# 주제어 기반 문서 클러스터링 알고리즘

장성호[0], 강승식
국민대학교 컴퓨터학부, 첨단정보기술연구센터
{mpquake, sskang}@cs.kookmin.ac.kr

# Keyword-based Document Clustering Algorithm

Sung-Ho Jang, Seung-Shik Kang
School of Computer Science, Kookmin University and Advanced Information Technology Research Center

요 약

높은 연관성을 갖는 문서들을 서로 집단화시키는 문서 클러스터링은 문서와 문서간의 연관성을 확인할 수 있는 문서의 주제어 추출이 중요한 문제이며, 일반적인 정보검색 시스템에서 사용하는 출현빈도에 의한 주제어 추출은 성능 향상에 한계가 있다. 또한, 문서 클러스터링은 문서를 집단화시키기 위해 문서간 연관성을 확인하기 위해 유사도 계산에 따른 시간과 공간을 많이 소비하는 문제를 가지고 있다. 본 논문에서는 주제어 추출 기법을 적용하여 주제어 연관성에 의해 문서들을 집단화시키는 새로운 방법의 문서 클러스터링 알고리즘을 제안한다.

## 1. 서론

문서 클러스터링은 높은 연관성을 갖는 문서들을 낮은 연관성을 갖는 문서들과 분리해서 집단화하는 기법으로 문서와 문서간의 관련성을 확인하기 위해 문서간의 유사도를 계산하여 상호 관련성 여부를 확인한다[3,4,5,9]. 임의의 두 문서간의 유사도는 문서에서 출현한 용어들에 의해 결정되고, 용어들의 가중치는 그 문서를 특징짓는 문서 벡터를 구성한다.

이 때, 유사도 계산에 사용되는 기법으로는 Dice 상관계수, Jaccard 상관계수, 코사인 상관계수 등이 있다[2,6]. 이 계산 기법들에서 사용하는 용어의 가중치는 출현 빈도와 역문헌 빈도에 의해 계산되기 때문에 일상적으로 많이 사용되는 용어의 가중치가 높아지는 오류가 발생할 수 있다. 비주제어들이 유사도 계산에 사용됨으로 인하여 관련성이 적은 문서간의 유사도가 높아질 수 있기 때문이다. 따라서 단순히 출현빈도와 역문헌 빈도만으로 용어 가중치를 계산했을 때 유사도 계산에 의한 클러스터링의 성능 및 정확도를 향상시키는데 한계가 있다.

문서 클러스터링 기법들은 정보검색 시스템에서 전형적으로 나타나는 높은 차원을 가진 많은 데이터 집합을 클러스터하기 위해 많은 시간과 공간을 요구한다[6,7,11]. 이는 N개의 문서를 클러스터링하기 위해서 N개의 문서에 대해서 문서간 상호 관련성 여부를 확인하기 위해 반복적으로 유사도 계산을 수행해야 하고, 또한 계산 결과 저장을 위해 N×N의 유사도 행렬 공간이 필요하기 때문이다. 비록 N×N의 하삼각 행렬만 사용하더라도 기억 공간의 크기가 $O(N^2)$이 되므로 N이 충분히 클 경우 매우 많은 기억 공간이 필요하다. 때로는 문서 클러스터링에서 역파일 알고리즘이 유용하기는 하지만, 문서의 수가 많아질수록 유사도 계산에 필요한 시간과 유사도 행렬을 저장하는데 필요한 공간은 매우 크다[8,10].

본 논문에서는 문서를 대표할 수 있는 주제어를 추출하고 추출된 주제어를 이용하여 유사도 행렬을 사용하지 않는 문서 클러스터링 방법을 제안한다.

## 2. 주제어 추출

일반적으로 클러스터링을 위한 문서 벡터의 구성은 추출은 용어의 출현 빈도와 역문헌 빈도에 의존하고 있다. 그런데 문서에서 출현 빈도와 역문헌 빈도만으로 추출된 주제어는 문서 내용과는 관계없이 일상적으로 자주 사용되는 명사나 단순히 출현빈도만 높은 용어가 추출되는 오류가 발생할 수 있다. 유사도 계산 방식을 중심으로 한 클러스터링 기법은 불용어를 제외한 모든 용어들을 문서를 대표하는 용어로 간주하며, 단지 출현 빈도와 역문헌 빈도에 의해 그 가중치를 부여하여 문서 벡터를 구성한다.

따라서 기존의 방법에서는 주제어가 일치하지 않더라도 일상적인 용어들이 일치할 경우에 임계치보다 높은 값이 계산되면 하나의 클러스터에 속하는 현상이 발생하게 된다. 이러한 문제는 클러스터의 특징 벡터가 이를 대표하는 많은 용어들을 포괄하지 못하기 때문에 발생한다. 문서간 유사도 계산의 정확도를 높이기 위하여 본 논문에서는 문서의 내용을 대표하는 용어만을 추출하여 가중치를 계산하는 주제어 추출 방법을 사용한다[1].

일반적으로 어떤 문서의 주제어는 그 문서의 내용을 대표하는 용어들이며, 문서에서 주제어를 추출하려면 해당 문서의 내용을 분석하여 어떤 주제에 관한 문서인지를 파악해야 한다. 그러나 문서 유형에 따라 주제어 추출의 위치 및 대상이 달라질 수 있다. 학술 논문과 같이 '제목/요약/서론/결론'으로 구성된 문서는 제목과 요약에 출현한 용어가 주제어일 가능성이 높고, 신문기사의 경우는 주요 내용이 기사의 제목 및 본문의 앞 문장에 출현한 용어들이 주제어일 가능성이 매우 높다.

문서의 주제어를 추출하는데 사용될 수 있는 특성으로는 품사 정보와 격 정보 등 어절 단위의 용어 특성과 문장을 단위로 하는 용어의 구문론적 기능, 문서내에서 문장의 위치 및 역할에 의한 용어의 특성 등이 있다. 어절 단위의 용어 특성은 보통명사, 복합명사, 미등록어, 음절수, 조사 유형 등이다. 문장 단위의 구문론적 특성으로는 복합어 구성 여부, 주절/종속절의 주어/목적어 특성 등이 사용된다. 문서 단위의 문장 특성은 접속 부사 등 수사 어구에 의한 문장의 속성과 문장의 위치 정보 등이 활용된다.

문서의 주제어 추출을 위해 필요한 자연언어 분석 기법 중 구문 분석과 의미 분석은 정확도가 높지 않기 때문에 형태소 분석 결과만을 기반으로 용어가중치를 계산한 후 주제어를 추출한다. 문서에서 추출된 용어의 중요도는 '용어가 추출된 어절의 특성', '용어가 출현한 구/절 특성', '용어가 출현한 문장 특성'에 의해 계산된다.

각 용어의 가중치는 용어의 유형 및 조사 정보에 의해 '어

본 연구는 첨단정보기술 연구센터를 통하여 과학재단의 지원을 받았음.

절 단위' 가중치와 '문장 단위' 가중치를 부여한다. 어절 단위의 용어 가중치는 한글 문서의 경우 복합명사나 미등록 명사가 주제어일 가능성이 크기 때문에 복합명사의 용어 가중치를 기준으로 미등록어와 보통명사 등의 가중치를 경험적인 방법에 의해 부여한다. 문장 단위의 용어 가중치는 주절 혹은 종속절에 출현한 용어의 가중치를 조절하는 방법을 사용한다.

### 3. 주제어 기반 클러스터링 알고리즘

주제어 기반 클러스터링은 각 문서로부터 추출된 주제어 집합을 이용하여 클러스터를 생성한다. 클러스터링 알고리즘에 의해 최종적으로 생성되는 클러스터들의 집합을 $C$라 할 때 총 클러스터의 개수가 $n$이면 $C$는 순차적으로 생성되는 클러스터 $C_1$, $C_2$, ..., $C_n$ 으로 구성된다.

$$C = \{ C_1, C_2, ..., C_n \}$$

각 클러스터 $C_i$는 어느 클러스터에도 속하지 않는 문서 하나를 선택하여 초기화한다. 새로운 클러스터가 생성되면 초기 상태로부터 시작하여 클러스터가 안정화될 때까지 클러스터의 확장과 축소를 반복한다. 클러스터의 생성 과정에서 $i$-번째 클러스터 $C_i$의 초기 상태를 $C_i^0$ 라 하고 확장 혹은 축소에 의해 안정화되어 가는 과정에서 각 단계별로 클러스터 $C_i$의 상태를 구별하기 위해 $C_i$의 $j$-번째 상태를 $C_i^j$ 로 표현한다.

$C_i^j$ : $i$-번째 생성된 클러스터 $C_i$의 $j$-번째 상태

각 클러스터의 특성 벡터는 클러스터를 대표하는 주제어들의 가중치로 표현된다. 문서 $D$의 주제어 집합이 $K_D$이고 클러스터 $C_i$의 주제어 집합을 $K_{Ci}$라 할 때, 클러스터 $C_i$의 $j$-번째 상태인 $C_i^j$의 주제어 집합은 $K_{Ci}^j$로 표현된다.

문서 내용을 대표하는 주제어 집합으로부터 새로운 클러스터 $C_i$를 생성하는 알고리즘은 그림 1과 같다.

```
C_i^0 = { D };      // 임의의 문서 D
K_Ci^0 = K_D ;
C_i^1 = { D_x | for all k ∈ K_Ci^0, k ∈ K_Dx인 문서 D_x}
j = 1;
do {
  K_Ci^j = ⋃ K_Dx, where D_x ∈ C_i^j ;
  C_i^(j+1) = C_i^j ; // 클러스터 C_i의 next state
  for all D_x ∈ C_i^j begin
      s = sim(D_x, K_Ci^j) ;
      if ( s < 임계치 )
          C_i^(j+1) = C_i^(j+1) - {D_x}; // 문서 D_x 제거
  end_for
  j = j + 1;
} while (제거된 문서가 있음);
C_i = C_i^j ;       // 클러스터 C_i 생성
```

그림 1. 주제어 기반 클러스터링 알고리즘

① 클러스터 생성 및 초기화

첫 번째 단계인 '클러스터의 생성 및 초기화'는 클러스터가 결정되지 않은 임의의 문서 $D$를 선택하고, 클러스터 $C_i$의 초기 상태인 $C_i^0$에 문서 $D$를 포함시켜서 클러스터 $C_i$를 초기화한다.

$$C_i^0 = \{ D \}$$

이 때, 클러스터의 초기화에 사용된 문서 $D$를 '씨앗 문서'(또는 '초기 문서')라 한다. 씨앗 문서는 클러스터 $C_1$~$C_{i-1}$에 속하지 않은 문서들 중에서 무작위로 선택하거나 또는 첫 번째 문서를 선택한다.

문서 $D$의 주제어 집합 $K_D$를 문서 $D$에서 추출된 주제어 $k_1, k_2, ..., k_n$의 집합이다.

$K_D = \{ k \mid k$는 문서 $D$에서 추출된 주제어 $\}$

클러스터 $C_i$의 초기 상태 $C_i^0$의 주제어 집합 $K_{Ci}^0$은 아래와 같이 $K_D$로 초기화한다.

$$K_{Ci}^0 = K_D$$

② 클러스터 확장

클러스터 초기화 단계에서 클러스터 $C_i$의 초기 상태 $C_i^0$은 씨앗 문서 하나로 설정되고, 주제어 집합 $K_{Ci}^0$도 씨앗 문서의 주제어들로 초기화된다. '클러스터 확장' 단계에서는 씨앗 문서의 주제어들이 포함된 문서들을 씨앗 문서와 연관된 문서들로 간주하여 클러스터에 추가함으로써 초기 상태를 확장한다. 즉, 클러스터 $C_i$의 다음 상태 $C_i^1$는 $K_{Ci}^0$의 각 주제어(씨앗 문서에서 추출된 주제어)가 출현한 모든 문서를 $C_i^1$에 포함시켜 클러스터를 확장한다.

$C_i^1 = \{D_x \mid k \in K_{Ci}^0$인 주제어 $k$에 대해 $k \in K_{Dx}$인 문서 $D_x\}$

클러스터 확장은 1차 확장, 2차 확장, 3차 확장 기법을 사용할 수 있다. 1차 확장은 씨앗 문서에서 추출된 주제어에 의한 확장이고, 2차 확장은 1차 확장 결과로 클러스터에 추가된 문서의 주제어에 의한 확장이다. 3차 확장은 2차 확장에 의해 추가된 주제어에 의한 확장이다. 클러스터의 확장을 몇 차까지 할 것인지는 실험에 의해 결정한다.

클러스터의 확장에 따라 주제어 집합 $K_{Ci}^0$도 클러스터 $C_i^1$의 모든 문서들에 출현한 주제어들의 집합 $K_{Ci}^1$으로 확장된다. $C_i^j$의 주제어 집합은 $K_{Ci}^j$는 $C_i^j$의 모든 문서에 대한 주제어 집합의 합집합이다.

$$K_{Ci}^j = \bigcup K_{Dx}, \text{ where } D_x \in C_i^j$$

클러스터 $C_i^j$에 대한 주제어 집합 $K_{Ci}^j$은 해당 클러스터의 특징 벡터를 구하는데 사용된다. 클러스터의 특징 벡터는 주제어의 출현 빈도와 역문헌 빈도에 의해 계산된 주제어 가중치로 구성되며, 이 특징 벡터는 임의의 문서와 클러스터간의 유사도 계산에 사용된다.

③ 클러스터 축소-완성

이 단계는 확장된 클러스터 $C_i^1$에서 비관련 문서를 제거하여 확장된 클러스터의 크기를 축소하고 완전한 하나의 클러스터를 생성하는 단계이다. $j$ 값을 1로 시작하여 클러스터 $C_i^j$에 포함된 모든 문서들과 주제어 집합 $K_{Ci}^j$로 기술되는 $C_i^j$의 특징 벡터 사이의 유사도 계산에 의해 유사도가 낮은 비

관련 문서들을 $C_i^j$로부터 제외한다. 비관련 문서를 제거한 결과는 클러스터 $C_i^j$의 다음 상태인 $C_i^{j+1}$이 생성된다. 이 과정을 반복적으로 수행해서 계속 새로운 $C_i^{j+1}$와 $K_{Ci}^{j+1}$를 생성하면서 비관련 문서들을 제거해 나간다. 최종적으로, 비관련 문서가 모두 제거된 후에 남아있는 문서만으로 클러스터 $C_i$를 완성한다.

④ 클러스터링 종료

클러스터 $C_i$가 완성되었으면 동일한 방법으로 새로운 클러스터 $C_{i+1}$을 생성한다. 만약 새로운 클러스터 생성을 위해서 완성된 클러스터에 있는 문서들을 제외한 나머지 문서에서 새로운 임의의 문서 $D$를 선택하여 새로운 초기 클러스터 $C_{i+1}^0$와 새로운 초기 주제어 집합 $K_{Ci+1}^0$을 초기화한 후 클러스터의 확장과 축소 과정을 반복한다. 새로운 클러스터를 생성하기 위한 문서가 없는 경우, 즉 모든 문서가 클러스터에 포함되었으면 클러스터링을 종료한다.

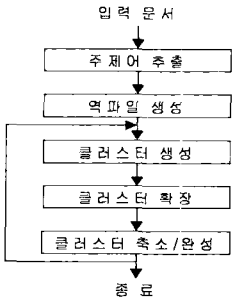


그림 2. 클러스터링 시스템의 구조

## 4. 설계 및 구현

주제어 기반 클러스터링 시스템의 구조는 그림 2와 같다. 입력 문서로부터 주제어를 추출하고 주제어가 출현한 문서와 주제어의 가중치를 역파일 형태로 구성한다. 각 주제어에 대한 역파일 구조는 초기 클러스터에 주제어가 출현한 문서들을 추가하여 클러스터를 확장하는데 적합한 구조이다.

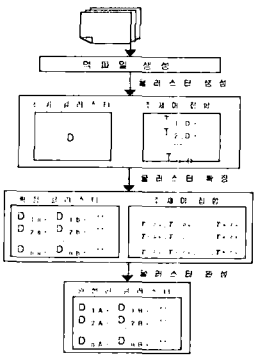

그림 3. 클러스터링 과정 예

그림 3은 클러스터의 초기화 및 확장/축소/완성 과정의 예를 보여 주고 있다. 클러스터의 초기화에 사용된 문서 $D$를 씨앗 문서로 하는 초기 클러스터를 생성하고, 초기 클러스터에 속한 씨앗 문서로부터 추출된 주제어 $T_{1,D}$, $T_{2,D}$, …, $T_{n,D}$로 초기 클러스터의 주제어 집합이 생성된다. 각 주제어가 출현한 문서들($D_{1a}$, $D_{1b}$, …, $D_{2a}$, $D_{2b}$, …, $D_{na}$, $D_{nb}$, …)을 초기 클러스터에 추가하여 클러스터를 확장한다.

확장된 클러스터에 추가된 문서들로부터 주제어를 추출하여 초기 주제어 집합을 확장하고, 확장된 주제어 집합과 클러스터에 포함된 문서들($D_{1a}$, $D_{1b}$, …, $D_{2a}$, $D_{2b}$, …, $D_{na}$, $D_{nb}$, …)간의 유사도 계산에 의해 비연관 문서를 제거함으로써 새로운 클러스터를 완성한다.

## 5. 결론

기존의 문서 클러스터링 알고리즘은 관련도가 높은 문서들을 군집화하기 위해 문서간 유사도를 기반으로 연관 문서를 판단하는 방법을 사용하고 있다. 출현 빈도와 역문헌 빈도에 의한 용어 가중치 계산 문제와 문서 클러스터링의 시간과 공간에 대한 문제를 해결하는 방법으로 주제어 기반의 문서 클러스터링 알고리즘을 제안하였다. 이 알고리즘은 문서간 유사도 대신에 공통 주제어를 포함하는 문서들을 선별하여 클러스터의 중심을 계산하고, 연관도가 낮은 문서를 제거하여 다시 클러스터의 중심을 계산하는 방법에 의해 클러스터링을 수행하는 방법이다.

이 방법은 유사도 대신에 공통 주제어를 기반으로 클러스터링을 수행하기 때문에 유사도 행렬을 구성하지 않아도 된다. 다만, 특정 주제어가 출현한 문서들을 추출할 수 있도록 역파일을 구성한다. 제안한 알고리즘에서 초기 클러스터를 생성하는 씨앗 문서의 선정 방법, 클러스터 확장 깊이, 클러스터에서 제외되는 문서의 유사도 기준에 따른 알고리즘의 효율성 및 정확도가 어떻게 변하는지는 향후 실험에 의해 평가되어야 할 부분이다.

## 참고문헌

[1] 강승식, 이하규, 손소현, 홍기채, 문병주, "조사 유형 및 복합명사 인식에 의한 용어 가중치 부여 기법", 한국정보과학회 가을 학술발표논문집, Vol.28, No.2, pp.196-198, 2001.
[2] 김영택, 자연언어처리, 생능출판사, 2001.
[3] Anderberg, M. R., "Cluster Analysis for Applications", New York: Academic, 1973.
[4] Can, F., and E. A. Ozkarahan, "Dynamic Cluster Maintenance", Information Processing & Management, Vol. 25, pp.275-291, 1989
[5] Dubes, R., and A. K. Jain, "Clustering Methodologies in Exploratory Data Analysis", Advances in Computers, Vol. 19, pp.113-227, 1980.
[6] Frakes, W. B. and R. Baeza-Yates, Information Retrieval, Prentice Hall, 1992.
[7] Murtagh, F., "Complexities of Hierarchic Clustering Algorithms: State of the Art", Computational Statistics Quarterly, Vol. 1, pp.101-113, 1984.
[8] Perry, S. A., and P. Willett, "A Review of the Use of Inverted Files for Best Match Searching in Information Retrieval Systems", Journal of Information Science, Vol. 6, pp.59-66, 1983.
[9] Sibson, R. "SLINK: an Optimally Efficient Algorithm for the Single-Link Cluster Method", Computer Journal, Vol. 16, pp.328-342, 1973.
[10] Willett, P., "Document Clustering Using an Inverted File Approach", Journal of Information Science, Vol. 2, pp.223-231, 1980.
[11] Willett, P., "Recent Trends in Hierarchic Document Clustering: A Critical Review", Information Processing and Management, Vol. 24, No.5, pp.577-597, 1988.