지를 살펴보고자 한다. 이를 위해 Sequence1의 경계변수 $\rho = 0.008$에 대한 용어 클러스터를 식 (3-5)에 의해 구성하고, 각 용어 클러스터에서 가중치 값이 가장 큰 용어 10개를 <표 4>에 보였다.

〈표 4〉 용어 클러스터의 예

Word$_1$(CRANFIELD)	Word$_2$(CISI)	Word$_3$(MEDLINE)
heat	library	nickel
plate	libraries	amyloid
transition	literature	fatty
hypersonic	technical	fetal
temperature	scientific	renal
theory	books	selenite
transfer	citation	acids
roughness	user	amyloidosis
supersonic	research	patients
surface	index	day

위의 <표 4>에서 볼 수 있듯이 각 클러스터에 대한 용어 클러스터는 각 클러스터에 소속된 문서들의 내용을 잘 표현해준다. 이는 SVD의 singular 벡터가 전체 문서 벡터에 대한 글로벌한 특성을 갖는 반면, 개념 벡터는 각 클러스터에 대해 지역화되기 때문이다[7]. 따라서 Concept ART에 의해 생성된 개념 벡터는 각 클러스터의 키워드로 사용할 수 있으며, 이는 Concept ART가 후처리 클러스터링의 요약 조건을 만족시킬 수 있다는 것을 의미한다.

6.3 GUINEA에 대한 실험

이번 실험은 질의어 "Guinea"에 대해, 검색 엔진 Google이 리턴해준 상위 185개 문서들을 문서 제목과 발췌문만을 가지고 클러스터링하는 실험이다. 먼저 HTML 태그를 모두 제거한 다음, 불용어와 두 개 이하의 문서에서 발생한 용어는 제거한다[3]. 그 결과 각 문서 벡터는 159차원의 sparse 벡터(96% sparsity)가 된다. 이를 Concept ART에 의해 클러스터링한 결과는 다음의 <표 5>와 같다. 이때 키워드는 각 클러스터에 대한 용어 클러스터 중, 상위 6개의 용어들이며, 요약은 각 클러스터에 소속된 문서들 중 해당 클러스터의 개념 벡터와 가장 큰 코사인 유사도를 갖는 문서의 제목이다.

실험 결과 Concept ART는 Guinea를 포함하는 여러 그룹, 즉, Guinea pig(모르모트), Papua New Guinea, Guinea-

Bissau, Equatorial Guinea등을 잘 분류해 낼 수 있으며, 개념 벡터를 기반으로 하는 키워드와 요약은 해당 클러스터의 내용을 잘 표현해주고 있다(평가 기준 2 : 요약). 다시 말해, Concept ART는 전체 문서가 아닌 발췌문과 문서 제목만으로 이루어진 문서 벡터만으로도 각 클러스터들을 잘 분류할 수 있고(평가 기준 1 : 관련성, 평가 기준 2 : 발췌문-허용오차, 평가 기준 8 : 메모리 복잡도), 클러스터의 개수와 같은 사전 지식이 요구되지 않으며(평가 기준 7 : 사전 지식), 사용자의 필요에 따라 경계 변수값을 변경하여 분류 강도를 조정할 수도 있다. 또한 Concept ART와 sparse한 문서 벡터의 특성상, 그 수행 속도가 매우 빠르다는 장점을 갖는다(평가 기준 5 : 속도).

〈표 5〉 Concept ART를 이용한 GUINEA의 클러스터링 결과

	ρ = 0.01, Cluster 1, size = 55
keywords	pig, pigs, page, home, cavy, cavies
summary	Guinea Pig Page(Cavy, Cavies)
	ρ = 0.01, Cluster 2, size = 53
keywords	papua, png, online, links, government, independent
summary	Pupua New Guinea Search Engine
	ρ = 0.01, Cluster 3, size = 28
keywords	bissau, africa, world, center, ocean, top
summary	Guide To Law Online : Guinea-Bissau
	ρ = 0.01, Cluster 4, size = 23
keywords	equatorial, republic, weather, detailed, part, human
summary	Ethnologue : Equatorial Guinea
	ρ = 0.01, Cluster 5, size = 13
keywords	conakry, adventure, september, excite, flight, paradise
summary	Welcome to ParadiseLive
	ρ = 0.01, Cluster 6, size = 10
keywords	information, network, african, country, december, internet
summary	Guinea - Consular Information Sheet
	ρ = 0.01, Cluster 7, size = 3
keywords	species, located, http, location, www
summary	Papua New Guinea OrchidNews : Species Photos

7. 결론 및 향후 연구과제

본 논문에서 제안된 Concept ART는 서론에서 제시된 후처리 클러스터링 알고리즘의 여러 가지 평가 기준 중, 관련성, 요약, 발췌문-허용오차, 속도, 사전 지식, 메모리 복잡도 등 대부분의 요구조건들을 만족한다. Zamir에 의해 제시된 요구 조건 중, 점증성은 검색 엔진에 의해 리턴된 문서가 도착하는 즉시 클러스터링을 수행함을 의미하는데, 이런 의미의 점증성은 문서 데이터를 벡터 공간으로 모델링하는 경우, 불가능한 요구조건이다. 왜냐하면 각 문서들이 갖는 특징량, 즉 용어들이 전체 문서 집합으로부터 추출된 것이므로 벡터 모델의 경우, 모든 집합들이 모두 도착한 후에야 문서 벡터를 구성할 수 있다. 그러나 일반적으로 언급되는 점증적 갱신의 경우, Concept ART는 전체 시스템을 재구성할 필요없이 새로운 입력 패턴을 학습할 수 있으므로 점증적 학습이 가능하다. 또한 실험에서도 살펴보았듯이, Concept ART는 구조적 특성상, 후처리 클러스터링뿐만 아니라 범용의 클러스터링 알고리즘으로도 그 응용이 가능하다. 향후 연구에서는 Concept ART에 퍼지 개념을 도입하여, 후처리 클러스터링의 나머지 요구 조건, 중복을 허용할 수 있는 Fuzzy Concept ART를 개발하고자 한다.

참 고 문 헌

[1] O. Zamir and O. Etzioni, "Web Document Clustering : A Feasibility Demonstration," Proceedings of the 19th International ACM SIGIR Conference on Research and Development in Information Retrieval(SIGIR '98), pp.46-54, 1998.

[2] A. Leouski and W. B. Croft, "An Evaluation of Techniques for Clustering Search Results," Technical Report IR-76, University of Massachusetts at Amherst, 1996.

[3] D. S. Modha and W. S. Spangler, "Clustering Hypertext With Applications To Web Searching," Proceedings of ACM Hypertext Conference, 2000.

[4] M. A. Hearst and J. O. Pedersen, "Reexamining the Cluster Hypothesis : Scatter/Gather on Retrieval Results," Proceedings of ACM SIGIR '96, pp.76-84, 1996.

[5] O. Zamir and O. Etzioni, "Grouper : A Dynamic Clustering Interface to Web Search Results," available at http : //www.cs.washington.edu/zamir/papers/www8.ps.gz

[6] 박민우, "검색엔진의 과거와 현재 그리고 미래", 마이크로소프트웨어, pp.220-235, 2000.

[7] I. S. Dhillon and D. S. Modha, "Concept Decomposition for Large Sparse Text Data using Clustering," Technical Report RJ 10147(9502), IBM Almaden Research Center, 1999.

[8] N. Vlajic and H. C. Card, "Categorizing Web Pages using Modified ART," IEEE Canadian Conference, Vol.1, pp.313-316, 1998.

[9] W. B. Frakes and R. Baeza-Yates, "Information Retrieval : Data Structures and Algorithms," Prentice Hall, Englewood Cliffs, New Jersey, 1992.

[10] J. J. Fan, "MC : A Fast Sparse Matrix Generator For Large Text Collections," available at http : //www.cs.

utexas.edu/users/jfan/dm/.

[11] Available at http : //www.cs.utexas.edu/users/inderjit/Resources/sparse_matrices.

[12] G. A. Carpenter, S. Grossburg, and D. B. Rosen, "Fuzzy ART : An Adaptive Resonance Algorithm for Rapid, Stable Classification of Analog Patterns," Proceedings of 1991 International Conference Neural Networks, Vol. II, pp.411-416, 1991.

[13] A. Baraldi and E. Alpaydin, "Simplified ART : A New Class of ART Algorithms," International Computer Science Institute, TR 98-004, 1998.

임 영 희

e-mail : yheeim@dju.ac.kr

1994년 고려대학교 전산학과 졸업(학사)

1996년 고려대학교 대학원 전산학과 (이학석사)

2001년 고려대학교 대학원 전산학과 (이학박사)

1996년~2001년 고려대학교 전산학과 시간강사

2001년~현재 대전대학교 컴퓨터정보통신공학부 강의전담교수

관심분야 : 정보검색, 텍스트 마이닝, 데이터 마이닝, 데이터베이스 보안 등