Journal of Korean
Data & Information Science Society
2003, Vol. 14, No.4 pp. 853~861

An Improved K-means Document Clustering using Concept Vectors

Yang Kyu Shin[1)]

Abstract

An improved K-means document clustering method has been presented, where a concept vector is manipulated for each cluster on the basis of cosine similarity of text documents. The concept vectors are unit vectors that have been normalized on the n-dimensional sphere. Because the standard K-means method is sensitive to initial starting condition, our improvement focused on starting condition for estimating the modes of a distribution. The improved K-means clustering algorithm has been applied to a set of text documents, called Classic3, to test and prove efficiency and correctness of clustering result, and showed 7% improvements in its worst case.

Keywords : *K-means clustering, document clustering, concept vector.*

1. 서론

다변량 분석의 하나인 군집분석은 일정 수의 속성을 가진 개체들을 특성 속성을 기준으로 분류하여 유사한 개체들끼리 군집으로 분류하는 것을 의미한다. 군집분석의 일반적 과정은 개체의 속성을 나타내는 변수의 측정, 변수를 이용한 개체들 간의 유사성 측정, 유사한 개체들끼리 그룹화하는 군집화, 그리고 군집의 특성 분석 등 네 가지 과정으로 이루어진다. 그 중 군집화 방법은 크게 계층적 방법과 비 계층적 방법으로 나누어지는데, 계층적 방법은 미리 군집의 수를 정하지 않으며 한 군집 내에 다른 군집들이 포함되어 있어 트리 구조를 취하는 형식이다. 반면에 비 계층적 방법은 미리 군집의 수를 결정하여 군집화하고 군집 내에 다른 군집이 포함되지 않는 형식이다.

1) 경상북도 경산시 유곡동 290번지 대구한의대학교 정보과학부 교수
E-mail : yks@dhu.ac.kr

군집화에 대한 이론적인 관점은 군집화를 밀도추정문제로 보는 것이다. 군집화 할 데이터 각각은 관측변수 외에도 데이터가 속할 군집을 나타내는 관측되지 않은 은닉변수를 포함한다고 가정한다. 즉, 데이터는 관측변수와 은닉변수가 혼합된 혼합모델에서 발생한다고 볼 수 있으며, 데이터가 속할 군집의 이름은 데이터 발생 당시에는 감추어져 있다고 가정한다. K개의 군집 $C_1, \ldots, C_K$를 가진 혼합모델 M에서 데이터 x에 대한 확률 $\Pr(x \mid M)$은

$$\Pr(x \mid M) = \sum_{i=1}^{K} W_i \cdot \Pr(x \mid C_i, M)$$

으로 나타낼 수 있으며, 이 때 W_i는 가중치이다. 이러한 관점에서 보면 최적의 군집화란 혼합모델 M에 관해 데이터의 우도를 극대화시킬 수 있는 모수 C_i와 W_i를 찾는 것이다. K-평균 군집화는 각 군집이 구면상의 가우스분포로 표현되며, 각 데이터는 하나의 군집에만 속하고, 가중치 W_i는 모두 동일하다는 세 가지 특징을 가진다.

한편 인터넷과 정보통신 기술의 발전으로 전자저널, 뉴스그룹, 전자우편, 웹 문서, 그리고 각종 업무용 문서 등 다양한 전자 문서들이 폭증하고 있는 상황에서 이들 문서를 유사한 내용을 가진 것끼리 군집화할 필요성은 계속 증가하고 있다. 하지만 문서 군집화는 다변량 분석에서 가정하는 일반적인 군집화 환경과 비교할 때 중요한 차이점이 있다. 즉, 문서를 하나의 데이터로 볼 때 문서 데이터가 가지는 관측변수인 단어의 수가 문서마다 일정하지 않다는 점이다. 따라서 문서의 군집화에 일반적으로 사용되는 벡터공간 모델에서는 주어진 문서 집합을 고차원의 희소행렬로 표현하는데, 이 행렬에는 문서에서 의미를 가지는 단어만을 추출하여 속성(관측변수)으로 설정한 후, 속성의 빈도수를 값으로 가지는 열벡터로 각각의 문서를 나타내게 된다.

행렬계산을 이용하여 유사 문서를 찾는 방법에 관한 연구가 최근에 많이 이루어지고 있다. Berry(1995) 등은 singular value decomposition을 이용하여 단어와 문서 사이에 서로 연관된 잠재적인 관계를 찾는 방법을 제안하였고, Kolda(1997)는 행렬의 근사법(approximation)을 효율적으로 계산할 방법을 제시하였으며, Papadimitriou(1998) 등은 확률적 프로젝션을 이용한 행렬 근사법을 연구하였다. 그리고 Dhillon(2000, 2001)등은 행렬을 이용한 대규모 문서의 군집화 방법으로 구면형 K-평균 군집화 기법을 제안하였다.

문서의 군집화에 사용하는 기법들은 대부분 반복 계산을 통한 수렴의 응용으로 볼 수 있는데, 이들은 많은 수의 지역 극소값들(local minima) 중 어느 하나로 수렴시키게 된다. Bradley와 Fayyad(1998)는 주어진 초기 입력에서 더욱 정교한 시작 조건을 생성하는 방법을 제안하였는데, 이 방법은 분포의 형태를 추정하는 효율적인 기법에 바탕을 두고 있으며, 생성된 정교한 시작 조건은 반복 알고리즘을 통해 더 나은 지역 극소값으로 수렴하게 됨을 밝혔다. 본 논문에서는 이 방법을 K-평균 군집화에 적용하여 개선시켰으며, 개선된 K-평균 군집화 기법을 사용하여 텍스트 문서의 군집화를 실험하였고, 그 결과를 분석하였다.

2. 벡터공간 모델

문서를 군집화하기 위해 먼저 주어진 문서에서 의미를 갖지 않는 관사, 접속사, 조사 등을 제거하고 고유한 단어만을 모두 추출한다. 그런 다음 각 문서에 나타난 단어 각각의 빈도수를 계산하여 빈도수가 특별히 큰 단어나 빈도수가 극히 적은 단어들을 제거하는데 이러한 단어들을 기능성 단어라 부른다. 이와 같이 의미가 없는 단어와 기능성 단어를 제거하고 난 다음 남은 단어들이 m개일 때 각 단어를 알파벳순으로 정렬한 후 1부터 m까지 번호를 붙인다. 또한 주어진 문서집합의 문서들이 n개일 때 이들 문서 역시 1부터 n까지 번호를 붙인다.

문서 i에 나타나는 단어 j의 개수가 f_{ij}이고 단어 j를 포함하는 문서의 수가 d_j라고 하면, R^m 상에서 n개의 문서벡터 $\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n$을 만들 수 있다. 즉, $1 \le j \le m$인 j에 대해 벡터 $\vec{x}_i$의 j번째 원소 x_{i_j}는 $x_{i_j} = t_{i_j} \times g_j \times s_i$과 같이 나타낼 수 있다. 이 때 t_{i_j}는 f_{ij}만을 이용해 구할 수 있는 값으로 특정 문서 내에서 단어의 상대적인 중요도를 나타낸다. 그리고 g_j는 d_j로부터 구할 수 있는데, 문서집합 전체에서 계산된 단어의 중요도를 의미한다. 마지막으로 s_i는 벡터 $\vec{x}_i$의 정규화 요소이다. 이와 같이 가중치 값들을 곱하는 이유는 문서벡터 사이의 판별 능력을 증가시키기 위함이고, 결과적으로 군집화를 포함한 문서 처리의 다양한 분야에서 정확도와 효율성을 함께 증가시킬 수 있게 된다.

벡터 $\vec{x}_i$의 j번째 원소를 나타내기 위해 곱한 세 가지 가중치들을 구하는 방법은 매우 다양하다. Kolda(1997)는 t_{i_j} 를 구하는 방법이 5가지, g_j 를 구하는 방법이 5가지, 그리고 s_i를 구하는 방법이 2가지가 있음을 밝혔다. 따라서 하나의 벡터를 구하는 방법은 모두 50가지가 된다. 본 논문에서는 일반적으로 사용하는 방식을 따르도록 하는데, 이때 $t_{i_j} = f_{ij}$, $g_j = \log(n/d_j)$, 그리고 s_i는 $\vec{x}_i$가 L^2 norm 길이로 1이 되도록 하는 표준화 상수다. 그러므로 s_i를 $s_i = \left(\sum_{j=1}^{m} (t_{i_j} g_j)^2 \right)^{-1/2}$으로 설정하여 크기가 1인 문서벡터의 방향을 얻게 된다.

3. K-평균 알고리즘

3.1 개념벡터

문서벡터 $\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n$은 단위 구 R^m 상의 점들이다. 또한 가중치들의 속성상 모든 문서벡터는 음이 아니게 된다. 따라서 음이 아닌 문서벡터들의 유사성을 계산하기 위해 벡터의 내적을 이용하는데, 단위 벡터 $\vec{x}$와 $\vec{y}$의 코사인 유사성은 다음과 같다:

$$\vec{x}^{\,t} \vec{y} = \|\vec{x}\| \times \|\vec{y}\| \cos(\langle \vec{x}, \vec{y} \rangle) = \cos(\langle \vec{x}, \vec{y} \rangle)$$

여기서 $\langle \vec{x}, \vec{y} \rangle$는 두 벡터 사이의 각으로 $0 \le \langle \vec{x}, \vec{y} \rangle \le \pi/2$이다.

문서벡터 $\vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n$을 K개의 군집으로 분할한 결과가 $C_1, C_2, \cdots, C_K$라고 하면 다음 식 (1)이 성립 한다.

$$\bigcup_{j=1}^{K} C_j = \{ \vec{x}_1, \vec{x}_2, \cdots, \vec{x}_n \} \qquad (1)$$
$$C_a \cap C_b = \emptyset, \text{ if } a \neq b$$

또한 K개의 군집 각각에 대해, 군집에 속한 문서벡터들의 평균벡터(또는 중심점)를 이용하여 개념벡터를 정의한다. 먼저 군집 C_j의 평균벡터 δ_j는 식 (2)와 같다.

$$\delta_j = \frac{1}{n_j} \sum_{x \in C_j} \vec{x} \qquad (2)$$

여기서 n_j는 군집 C_j에 포함된 문서의 수이다. 또한 이 평균벡터는 단위 길이가 아닐 수 있으므로 이를 단위 길이로 정규화한 개념벡터를 정의한다. 군집 C_j의 개념벡터 $\vec{x}_j$는 식 (3)과 같다.

$$\vec{x}_j = \frac{\delta_j}{\|\delta_j\|} \qquad (3)$$

개념벡터가 가지는 중요한 성질 중 하나는 임의의 음 아닌 단위벡터 $\vec{y}$에 대해 다음과 같은 Cauchy-Schwarz 부등식 (4)가 성립한다.

$$\sum_{x \in C_j} \vec{x}^{\,t} \vec{y} \leq \sum_{x \in C_j} \vec{x}^{\,t} \vec{x}_j \qquad (4)$$

즉, 개념벡터는 군집 C_j 내의 모든 문서벡터들에 코사인 유사도로 가장 가까운 벡터임을 의미한다. 만약 군집내의 모든 벡터들이 동일하다면 $\sum_{x \in C_j} \vec{x}^{\,t} \vec{x}_j$은 최대값 1을 가질 것이다. 반대로 벡터들이 다양하고 넓게 분포되면 값이 작아질 것이며 궁극적으로는 0에 가까이 간다.

3.2 K-평균 군집화 방법

K-평균 군집화 방법은 효율적이고 효과적인 반복 휴리스틱을 사용하는 알고리즘으로 랜덤하게 생성한 초기 군집, 문서벡터 집합, 그리고 군집의 수 K를 입력으로 받아 K개의 m 차원 개념벡터 집합을 생성해 낸다.

[K-평균 군집화 알고리즘]

1단계: $j = 1, \ldots, K$에 대해 다음을 반복

a. 반복 단계 t=0에서의 초기 군집 $C_j^{(0)}$를 랜덤하게 생성한다.

b. 군집 $C_j^{(0)}$의 개념벡터 $\vec{x}_j^{(0)}$는 $C_j^{(0)}$의 중심점(centroid)으로 설정한다.

2단계: 각 문서벡터 $\vec{x}_i (i = 1, \ldots, n)$에 대해, $\vec{x}_i$와 코사인 유사도가 가장 큰 개념벡터를 찾는다. 다음으로 이전 개념벡터 $\vec{x}_j^{(t)}$에서 유도되는 새로운 군집

$C_j^{(t+1)}$을 식 (5)로 계산한다.

$$C_j^{(t+1)} = \left\{ \vec{x} \in \{\vec{x_i}\} \ : \ \vec{x}^{\,t} \vec{x}_j^{(t)} \geq \vec{x}^{\,t} \vec{x}_l^{(t)}, 1 \leq l \leq K \right\}, 1 \leq j \leq K \quad (5)$$

다시 말해, $C_j^{(t+1)}$는 개념벡터 $\vec{x}_j^{(t)}$에 가장 가까운 모든 문서 벡터들의 집합이다. 이 때 어떤 문서벡터가 여러 개의 개념벡터와 같은 정도로 가깝다면 임의의 군집에 배정한다.

3단계: 군집에 대응하는 새로운 개념벡터를 계산한다.

4단계: 만약 "종료 조건"을 만족하면, 개념벡터 $\vec{x_j}^{(t+1)}$ $(1 \leq j \leq K)$을 돌려주고 멈춘다. 그렇지 않으면 t를 1 증가시키고 다시 2단계로 진행한다.

이상의 K-평균 군집화 방법은 데이터의 우도를 국소적으로 가장 크게 하는 모수 집합 $\{ C_1, C_2, \cdots, C_K \}$로 수렴시키게 된다. 이때 국소적으로 모수의 극대값에 수렴한다는 것은 초기값에 매우 민감하다는 의미를 포함하는데, 군집화의 초기값을 결정하는 초기화 방법은 지금까지 별로 연구된 바 없다. 대부분의 군집화 연구에서 초기값의 설정은 사용자가 직접 제시하거나 혹은 기계적인 방법으로 랜덤하게 생성하여왔다. 여기서는 시작점의 적절한 선택이 숫자데이터로 표현될 수 있는 문서집합의 군집화에 성능을 개선할 수 있음을 보이고자 한다.

4. K-평균 군집화 방법의 개선

군집화 문제의 해는 각 군집 모델의 모수화이며, 모수화는 데이터의 결합확률밀도의 최빈수를 찾아 그 최빈수를 군집의 중심점으로 두는 것이다. 따라서 군집화 기법은 밀도를 추정하고 추정된 밀도함수의 최빈수를 찾는 작업이라고도 볼 수 있다. 하지만 고차원 공간에서의 밀도 추정은 매우 어려운 문제로서 시작점을 최빈수에 가까운 점으로 풀이하고자 한다.

기본적인 아이디어는 모집단 데이터에 대한 부분 샘플링을 반복하면 최빈수 근처의 대표값에서 샘플링이 이루어진 것처럼 수렴하게 될 것이란 점이다. 이 때 작은 부분 샘플 상에서 군집화 하여 얻은 결과는 모집단 데이터의 실제 평균의 우수한 초기 추정치로 볼 수 있다. 다만 주의해야 할 점은 작은 부분 샘플을 잡음으로서 노이즈를 가진 추정치가 발생할 가능성이 매우 높기 때문에 이러한 문제를 해결하기 위해 노이즈 교정기법을 사용한다.

노이즈 교정기법은 먼저 J개의 부분샘플을 추출하여 각각을 K-평균 기법으로 군집화하여 J개의 군집 추정치를 생성한다. 다음으로 J개의 답에 존재할 수 있는 노이즈를 제거하기 위해 J개의 샘플 각각에 만들어진 K개의 군집을 다시 K-평균으로 군집화하여 최종적인 K개의 군집을 생성한다.

[K-평균 군집화 알고리즘 개선을 위한 초기치 계산 알고리즘]

1단계: $SC\_Set = \emptyset$

2단계: $i=1,\ldots,J$ 에 대해 다음을 반복

a. S_i를 임의로 잡은 문서 데이터집합의 작은 부분샘플로 둔다.

b. $SC_i = Sample\_KMeans(Start\_Point, S_i, K)$.

c. $SC\_Set = SC\_Set \cup SC_i$

3단계: $i=1,\ldots,J$ 에 대해 다음을 반복

$RC_i = KMeans(SC_i, SC_Set, K)$을 계산한다.

4단계: $RC = ArgMin_{RC_i}\{Distortion(RC_i, SC_Set)\}$이라 둔다.

5단계: RC를 돌려준다.

즉, K-평균 군집화 방법을 개선하기 위한 초기치 계산법은 먼저 J개의 부분샘플 집합 $S_1,\ldots,S_J$ 를 랜덤하게 샘플링하는 것으로 시작한다. 부분샘플 각각을 K-평균 방법으로 군집화 하는데, 이 때 공집합인 군집이 발생하면 공집합 대신 부분샘플 전체의 평균으로 치환한 후 다시 군집화한다. J개의 부분샘플 각각에서 얻은 K개씩의 군집화 결과 집합 $SC_1,\ldots,SC_J$를 합 집합하여 SC_Set를 만든다. $SC_1,\ldots,SC_J$를 초기집합으로 하여 SC_Set을 다시 K-평균 방법으로 군집화한 결과 군집 $RC_1,\ldots,RC_J$를 얻는다. 이와 같이 만들어진 결과 군집 RC_i는 SC_Set 상에서 데이터의 왜곡이 최소가 되는 초기점이 된다.

초기치 계산 알고리즘에 사용된 함수 중에서 $KMeans(Start_Point, S_i, K)$는 3.2절의 방법처럼 출발점 $Start_Point$를 이용하여 문서 데이터 집합 S_i를 군집화하여 K개의 m차원 개념벡터를 생성함을 나타낸다. $Sample_KMeans(Start_Point, S_i, K)$는 $KMeans(Start_Point, S_i, K)$처럼 군집화하는데, 이 때 만약 K개의 군집 중 공집합이 발생하면 (이러한 경우는 작은 부분샘플에서 흔히 발생할 수 있는 상황임) 해당되는 빈 군집의 초기 출발점에서 거리가 가장 먼 원소로 채운 후 다시 $KMeans(Start_Point, S_i, K)$를 실행함을 나타낸다.

$Distortion(RC_i, SC_Set)$은 K개의 평균에 대한 추정치 벡터 RC_i와 데이터 집합 SC_Set에서 각 벡터와 가장 가까운 개념벡터와의 코사인 값들의 합을 계산한다. 계산된 결과 스칼라로 데이터 집합에 대한 군집들의 정확한 정도를 측정한다.

5. 실험 결과 분석

본 논문에서 사용한 Classic3 문서 데이터 집합은 3,893개의 파일로 구성되어 있는데, 그 중 1,400개의 CRANFIELD 문서는 항공분야의 논문에서 발췌한 것이고, 1,033개의 MEDLINE 문서는 의학 논문집에서 발췌한 것이며, 나머지 1,460 개의 CISI 문서는 정보검색에 관련된 논문들이다. 이 문서 데이터 집합에 해당하는 벡터공간 모델을 구성하기 위해, 문서에서 고유한 단어들을 추출하여 15% 미만이나 85% 이상 나타난 단어들을 제거하고 최종적으로 52개의 단어로 희소행렬을 구성하였다(Duff(1992

등). 즉, 이 희소행렬은 52차원 열벡터가 문서의 개수에 해당하는 3,893개의 행으로 구성된다. 실험은 200회의 군집화 실행동안 수렴하게 될 개념벡터와의 유사성을 모두 더한 목표함수와 반복계산 회수에 대해 각각 측정하고, 같은 데이터에 대해 3개와 6개의 군집으로 각각 군집화를 실행하였다. 개념벡터와의 유사성의 합인 목표함수는 군집 $C_j\ (1 \le j \le K)$의 밀착성을 나타내는 $\sum_{x \in C_j} \vec{x}^{\,t}\, \vec{x}_j$를 모든 군집에 대해 합친 것으로 식 (6)으로 정의한다.

$$\sum_{j=1}^{K} \sum_{x \in C_j} \vec{x}^{\,t}\, \vec{x}_j \tag{6}$$

1) 3개의 군집으로 군집화한 목표함수와 반복횟수

3개의 군집으로 군집화할 때 목표함수와 반복횟수는 각각 <표 1>과 <표 2>에 나타나 있다. <표 1>의 결과에서처럼 목표함수의 특성은 분산을 제외하면 거의 같다고 볼 수 있다. 목표함수에 대한 분산이 개선된 K-평균 군집화가 더 작다는 것은 초기값 조정으로 목표함수 값에 큰 변화 없이 수렴하고 있음을 보인다. 또한 <표 2>처럼 이러한 결과를 얻는데 소요된 반복횟수 역시 최대값, 평균 모두 개선된 K-평균 군집화에서 약 10%정도 줄어들었음을 볼 수 있다. 즉, 수렴 속도가 약 10% 정도 개선된 결과임을 알 수 있다.

<표 1> 3개의 군집으로 처리한 목표함수

방법 / 변수	K-평균 군집화	개선된 K-평균 군집화
최소값	2501.97	2501.96
최대값	2533.76	2533.76
평균	2533.3397	2533.3967
분산	10.1094	10.0645

<표 2> 3개의 군집으로 처리한 반복횟수

방법 / 변수	K-평균 군집화	개선된 K-평균 군집화
최소값	11	14
최대값	102	92
평균	43.67	34.18
분산	768.03	3604.8

2) 6개의 군집으로 군집화한 목표함수와 반복횟수

<표 3>과 <표 4>는 6개의 군집으로 군집화할 때 목표함수와 반복횟수이다. <표 3>의 목표함수 결과 역시 3개의 군집화처럼 분산 외에는 거의 결과는 같다고 볼 수 있다. 하지만 이 경우 개선된 K-평균 군집화에서 목표함수의 분산이 특별히 큰 이유는 수렴할 때 함수 값이 큰 폭으로 변한다고 추정된다. 이에 대한 타당한 근거로는 <표 4>의 반복횟수의 최대값이 K-평균 군집화에서 약 7.5%의 줄어든다는 사실이다. 다시 말해서 6개의 군집으로 군집화할 때 수렴 속도를 나타내는 반복횟수 역시 K-평균 군집화가 더 적음을 보여준다.

<표 3> 6개의 군집으로 처리한 목표함수

방법 / 변수	K-평균 군집화	개선된 K-평균 군집화
최소값	2655.66	2655.67
최대값	2678.35	2678.36
평균	2674.5341	2674.2695
분산	17.822	370.87*

<표 4> 6개의 군집으로 처리한 반복횟수

방법 / 변수	K-평균 군집화	개선된 K-평균 군집화
최소값	18	19
최대값	86	80
평균	36.74	37.81
분산	306.4	350.59

이상에서 본 것처럼 같은 군집화 결과를 얻는데 요구되는 반복계산 횟수는 초기치 계산을 통한 개선된 K-평균 방법이 적어도 7% 이상 개선효과가 있음을 알 수 있다. 또한 이러한 수렴 속도의 변화를 긍정적 효과라고 볼 수 있는 근거는 목표함수의 대표값이 두 가지 방법 모두 유사하다는 것이다.

6. 결론

군집화는 데이터 마이닝을 포함한 많은 분야에서 중요한 응용 기술의 하나이다. 텍스트 데이터의 군집화는 대량의 비정형 텍스트 문서 집합을 개념적으로 서로소인 집합으로 문서들을 그룹화하는 것을 말한다. 여기서는 개념벡터를 이용하여 문서들을

자동으로 군집화하는 방법을 제안하였다. 제안한 방법은 K-평균 군집화 방법이 가지는 초기 시작점 선택의 문제점을 개선하였고, 이를 통해 군집화 수렴속도를 높일 수 있음을 실험을 통해 확인하였다.

개선된 K-평균 군집화 방법을 Classic3라는 3,893개의 문서 집합에 적용하여 3개와 6개의 군집으로 각각 군집화할 때 목표함수가 임계치(오차) 범위 내에서 동일하였고, 반복횟수는 최대값이 각각 9.8%와 7.5% 줄어들었다. 다만 본 연구는 영문 문서에 대해서만 적용하였고, K-평균 군집화의 특성상 비 계층적 군집화를 대상으로 하였다는 한계점을 밝혀둔다. 따라서 앞으로 한글 문서 데이터의 효율적인 계층적 군집화 기법의 연구와 개발이 필요하다.

참고문헌

1. 김기영, 전명식 (1999). 다변량 통계자료분석, 자유아카데미.
2. Berry, M. W., Dumais, S. T. and O'Brien, G, W. (1995). Using Linear Algebra for Intelligent Information Retrieval, SIAM Review, 37(4), 573-595.
3. Bradley, P. S. and Fayyad, U. M. (1998). Refining Initial Points for K-Means Clustering, Proc. 15th International Conference on Machine Learning, 91-95.
4. Dhillon, I. S. and Modha, D. S. (2000). Concept Decompositions for Large Sparse Text Data using Clustering, IBM Research Report RJ 10147.
5. Dhillon, I. S., Fan, J. and Guan, Y. (2001). Efficient Clustering of Very Large Document Collections, Data Mining for Scientific and Engineering Applications, Kluwer Academic Pub.
6. Duff, I. S., Grimes, R. G. and Lewis, J. G. (1992). User's Guide for the Harwell-Boeing Sparse Matrix Collection, Boeing Computer Technical Report, 86-92.
7. Kolda, T.G. (1997). Limited-Memory Matrix Methods with Applications, PhD Thesis, The Applied Mathematics Program, University of Maryland.
8. Papadimitriou, C. H., Raghavan, P., Tamaki, H. and Vempala, S. (1998). Latent Semantic Indexing: A Probabilistic Analysis, Proc. 7-th ACM-SIGACT-SIGMOD-SIGART Symposium, 159-168.

[2003년 8월 접수, 2003년 11월 채택]