클러스터 중심 결정 방법을 개선한 K-Means 알고리즘의 구현

이 신 원[†] · 오 형 진[††] · 안 동 언[†††] · 정 성 종[††††]

요 약

K-Means 알고리즘은 재배치 기법의 일종으로 K개의 초기 센트로이드를 중심으로 K개의 클러스터가 될 때까지 클러스터링을 반복하는 것이다. 알고리즘의 특성상 K-Means 알고리즘은 초기 클러스터 센트로이드(중심) 및 클러스터 중심을 결정하는 방법에 따라 다른 클러스터링 결과를 얻을 수 있다. 본 논문에서는 K-Means 알고리즘을 이용한 초기 클러스터 중심 및 클러스터 중심을 결정하는 방법을 개선한 변형 K-Means 알고리즘을 제안한다. 제안한 알고리즘의 평가를 위하여 SMART 시스템의 16가지 가중치 계산 방식을 이용하여 성능을 평가한 결과 변형 K-Means알고리즘이 K-Means 알고리즘보다 재현률과 F-Measure에서 20%이상 향상된 결과를 얻을 수 있었으며 특정 주제 아래 관련 문서가 할당되는 클러스터링 성능이 우수함을 알 수 있었다.

An Implementation of K-Means Algorithm Improving Cluster Centroids Decision Methodologies

Shin-Won Lee[†] · Hyung-Jin Oh[††] · Dong-Un An[†††] · Seong-Jong Jeong[††††]

ABSTRACT

K-Means algorithm is a non-hierarchical (plat) and reassignment techniques and iterates algorithm steps on the basis of K cluster centroids until the clustering results converge into K clusters. In its nature, K-Means algorithm has characteristics which make different results depending on the initial and new centroids. In this paper, we propose the modified K-Means algorithm which improves the initial and new centroids decision methodologies. By evaluating the performance of two algorithms using the 16 weighting scheme of SMART system, the modified algorithm showed 20% better results on recall and F-measure than those of K-Means algorithm, and the document clustering results are quite improved.

키워드 : 문서 클러스터링(Document Clustering), K-Means 알고리즘(K-Means Algorithm), 클러스터 중심(Cluster Centroids)

1. 서 론

웹문서의 양이 급격히 넘쳐나는 오늘날에는 기존의 정보 검색 시스템에서 사용자의 질의에 대한 긴 검색결과 목록으로는 정보를 선별하고 유용한 지식을 획득하기에는 많은 시간과 노력을 필요로 한다. 따라서 사용자의 요구에 적합한 검색결과를 가공한 후 결과들 사이의 관계 및 생각지도 못한 유용한 지식을 획득하는 문서 클러스터링 기법이 문제해결 방법의 한 방법으로 등장하였다. 문서 클러스터링은 다량의 문서를 유사한 문서들끼리 그룹화하여 특정 주제 아래 자동 분류하는 것으로써 사용자가 특정 정보에 대한 검색 요구를 하였을 때 모든 문서를 검색하는 대신 사용자의 요구와 가장 가까운 주제의 클러스터 내의 문서만을 검색함으로써 정보 탐색 시간을 절약할 수 있고, 검색의 효율을 향상시킬 수 있다[3, 4]. 문서 클러스터링 기법은 정보 검색 시스템의 전체 문서 집합을 오프라인에서 미리 클러스터링하여 질의를 요청할 때 해당 질의와 가장 유사한 클러스터에 대해서만 검색을 수행하는 클러스터링 기법과 질의 검색 결과를 온라인상에서 즉시 수행하는 클러스터링으로 나눌 수 있으며 본 논문에서는 후자 클러스터링에 대해 논의한다[9].

본 논문에서는 클러스터링의 성능에 영향을 미치는 클러스터링 중심 결정방법에 대해 논하며 논문의 구성은 다음과 같다. 2장에서는 관련연구를 살펴보고, 3장에서는 기존의 K-Means 알고리즘을 살펴보고, K-Means 알고리즘의 클러스터 중심 결정을 개선한 변형 K-Means 알고리즘을 제안하며, 4장에서는 실험결과와 분석을 기술한다. 마지막으로 5장

※ 본 연구는 한국과학재단 목적기초연구(R01-2003-000-11588-0) 지원으로 수행되었음.
† 정 회 원 : 전북대학교 전자정보공학부
†† 준 회 원 : 3SOFT Technical Consultant
††† 종신회원 : 전북대학교 전자정보공학부 교수
†††† 정 회 원 : 전북대학교 전자정보공학부 교수
논문접수 : 2004년 8월 20일, 심사완료 : 2004년 9월 24일

에서는 결론을 맺는다.

2. 관련 연구

문서 클러스터링은 정보검색의 효율성과 유효성을 증대시키기 위한 목적으로 사용한다. 대표적인 문서 클러스터링의 방법론은 클러스터링의 결과로 생성되는 그룹의 구조에 따라서 계층적 클러스터링(hierarchical clustering method)과 비계층적 클러스터링(non-hierarchical clustering method)으로 나눌 수 있는데, 각각의 방법론에 따라 여러 가지 구현 알고리즘이 있다.

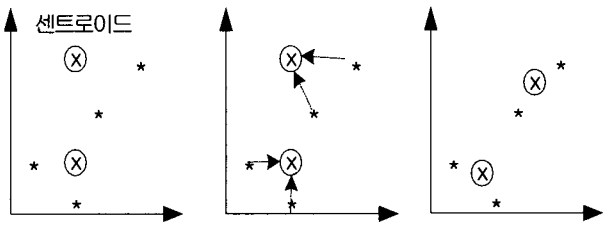
비계층적 클러스터링은 입력되는 문서의 순서에 따라 클러스터링 결과가 달라지는 단일 처리 방법(single pass method)과 이의 단점을 보완한 재배치 방법(reallocation method)이 있다. 계층적 클러스터링은 문서간의 유사도 정보를 토대로 단계적으로 계층적인 클러스터를 형성하는 방법으로 응집 알고리즘(agglomerative method)과 분할 알고리즘(divisive method)이 있다. 계층적 응집 알고리즘에는 단일 링크 방법(single link method), 완전 링크 방법(complete link method), 그룹 평균 연결 방법(group average link method) 등이 있다[8].

일반적으로 대규모 웹문서를 처리하는 클러스터링 알고리즘은 수백만 건을 처리하는 정보검색 시스템에 처리의 과부하를 주는 것을 피하고 동시에 메모리를 적게 사용해야 할 필요가 있다. 인공지능 분야에서 개발된 기존의 클러스터링 알고리즘들은 고차원의 대규모 데이터 집합으로 문서들을 벡터로 표현할 때 매우 고차원적이며, sparse한 특성을 나타내어 많은 메모리를 요구한다. 따라서 문서에서 중요한 내용은 포함하면서 중요하지 않은 부분을 제외시킨다면 클러스터링의 성능에 크게 영향을 미치지 않으면서 메모리의 요구사항을 줄일 수 있다는 장점을 가질 수 있다[9].

대규모 웹문서를 처리하기 위해 정보검색 시스템에서는 처리속도가 빠르며, 구현과 계산 복잡도를 요구하는 알고리즘을 사용하여 사용자의 검색 요구에 빠르게 응답해야 할 필요성이 있다. 본 논문에서 선택한 K-Means 알고리즘은 비계층적이며 재배치 기법을 사용하는 방법으로서 다양한 어플리케이션에서 사용하는 표준기술이며, 클러스터링 속도 측면에서 계층적 클러스터링 알고리즘보다 처리시간이 적게 걸리기 때문에 K-Means 알고리즘을 클러스터링 기법으로 사용하였다.

3. K-Means 알고리즘과 변형 K-Means 알고리즘

K-Means 알고리즘은 비계층적 클러스터링 기법으로 문서와 클러스터의 중심값을 나타내는 센트로이드(centroid)와의 유사도를 측정하여 문서를 적합한 클러스터에 재배치하는 기법이다. 여기에서 클러스터 센트로이드(중심)는 클러스터에 속하는 문서들의 평균 벡터값을 이용한다. 초기의 클러스터를 형성하고 (그림 1)과 같이 이를 계속적으로 정련하는 과정을 통해 최종의 클러스터를 형성한다. 이 기법은 초기 클러스터의 선택에 따라서 클러스터의 결과가 달라지며 특히 초기 클러스터 중심을 어떻게 선택하는가에 따라서 빠른 시간에 최적의 클러스터링 결과가 나오는 경우와 그렇지 않은 경우가 존재한다[10-13].



(그림 1) 클러스터 중심 생성 과정

3.1 K-Means 알고리즘

클러스터에 포함되어 있는 문서들의 특성을 나타내는 클러스터 중심(Cluster Centroid) 또는 클러스터 대표는 단어와 가중치 쌍으로 이루어진 벡터로 표현한다[9, 10]. K-Means 알고리즘은 특성상 생성된 센트로이드(클러스터 중심)에 따라 클러스터링 결과가 달라지며 특히 초기 센트로이드를 어떻게 선택하는가에 따라 빠른 시간에 최적의 클러스터링 결과가 나오는 경우와 그렇지 않은 경우가 존재하며, 클러스터링에 영향을 미치는 또 다른 요소는 클러스터링 과정에서 발생하는 새로운 클러스터 중심(Cluster Centroid)을 결정하는 것이다. K-Means 알고리즘에서는 클러스터에 속하는 문서들의 색인어와 가중치만을 단순히 하나의 클러스터 벡터로 병합하였으며 새로운 클러스터 중심은 다음의 식 (1)과 같다.

$$\overline{c_j} = \frac{1}{|\overline{c_j}|} \sum_{l=1}^{|c_j|} \overline{d_l} \qquad (1)$$

c_j : j^{th} 클러스터 벡터

d_l : l^{th} 클러스터에 할당된 문서 벡터

K-Means 알고리즘에서는 클러스터 중심을 L-차원의 공간에서 벡터($x_{i1}, x_{i2}, \cdots, x_{iL}$)로 표현하였을 때 클러스터에 속하는 문서를 대표하는 색인어와 가중치만을 단순히 하나의 클러스터 벡터로 머지(merge)한 것이다. <표 1>은 K-eans 알고리즘의 동작을 설명한 것이다[14].

〈표 1〉 K-Means 알고리즘

1. K값 클러스터 개수를 구한다.
2. K개의 초기 중심값(proto-centroids)을 구한다.
3. 각 문서(d)들과 중심값(c) 사이의 거리를 구한다.

$$dist(\overline{d_i}, \overline{c_j}) = \sqrt{\sum_{k=1} (d_{ki} - c_{kj})^2},$$

$i = 1,2,\cdots,n$ n: 전체문서개수
$j = 1,2,\cdots,K$ k: *centroid*의개수

4. 문서를 가장 짧은 거리의 중심값에 할당한다.

$$\arg\min dist(\overline{d_i}, \overline{c_j})$$
$$.i = 1,\cdots,n, j = 1,\cdots,k$$
$$d_i \in G_c, \quad \text{if } dist(\overline{d_i}, \overline{c_j}) < dist(\overline{d_i}, \overline{c_l})$$
$$(\text{for } all\ l = 1,2,\cdots,k\ l \neq j)$$

5. 새로운 클러스터 중심값을 재계산 한다.

$$\overline{c_j} = \frac{1}{|\overline{c_j}|} \sum_{l=1}^{|c_j|} \overline{d_l}$$

6. 이전의 중심값과 새로운 중심값을 비교하여 벡터간 차이가 거의 없을 때까지 반복한다.

If $\max\ \delta(\overline{c_j^{old}}, \overline{c_j^{new}}) < \theta$ *then return*
else goto 3

3.2 변형 K-Means 알고리즘

K번째 클러스터 중심을 결정하는 방법을 색인어와 가중치로 표현하는 임의의 한 개의 문서를 선택하는 K-Means 알고리즘과는 다르게 제안한 변형 K-Means 알고리즘에서는 초기 클러스터 중심을 결정할 때 문서를 3개(m = 3)로 선택하여 중복된 색인어를 제외하고 병합한 후 초기 클러스터 중심 벡터로 설정하였다. 변형한 초기 클러스터 중심은 다음의 식 (2)와 같다.

$$c_i^{initial} = \sum_{j}^{m=3} d_j \tag{2}$$

$c_i^{initial}$: i^{th} 클러스터 벡터
d_j : j^{th} 문서 벡터

클러스터링을 수행하는 과정에서 문서와 클러스터들 간의 거리는 하나의 문서만을 초기 클러스터로 설정할 때 문서 클러스터와 클러스터간 거리는 전체적으로 커지며, 최단 거리 문서-클러스터 거리판별에 영향을 주게 된다.

변형 K-Means 알고리즘에서 새로운 클러스터 중심 벡터는 클러스터에 포함된 모든 문서들이 갖는 색인어의 가중치의 평균으로 계산한다. 클러스터 중심 c_i와 문서 d_j가 병합되어서 생성된 클러스터 중심은 식 (3)과 같이 계산한다.

$$c_i^{new} = \frac{m_i \cdot c_i + m_{ij} \cdot d_{ij}}{m_i + m_{ij}} \tag{3}$$

c_i : i^{th} 클러스터 벡터
d_{ij} : i^{th} 클러스터에 할당된 j^{th} 문서 벡터
m_i : i^{th} 클러스터의 크기
m_{ij} : i^{th} 클러스터에 할당된 j^{th} 문서의 크기
c_i^{new} : i^{th} 새로운 클러스터 중심 벡터

변형 K-Means 알고리즘에서 생성된 클러스터 중심은 클러스터에 속하는 문서들이 클러스터 중심을 형성하는 과정에서 문서에 표현되어 있는 색인어들의 가중치들로 자신들의 특성을 반영하며 서로 이웃한 문서들에게 영향을 미치게 되어 문서간의 문맥을 고려한 클러스터링 효과를 얻을 수 있게 된다.

3.3 색인어와 가중치 계산 방법

색인어 가중치 부여는 문서와 문서를 비교하기 위해서 분류자질, 즉 단어에 적절한 가중치를 부여하는 방법이다. 일반적으로 모든 문서에 나타나는 단어는 문서를 구분할 수 있는 색인어에서 제외하며 소수의 문서에만 나타나는 단어를 색인어로 선택한다. 따라서 문서 내용을 설명하는데 같이 사용된 단어라 할지라도 다양한 비중을 가지고 있으며, 문서 내에서 색인어의 중요성에 대한 척도로서 문서의 각 단어에 대한 가중치를 부여한다. <표 2>는 색인어 SMART 시스템에서 사용하고 있는 다양한 가중치 계산 방법이다[10].

〈표 2〉 가중치 계산방법

약어	수 식	설 명
단어 빈도수(Term Frequency)		
B	1.0	Binary 0 or 1
N	tf	어휘 빈도수에 대한 정규화
A	$0.5 + \frac{tf}{\max\ tf}$	어휘 빈도수에 로그를 취함
L	$\ln tf + 1.0$	어휘 빈도수에 로그를 취함
문서 빈도수(Document Frequency)		
N	1.0	문서 빈도수가 영향 없음
T	$\ln \frac{N}{n}$	역문서 빈도수를 반영 N : 전체 문서 개수 n : 어휘가 나타난 문서의 개수
정규화(Normalization)		
N	1.0	정규화 하지 않음
C	$\frac{1}{\sum w_i^2}$	코싸인 정규화

단어의 가중치 계산에 주로 반영하는 요소는 어휘 빈도수(Term Frequency), 역 문서 빈도수(Inverse Document Frequency), 문서 길이에 대한 정규화 요소(Normalization)이다. TF는 문서 내에서 색인어의 출현 빈도수를 나타내며, IDF는 전체 문서 수에서 색인어가 나타나는 문서 수의 역수를 의미하며 정규화 요소는 문서의 길이를 조정하기 위한 것이다.

<표 2>로부터 가중치는 세 가지 요소에 대한 조합으로 BNN, BNC, BTN, BTC, NNN, NNC, NTN, NTC, ANN, ANC, ATN, ATC, LNN, LNC, LTC 조합이 가능하다. 본 논문에서는 각 문서를 대표하는 색인어에 대한 가중치를 로컬가중치, 글로벌 가중치, 정규화 요소로 표현하여 식 (4)와 같이 정의한다[1].

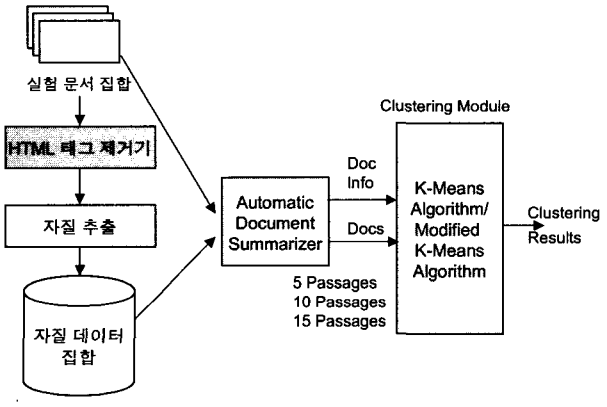
$$a_{ij} = l_{ij} g_i d_j \tag{4}$$

4. 실험 및 성능 평가

4.1 시스템 설계 및 구현

최근의 정보검색 시스템은 벡터를 계산할 때 소요되는 시스템 부하와 메모리 요구사항, 또한 실시간 처리를 고려하여 원문 전체를 사용하지 않고 자동으로 문서를 요약한 후 압축 문서를 사용하는 경향이다. 요약문서 작성은 전체 문서에서 문장별로 중요도를 계산하여 중요도가 높은 문장들을 뽑아내는 방법을 이용한다.

본 논문에서 구현한 문서 클러스터링 시스템은 아래의 (그림 2)와 같다[7]. 전체 시스템은 크게 자동문서 요약모듈과 클러스터링모듈로 구성되어 있다. 요약모듈은 HTML 태그 제거기, 자질 추출 모듈로 구성되어 있다. 실험에 사용한 문서집합인 Reuter21578 Newswire는 HTML 태그로 구성되어 있기 때문에 HTML 태그 제거기를 사용하여 각 문서들을 구분하는 시작태그만을 제외하고 HTML 태그를 제거한다. 자질 추출 모듈은 각 문서의 특징을 나타낼 수 있는 자질을 문서로부터 추출하는 부분으로써, 여기에서 자질은 문서 번호, 문서 길이(불용어를 제외한 문서 길이), 전체 문서 수, 평균 문서 길이 등 문서 정보(Document Information)와 문서내 색인어 위치(Term location), 색인어 번호(Term id), 문서내 색인어의 빈도수(Term Frequency)로 구성되어 있다. 각 문서의 특성 정보를 가지고 있는 자질 정보들은 자질 데이터 저장소에 저장된다. 실험문서의 전체집합과 자질 데이터 저장소에 저장된 문서 정보들은 자동 문서 요약기(Automatic Document Summarizer)를 통하여 문서 내에서 가장 중요한 문장을 선택하여 요약문을 생성한다.



(그림 2) 문서 클러스터링 시스템

문서 클러스터링 모듈은 문서요약기로부터 생성된 요약문(Summarized Document)과 문서 정보(Term ID, TF, DF)를 이용하여 문서간의 유사도를 기반으로 클러스터링을 수행하며, 각 문서내의 색인어의 가중치를 부여하는 weighting 부분과 클러스터링을 담당하는 K-Means 알고리즘 담당 모듈로 구성되어 있다.

4.2 실험 소개

본 논문은 정보검색 시스템 Condor Project의 결과물이며, 실험은 자동 문서 요약기를 통해 실험 문서 원문을 요약한 후, 요약문의 출력인 패시지(passage)(문장내에서 출현한 명사의 수)수를 각각 5, 10, 15 패시지로 변화시켜 가면서 K-Means 알고리즘과 변형 K-Means 알고리즘을 이용하는 클러스터링 모듈의 입력으로 하였다. 실험을 위하여 요약문의 색인어에 대하여 SMART 시스템에서 제안한 가중치를 계산하는 방법 16가지(BNN, BNC, BTN, BTC, NNN, NNC, NTN, NTC, ANN, ANC, ATN, ATC, LNN, LNC, LTN, LTC 등)를 적용하여 클러스터링 하였다.

4.3 실험 데이터

본 논문에서 사용한 실험문서는 Reuter21578 Newswire이다(http://www.research.att.com/~lewis/reuters21578)[15]. Reuter21578 Newswire 문서는 총 21578개의 문서로 구성되어 있으며, 정보 검색 시스템의 성능을 평가하기 위한 표준 문서 집합 중의 하나이다. Reuter21578 문서 컬렉션은 거의 모든 문서가 TOPIC 태그를 가지고 있으며, 이 태그는 문서의 내용을 파악하여 해당 주제에 속하는 문서임을 나타낸다.

선택한 실험 문서는 Reuter21578문서에서 가장 많이 존재하는 TOPIC 10개를 선정하였으며, 각 TOPIC당 문서 10개씩, 총 100개의 문서를 실험 문서로 선택하였다. <표 3>은 선택된 10개의 고빈도 TOPIC을 보여주고 있다. TOPIC 1번에 해당하는 문서번호는 1~10, TOPIC 2번에 해당하는 문서는 11~20, TOPIC 10번에 해당하는 문서는 91~100번 문서이며, 문서를 클러스터링하면 각 클러스터마다 정확률과 재현률을 계산할 수 있다.

〈표 3〉 Reuter21578 newswire의 고빈도 Topic

Topic 1	EARN	Topic 6	TRADE
Topic 2	ACQ	Topic 7	INTEREST
Topic 3	MONEY-FX	Topic 8	GNP
Topic 4	GRAIN	Topic 9	WHEAT
Topic 5	CRUIDE	Topic 10	SHIP

클러스터의 개수(K)는 실험데이터 수 100에 square root를 사용하여 10개의 클러스터로 선택하였다. 실험 문서는 자동 문서 요약기[2]를 사용하여 요약하였다. 자동 문서 요약기의 출력은 문서에서 중요한 문장 5라인씩 패시지 수를 변화시켜 가면서(5, 10, 15 패시지) 출력하였으며, 출력은 클러스터링 모듈의 입력으로 사용된다[2, 5].

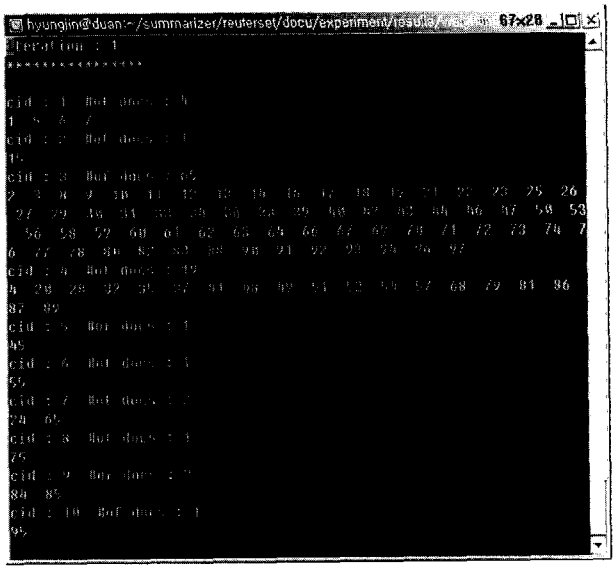
본 논문에서는 클러스터링을 수행할 때 문서 전문을 사용하지 않고, 자동문서 요약기를 사용한 이유는 클러스터링 속도 뿐만 아니라 실험 문서들의 특성 때문에 Topic에 해당하는 문서들의 색인어들이 특정한 주제아래 일관성을 갖추고

있음을 알 수 있으며 검색엔진 서버에 대한 부하를 줄일 수 있다는 장점을 갖게 되기 때문이다[6].

4.4 실험 결과 및 평가

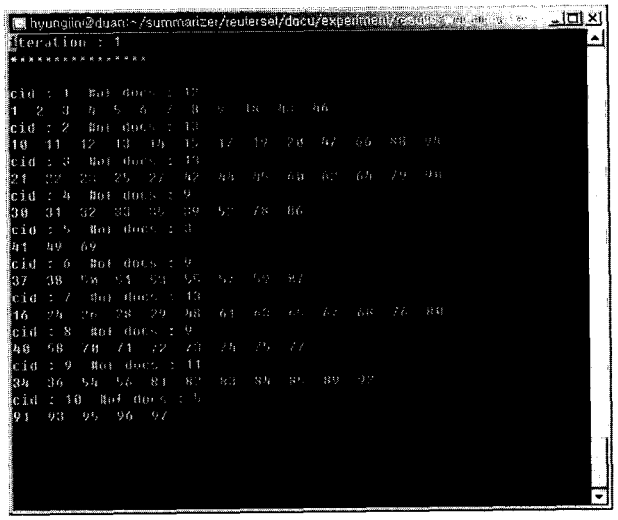
본 장에서는 K-Means 알고리즘과 제안하는 변형 K-eans 알고리즘의 성능을 비교 분석한다. 클러스터링 성능평가 척도는 클러스터링의 경우에는 생성된 클러스터가 어느 범주에 해당하는지, 또는 특정 문헌이 어느 범주로 자동 분류되었는지를 판정하기가 어렵지만 클러스터의 수인 K를 10개로 고정시켜, 동일한 환경에서 K-Means 알고리즘과 제안한 변형 K-Means 알고리즘의 성능을 상대적으로 평가하였다.

(그림 3)과 (그림 4)는 가중치 조합의 일부인 BNN을 사용하여 각각 K-Means 알고리즘과 변형 K-Means 알고리즘을 사용하여 100개의 문서를 클러스터링한 실험 결과이며, 이에 대한 정량적인 평가는 다음 절에서 논의하겠다. 각 그림은 할당된 문서의 수와 문서 번호를 함께 나타내고 있다. K-Means 알고리즘의 경우 하나의 클러스터에 많은 문서가 편중되어 있고 나머지 클러스터의 경우에도 할당된 문서가 너무 적어 각 클러스터의 주제를 파악하기가 힘들다. 반면, 변형 K-Means 알고리즘의 경우 각 클러스터에 할당된 문서가 균등하게 할당되어 있음을 발견할 수 있다. 각 그림에서 cid 1은 실험 데이터의 TOPIC 1번에 해당하는 문서가 많아 이 클러스터의 주제는 EARN이라 결정하였으며, cid 2는 TOPIC 2번인 ACQ, cid 10은 TOPIC 10번인 SHIP을 대표하는 주제로 판정한다.



(그림 3) K-Means 알고리즘의 BNN 적용 결과

실험 평가 방법은 정보 검색에서 보편적으로 측정하는 정확률과 재현률을 각각의 클러스터에 대하여 적용되었고 BNN, BNC, BTN, BTC, NNN, NNC, NTN, NTC, ANN, ANC, ATN, ATC, LNN, LNC, LTN, LTC 등 총 16가지 가중치 적용 방법에 대하여 평균 정확률과 평균 재현률을 정의한다. 전체적인 시스템의 성능을 평가하기 위하여 각 클러스터별 정확률과 재현률을 결합하고 평균을 취한 평균 정확률과 평균 재현률을 정의하였다. 또한 평균 재현률과 평균 정확률을 결합한 조화 평균 F-measure를 정의하여 재현률과 정확률을 하나의 척도로 나타내어 두 알고리즘의 성능을 그래프로 나타내어 본다.



(그림 4) 변형 K-Means 알고리즘의 BNN 적용 결과

각 클러스터의 정확률(Precision)은 아래의 식 (5)와 같다.

$$p=\frac{\text{해당 클러스터에 할당된 관련 문서 수}}{\text{해당 클러스터에 할당된 문서 수}} \tag{5}$$

각 클러스터의 재현률(Recall)은 아래의 식 (6)과 같다.

$$R=\frac{\text{해당 클러스터에 할당된 관련 문서 수}}{\text{해당 클러스터에 관련된 문서 수}} \tag{6}$$

클러스터링의 평균 정확률(Average Precision)은 아래의 식 (7)과 같다.

$$AP=\frac{1}{K}\sum_{k=1}^{K}P_k,\ K=10 \tag{7}$$

클러스터링의 평균 재현률(Average Recall)은 아래의 식 (8)과 같다.

$$AR=\frac{1}{K}\sum_{k=1}^{K}R_k,\ K=10 \tag{8}$$

F-Measure(AP와 AR의 조화 평균)는 아래의 식 (9)와 같다.

$$F=\frac{2\cdot AP\cdot AR}{AP+AR} \tag{9}$$

〈표 4〉 평균 정확률과 평균 재현률

(단위 : %)

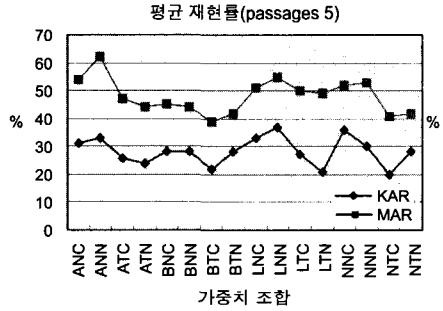
가중치 조합 \ 알고리즘	5 패시지				10 패시지				15 패시지			
	K	M-K	K	M-K	K	M-K	K	M-K	K	M-K	K	M-K
	AP	AP	AR	AR	AP	AP	AR	AR	AP	AP	AR	AR
BNN	72.2	52.6	28	44	72.8	60.1	23	56	80.5	50.6	15	48
BNC	73.8	54.3	28	45	39.7	50.7	19	53	49.1	49.1	14	48
BTN	72.2	51.0	28	42	69.4	60.1.	18	56	80.5	50.6	15	48
BTC	83.6	84.0	22	39	86.0	75.2	10	44	85.5	80.6	13	31
NNN	62.7	57.7	30	53	70.1	62.6	17	57	77.5	43.1	15	28
NNC	64.4	58.8	37	53	77.7	46.0	13	46	80.5	45.3	15	36
NTN	72.2	51	28	42	77.8	60.0	23	56	80.5	50.6	14	48
NTC	75.6	81.6	20	41	80.8	82.5	18	26	85.5	81.9	15	30
ANN	71.4	62.3	33	62	65.9	42.8	16	45	80.5	57.0	15	50
ANC	73.5	59.1	31	54	65.9	42.8	16	45	80.5	57.0	14	41
ATN	74.8	62.5	24	44	61.8	62.8	23	49	80.5	59.3	14	45
ATC	64.7	61.4	26	47	74.7	55.2	21	39	85.5	56.1	15	42
LNN	57.7	58.2	37	55	67.4	61.3	29	57	80.5	47.2	15	44
LNC	67.5	56	33	51	81.2	50.8	12	46	80.5	44.9	15	46
LTN	76.1	70.3	21	49	81.1	63.8	20	53	80.5	57.1	15	46
LTC	68.6	63.5	27	50	78.8	49.6	16	37	85.5	46.5	15	38

주) K는 K-Means 알고리즘, M-K는 변형 K-Means 알고리즘을 의미

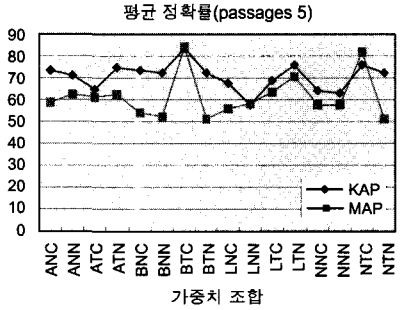
<표 4>와 (그림 5)에서 (그림 13)까지는 16가지 가중치 기법을 적용하였을 때의 클러스터링 결과를 나타낸 것으로 평균 정확률, 평균 재현률, F-Measure를 나타내고 있다. <표 5>는 초기 클러스터의 수를 변경한 경우와 클러스터 중심 계산 방법을 변경한 경우를 따로 고려하여 실험한 결과이다.

실험 결과 평균 정확률 측면에서는 K-Means 알고리즘이 보다 높게 나타났지만, 이것은 정확률을 계산할 때 클러스터에 할당된 문서가 매우 적고 하나의 클러스터에 편중되어 할당되었기 때문에 좋은 평가 척도라고 볼 수 없다. 평균 재현률과 F-Measure에서는 변형 K-Means 알고리즘이 20% 이상 좋은 성능을 보이고 있다. 재현률이 높다는 것은 특정한 주제 아래에 해당하는 문서가 제대로 할당된다는 것을 의미하며, 제안하는 기법이 특정한 주제 아래 문서가 할당되는 클러스터링 성능이 우수함을 알 수 있다.

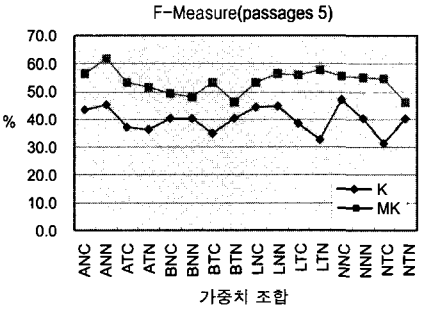
실험 데이터인 영문에서는 불용어 처리가 필수적이기 때문에 실험 결과로부터 요약문의 길이가 15 패시지인 경우 클러스터링 결과에 크게 영향을 미치지 않았으며, 요약문의 길이가 10 패시지인 경우보다는 5 패시지인 경우 원문의 내용을 크게 변형 시키지 않으면서도 원문을 압축한 결과를 얻을 수 있어 더 나은 클러스터링 결과를 얻을 수 있었다.



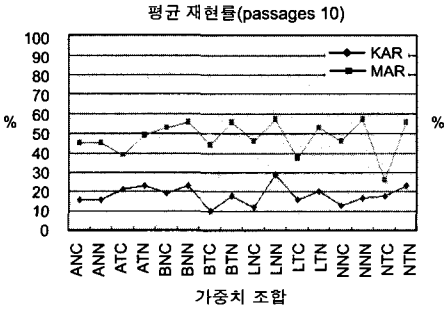
(그림 5) 5 패시지 평균 정확률



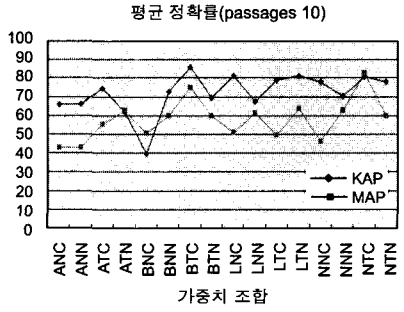
(그림 6) 5 패시지 평균 재현률



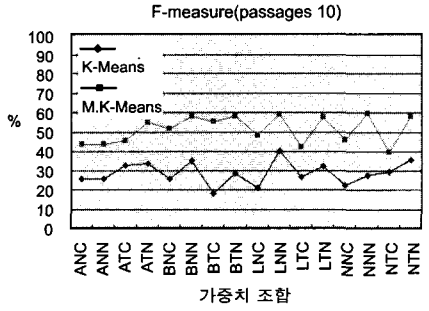
(그림 7) 5 패시지 F-Measure



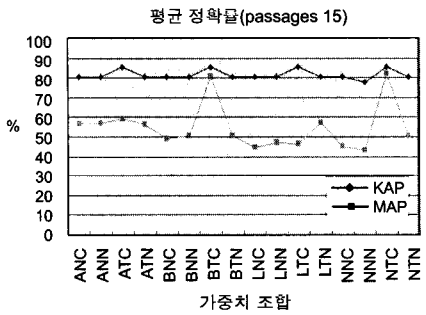
(그림 8) 10 패시지 평균 정확률



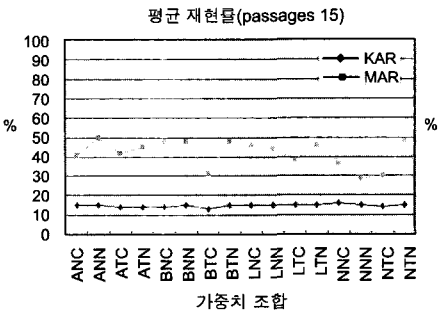
(그림 9) 10 패시지 평균 재현률



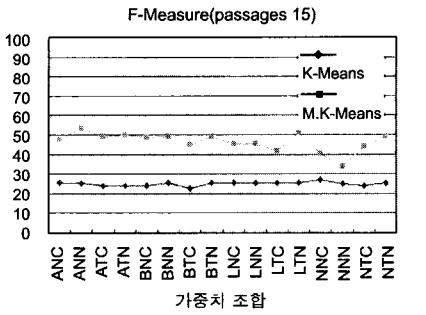
(그림 10) 10 패시지 F-Measure



(그림 11) 15 패시지 평균 정확률



(그림 12) 15 패시지 평균 재현률



(그림 13) 15 패시지 F-Measure

〈표 5〉 평균 정확률(AP)과 평균 재현률(AR)

(단위 : %)

알고리즘 / 가중치 조합	5 패시지							
	K	초기	중심	M-K	K	초기	중심	M-K
	AP	AP	AP	AP	AR	AR	AR	AR
BNN	72.2	79.5	68.5	52.6	28	42	27	44
BNC	73.8	80.9	67.9	54.3	28	42	26	45
BTN	72.2	86.3	76.3	51	28	37	27	42
BTC	83.6	83.6	75.5	84	22	38	23	39
NNN	62.7	69.4	53.2	57.7	30	50	38	53
NNC	64.4	72.3	58.7	58.8	37	46	40	53
NTN	72.2	77.9	82.8	51	28	39	22	42
NTC	75.6	82	75.7	81.6	20	40	20	41
ANN	71.4	74.1	65.5	62.3	33	52	45	62
ANC	73.5	77.9	63.3	59.1	31	45	40	54
ATN	74.8	82.4	69.7	62.5	24	39	32	44
ATC	64.7	73.8	67.4	61.4	26	38	30	47
LNN	57.7	68.6	50.3	58.2	37	50	39	55
LNC	67.5	72.1	49.6	56	33	46	33	51
LTN	76.1	75.7	69.7	70.3	21	39	30	49
LTC	68.6	71.5	57	63.5	27	45	27	50

주) K는 K-Means 알고리즘, 초기는 초기 클러스터 수 변경, 중심은 클러스터 중심 변경, M-K는 변형 K-Means 알고리즘을 의미

5. 결 론

본 논문에서는 문서 클러스터링 기법을 소개하고 재배치 기법의 일종인 K-Means 알고리즘의 초기 클러스터 선택과 새로운 클러스터 중심 결정을 변경한 변형 K-Means 알고리즘을 제안하였다. 제안한 변형 알고리즘은 초기 클러스터를 3개의 임의의 문서로 선택하여 색인어를 병합하였으며, 새로운 클러스터의 중심은 각 클러스터에 속하는 문서의 색인어 가중치를 평균한 것이다. 실험문서는 Reuter21578 news wire에서 고빈도 TOPIC에 해당하는 100개의 문서를 선택하였으며, 문서의 내용을 유지하면서 문서 크기를 줄이기 위하여 자동문서 요약기를 사용하여 요약문을 생성하였다. 제안한 변형 K-Means 알고리즘의 성능을 평가하기 위하여 각 문서의 색인어에 대해 총 16개의 가중치 기법을 적용하여 두 알고리즘을 갖춘 클러스터링 시스템에서 실험하였다. 실험 결과 요약문의 크기에 상관없이 평균 정확률 측면에서는 K-Means알고리즘이 높게 나타났지만 평균 재현률과 F-measure측면에서는 변형 K-Means 알고리즘이 20%이상 좋은 성능을 보이고 있고 초기 클러스터의 수를 변경하는 경우에는 정확률과 재현률에서 더 좋은 성능을 보이고 있음을 입증하였다.

참 고 문 헌

[1] 고지현, 오형진, 박순철, "LSI를 이용한 가중치 변화에 따른 클러스터링 결과 분석", 한국정보처리학회, 춘계학술발표논문집, pp.1009-1012, 2002.

[2] 김금영, 강인호, 안동언, 정성종, 박순철, "질의기반 자동문서 요약", 한국정보처리학회, 춘계학술발표논문집, pp.593-596, 2002.

[3] 김명철 외 공저, "최신 정보 검색론", 홍릉과학출판사, 2001.

[4] 김영택 외 공저, "자연언어처리", 생능출판사, 2001.

[5] 오형진, 고지현, 안동언, 정성종, "요약 문서 기반 문서 클러스터링", 한국정보처리학회, 춘계학술발표논문집, pp.589-592, 2002.

[6] 오형진, 변동률, 이신원, 박순철, 안동언, 정성종. "클러스터 중심 결정 방법에 따른 문서 클러스터링 성능 분석", 대한전자공학회, 하계학술대회, 2002.

[7] 오형진,"클러스터 중심 결정 방법을 개선한 변형 K-Means 알고리즘의 구현", 석사학위논문, 전북대학교, 2002.

[8] 이경순, "정보검색에서 벡터공간 검색과 클러스터 분석을 통한 문서 순위 결정 모델", 박사학위 논문, 한국과학기술원, 2001.

[9] 임영희, "후처리 웹문서 클러스터링 알고리즘", 정보처리학회 논문지B, pp.7-16, 한국정보처리학회, 2002.

[10] khaled Alsabti, Sanjay Ranka, Vineet Singh, "An Efficient K-Means Clustering Algorithm," IIPS 11th International Parallel Processing Symposium, 1998.

[11] Prabhakar Raghavans, Lecture Notes of Principles of Information Retrieval.

[12] Qin He, "A Review of Clustering Algorithms as Applied in IR," UIUCLIS--1999/6+IRG, 1999.

[13] Ray R. Larsons, Lecture Notes of Principles of Information Retrieval.

[14] Tapas Kanung, "The Analysis of a Simple k-Means Clustering Algorithm," Proc. of ACM Symposium on Computational Geometry Hong Kong, June, pp.12-14, 2000.

[15] http://www.research.att.com/~lewis/reuters21578.html.

이 신 원

e-mail : swlee9237@chonbuk.ac.kr
1990년 전북대학교 전산통계학과(이학사)
1992년 전북대학교 대학원 전산통계학과 (이학석사)
1994년 전북대학교 대학원 전자계산기공학과 수료
1995년~2004년 전북과학대학 컴퓨터과 재직
2004년~현재 전북대학교 시간강사
관심분야 : 정보검색, 한국어정보처리

오 형 진

e-mail : hyungjin@3soft.com
1999년 전북대학교 컴퓨터공학과(공학사)
2002년 전북대학교 대학원 컴퓨터 공학과 (공학석사)
2001년~2002년 미국 카네기멜론대학 언어기술연구소 방문연구
2003년~현재 3SOFT Technical Consultant
관심분야 : 검색엔진, 문서자동분류, 문서 군집, TDT 등

안 동 언

e-mail : duan@moak.chonbuk.ac.kr
1981년 한양대학교 전자공학과(공학사)
1987년 KAIST 전산학과(공학석사)
1995년 KAIST 전산학과(공학박사)
2001년~2002년 전북대학교 정보검색 시스템 연구센터 센터장
1995년~현재 전북대학교 전자정보공학부 부교수
관심분야 : 정보검색, 한국어정보처리, 문서분류, 문서요약

정 성 종

e-mail : sjchung@moak.chonbuk.ac.kr
1975년 한양대학교 전기공학과(공학사)
1981년 Houston대학교 전자공학과 (공학석사)
1988년 충남대학교 전산공학과 (공학박사)
1996년~1998년 전북대학교 전자계산소 소장
1985년~현재 전북대학교 전자정보공학부 교수
2001년~현재 전북대학교 BK21 전자정보사업단 단장
관심분야 : 정보검색, Grids