문서 시각화를 위한 개선된 클러스터링 알고리즘

신광철[0] 한상용
중앙대학교 컴퓨터 공학과
kcshin@archi.cse.cau.ac.kr[0] hansy@cau.ac.kr

Advanced Clustering Algorithm for Documents Visualization

Kwangcheol Shin[0] Sangyong Han
Dept. of Computer Science and Engineering, Chung-Ang Univ.

요 약

본 논문은 주어진 문서집합에 대한 유사도 검사를 통해 주어진 문서집합의 내용을 사용자가 직관적으로 파악할 수 있도록 하는 클러스터링 시각화 알고리즘에 관한 것이다. 제안하는 방법의 핵심은 주어진 문서 집합의 각 문서 사이의 유사도를 측정하여 각 문서 주변의 밀집도를 파악하고, 밀집도가 높은 문서들을 묶어 하나의 클러스터로 구성한 후, 구성된 각각의 클러스터의 키워드를 제공함으로 사용자가 해당 문서 집합의 내용을 보다 직관적으로 파악할 수 있도록 한 것이다. 우리는 TIME 데이터 집합에 대해 제시하는 알고리즘을 적용해 실험한 후 그 결과를 기존의 spherical k-means에 의해 클러스터링한 결과와 비교하여 제안하는 방법이 사용자에게 더 나은 시각화 정보를 제공함을 알아보았다.

1. 서 론

지식기반 사회가 되면서 넘쳐나는 정보를 선별하고 가공하여 유용한 지식을 획득하는 작업이 크게 부각되고 있다. 이러한 작업에는 색인, 검색, 필터링, 요약, 시각화 등이 포함되고, 이들은 모두 인간의 인식 행위의 기본이 되는 '분류'에 기반하고 있다. 지식 분류의 도구로서의 클러스터링은 1960년대 후반에 용어 분류와 문헌 분류에 대한 응용 연구로 시작되었고, 1990년대에 들어서면서 컴퓨터 처리 능력의 향상과 접근 가능한 정보의 폭증에 힘입어 클러스터링에 대한 관심이 부쩍 커지고 있다. 특히 최근에는 지식 분류와 시각화 분야에서 많은 응용이 시도되고 있다.

문헌 클러스터링에 대한 초기 연구는 클러스터 파일을 대상으로 검색 실험을 하기 위한 것이 대부분이었다[1]. 이후 유사한 환경에서 검색 성능을 향상시킬 목적으로 여러 클러스터링 알고리즘을 검토하는 연구가 이어졌다.[2]

1990년대에 들어와서 문헌 클러스터링은 검색 대상 문헌들을 클러스터 파일로 조직하기 위한 목적보다는 데이터베이스 또는 검색 결과의 브라우징이나 분류 자체를 목적으로 하는 연구들이 증가하였고[3], 검색 작업 이전에 데이터베이스의 내용을 시각적으로 보여 주기 위한 연구에서 더 발전하여 일단 검색한 결과를 시각적으로 보여 주거나 브라우징 할 수 있도록 하기 위한 연구로 발전하였다[4]. 또한 최근 들어 검색된 문헌들의 클러스터링을 통해 일차 검색 결과를 자동으로 정련하여 보여줌으로써 검색 성능을 향상시키기 위한 연구들이 나타나고 있다.[5]

본 논문은 이러한 클러스터링 기법의 발전 방향에서 주목받고 있는 주어진 어떤 문서집합에 대한 특성을 사용자가 시각적으로 보다 쉽게 내용을 파악(visualization)할 수 있도록 하는 새로운 클러스터링 알고리즘을 제안한다. 제안하는 알고리즘은 문서의 유사도를 기반으로 밀집도를 측정하여 높은 밀집도를 갖는 문서들을 문서 집합에서 추출하여 보다 정련된 클러스터 정보를 사용자에게 제공한다.

본 논문은 다음과 같이 구성되었다.

먼저 2장에서는 문서의 표현형태와 유사도를 측정하기 위한 개념벡터 응집도, 그리고 시각화를 위한 키워드에 대해 알아본다. 3장에서는 본 논문이 제시하는 알고리즘에 대해 설명하고 마지막 4장에서는 실험 결과와 결론에 대해 기술한다.

2. 문서의 표현 형태와 개념 벡터

2.1. 문서의 표현 형태

본 논문에서는 문서의 표현 형태로서 벡터 공간 모델을 이용하고자 한다. 벡터 공간 모델의 기본적인 아이디어는 각각의 문서를 가중치를 갖는 용어 빈도수의 벡터로 표현하는 것이다. 파싱(parsing)과 추출(extraction)의 전처리를 통해서 문서 i에 대해 용어 j의 빈도수 f_{ji}와 용어 j를 포함하는 문서의 수 d_j를 구한다.[6] 이와 같은 값을 이용하여, d차원의 공간 $R^d_{\geq 0}$상의 i번째 문서벡터 x의 j번째 원소를 다음 세가지 항의 곱으로 나타낼 수 있다.

$$x_{ji} = t_{ji} \cdot g_j \cdot s_i,\ 1 \leq j \leq d,\ 1 \leq i \leq n,$$
$$\textit{where } x \textit{ is a document vector}$$

여기서 t_{ji}는 용어 가중치 성분(term weighting component)으로서 그 값은 f_{ji}에 의해 결정되며, g_j는 전체 가중치 성분(global weighting component)으로서 d_j에 의해 결정된다. s_i는 x_i에 대한 정규화 성분(normalization component)이다. 직관적으로 t_{ji}는 용어의 상대적인 중요도를 의미하는 것이고 g_j는 한 단어의 전체 문서 집합에서의 전반적인 중요도을 의미하는 것임을 알 수 있다. 이와 같은 가중치 계산의 목표는 다양한 문서 벡터간의 구분을 확실하게 하여 더 나은 분류 효과를 얻는 것에 있다.[7]

위의 세 가지 성분을 선택하는 여러 가지 방안이 있으나[8], 본 논문에서는 대표적으로 많이 이용되는 정규화된 용어 빈도수(normalized term frequency)로 알려진 txn 법을 이용한다. 이 계산법은 $t_{ji}=f_{ji}$, $g_i=1$, $s_i = \left(\sum_{j=1}^{m} (t_{ji} g_j)^2 \right)^{-1/2}$ 로 하는 것이다. 주목할 점은 이러한 정규화가 $\|x_i\|=1$을 의미한다는 것이다. 즉, 각 문서 벡터는 $R^d_{\geq 0}$상의 단위 구(unit sphere)의 표면에 놓이게 되는 것을 의미하는 것이고, 이러한 정규화는 문서에 나타나는 용어의 방향성만을 유지하게 해주므로, 문서의 길이가 다르더라도 같은 주제를 다루는 문서들(즉, 유사한 용어들로 구성된 문서들)을 유사한 문서 벡터로 변환해 주는 효과가 있는 것이다.

2.2 개념벡터

벡터 공간 모델에 의해, 문서벡터는 d차원의 공간 $R^d_{\geq 0}$상의 $x_1, x_2, \cdots, x_n$로 표현될 수 있다[9]. 이 때 두 문서 벡터 x_i와 x_j사이의 코사인 유사도는 다음과 같은 두 벡터사이의 내적(inner product)으로 간단히 구할 수 있다.[6]

$$s(x_i, x_j) = x_i^T x_j = \|x_i\| \|x_j\| \cos(\theta(x_i x_j)) = \cos(\theta(x_i, x_j))$$

여기서 두 벡터사이의 각은 $0 \leq \theta(x_i, x_j) \leq \pi/2$ 이다.

$R^d_{\geq 0}$상의 n개의 문서 벡터들이 k개의 서로 다른 클러스터 $\pi_1, \pi_2, \cdots, \pi_k$로 나누어진다고 가정하면, $\{\pi_j\}_{j=1}^k$ 대해 π_j에 속한 문서들의 평균 벡터 혹은 중점 벡터는 다음과 같이 정의된다.

$$m_j = \frac{1}{n_j} \sum_{x \in \pi_j} x$$

여기서 n_j은 π_j에 속하는 문서벡터의 수이다. 이때 중점 벡터 m_j를 다음과 같이 단위 노름(norm)을 갖도록 정규화하면, 중점벡터의 방향성만을 갖는 개념벡터(concept vector) c_j를 정의할 수 있다.

$$c_j = \frac{m_j}{\|m\|}$$

위와 같이 정의된 개념 벡터 c_j는 다음과 같은 중요한 특성을 갖는다. d차원의 공간 $R^d_{\geq 0}$상의 임의의 단위 벡터 z에 대해 Cauchy-Schwarz 부등식이 성립한다.

$$\sum_{x \in \pi_j} x^T z \leq \sum_{x \in \pi_j} x^T c_j \qquad (2\text{-}1)$$

위의 식 (2-1)에 의해 개념 벡터 c_j는 클러스터 π_j에 속해있는 모든 문서 벡터에 대해 가장 근접한 코사인 유사도를 갖는 벡터임을 알 수 있다.

2.3. 응집도 측정과 키워드 추출

2.3.1 응집도 측정

우리는 앞에서 설명한 식 (2-1)을 통해 클러스터 π_j에 대해 클러스터의 응집도(coherence) 혹은 클러스터링의 질(quality)을 다음의 수식을 통해 측정할 수 있다.[9]

$$\sum_{x \in \pi_j} x^T c_j \qquad (2\text{-}2)$$

만약 하나의 클러스터에 있는 모든 문서의 벡터가 동일하다면 해당 클러스터의 평균 응집도는 1이라는 최고의 값이 될 것이다. 이것은 바꿔 말하면, 하나의 클러스터에 있는 문서벡터들이 매우 넓게 퍼져있다면 평균 응집도는 0에 가까운 값이 되는 것을 의미한다. 결과적으로 응집도가 높은 클러스터가 잘 만들어진 클러스터라 할 수 있다.

2.3.2 키워드 추출

클러스터링 수행 결과를 사용자에게 어떤 형태로 서비스 할 것인가는 클러스터링 과정 못지않게 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.

$R^d_{\geq 0}$상의 n개의 문서 벡터들이 k개의 서로 다른 클러스터 $\pi_1, \pi_2, \cdots, \pi_k$로 나누어진다고 가정하면, 문서 클러스터 π_j의 키워드는 용어 클러스터 $word_j$로 표현할 수 있다. 클러스터 π_j의 대표벡터인 개념벡터 c_j의 각 용어 중 다른 개념벡터에서의 가중치보다 큰 가중치를 갖는 용어는 용어 클러스터 $word_j$에 속하게 된다:[6]

$$word_j = \{kth\ word : 1 \leq k \leq d, c_{k,j} \geq c_{k,m}, 1 \leq m \leq c, m \neq j\}$$

이 때 d는 전체 용어의 개수이다. 이렇게 구성된 각 클러스터에 대한 용어 클러스터 $word_j$는 사용자로 하여금 해당 클러스터의 내용을 파악하는데 도움을 준다.

3. 제시하는 알고리즘

본 논문에서 제시하는 클러스터링 알고리즘은 주어진 문서 집합 내에서의 서로 유사도가 높은 문서들을 우선적으로 추출해 냄으로써 문서집합의 핵심이 되는 내용을 효과적으로 파악하는 것이다.

알고리즘에 대한 설명은 다음과 같다.

n개의 문서벡터가 있을 때, x_1부터 x_n까지의 문서를 차례대로 선택하여, 선택된 문서벡터 $x_i(1 \leq i \leq n)$와 나머지 문서 사이의 유사도를 측정하여 x_i에 대해 식 (3-1)을 만족하는 문서들의 개수 $k(x_i)$를 구한다. 여기서 ρ값은 일정 유사도를 나타내는 것으로서 응집도가 일정 값 이상인 클러스터를 형성하기 위해 사용된다.

$$s(x_i, x_j) \geq \rho, 0 \leq j \leq n, i \neq j, \quad \textit{whre } \rho \textit{ is a control parameter} \qquad (3\text{-}1)$$

각 벡터 상호간의 유사도가 식 (3-1)에 의해 검사가 이루어지면 각 문서 벡터 x_i중 다음의 조건을 만족하는 문서 벡터 x_i^*를 찾아 x_i^*중심으로 식 (3-1)을 만족하는 데이터들로 이루어진 클러스터를 형성한다. 여기서 σ는 클러스터에 포함되어야 하는 최소 데이터 수에 대한 값으로 클러스터가 일정 수 이상의 데이터를 갖도록 하기 위해 사용된다.

$$x_i^* = \{\arg \max_{i=1,\cdots,n} \{k(x_i)\} \geq \sigma\}, \quad \textit{where } \sigma \textit{ is a control parameter} \qquad (3\text{-}2)$$

x_i^*를 찾아 클러스터를 형성한 후, 클러스터에 포함된 문서 벡터들은 전체 문서 집합에서 제외한다. 클러스터에 포함되지 않은 문서 벡터들에 대해 x_i^*가 존재하지 않을 때까지 위의 과정을 반복한다. 위의 모든 과정이 끝나면 어떤 문서 벡터를 중심으로 해도 유사도가 ρ이상이면서 데이터 수가 σ이상인 클러스터를 만들 수 없는 문서벡터들이므로 남아 있는 문서 벡터들을

[표 1] 제안하는 알고리즘

1. 문서집합에 포함된 n개의 모든 문서를 L_2노름을 갖는 문서 벡터 $x_i(1 \leq i \leq n)$로 전환한다.
2. 하나의 문서벡터(x_i)를 선택하고, 선택된 문서와 나머지 문서 사이의 유사도를 측정하여 일정 수치(ρ)를 넘는 것의 개수 $k(x_i)$를 구한다.
   $s(x_i, x_j) \geq \rho, 0 \leq j \leq n, i \neq j,$ *whre* ρ *is a control parameter*
3. 문서집합에 포함된 모든 문서벡터에 대해 2의 과정을 반복한다.
4. 다음 식을 만족하는 x_i^*를 구해 x_i^*를 중심으로 유사도가 ρ이상인 문서벡터들을 하나의 클러스터를 형성하고, 클러스터에 포함된 문서벡터들은 전체 문서 집합에서 제외한다.
   $x_i^* = \{\arg \max_{i=1,\cdots,n} \{k(x_i)\} \geq \sigma\},$ *where* σ *is a control parameter*
5. x_i^*가 존재하지 않을 때까지 2에서 4까지의 과정을 반복한다.
6. 2에서 5까지의 과정이 끝난 후 남아 있는 문서 벡터들을 하나의 클러스터로 형성한다.

묶어 하나의 클러스터로 형성한다.
이로써 일정 응집도 이상을 갖는 클러스터를 해당 문서집합으로부터 추출해 낼 수 있고, 추출한 클러스터의 키워드를 이용해 사용자는 직관적으로 문서집합의 내용을 쉽게 파악해 낼 수 있다.
[표 1]은 제시하는 알고리즘의 의사 코드이다.
다음 장에서는 실험을 통한 결과를 토대로 제시하는 알고리즘의 효용에 대해 설명한다.

4. 실험 결과 및 결론

본 논문에서 제시하는 알고리즘의 효용성을 측정하기 위해 TIME지에서 추출한 423개의 문서[11]를 대상으로 실험했다. 먼저 실험을 위해 423개의 문서 데이터를 대규모의 문서집합으로부터 자동으로 문서 벡터를 형성해 주는 MC 프로그램[10]을 이용하여 벡터화한다. 이 때 불용어와 0.5%이하의 low-frequency, 15%이상의 high-frequency를 갖는 단어를 제거한다[9]. 벡터화된 문서집합을 대상으로 평균 응집도가 0.42이상이면서 클러스터에 포함된 데이터수가 20개 이상인 문서들을 추출하면 5개의 클러스터가 형성되고, 어느 클러스터에도 포함되지 못한 문서들을 하나로 묶어 마지막 클러스터를 형성한다.[표 2]
실험결과를 비교하기 위해 고차원의 문서벡터를 클러스터링 하는 spherical k-means 알고리즘[9]을 구현하여 실험하였다[표 3]. 표의 첫째 항은 클러스터 번호이고 둘째 항은 해당 클러스터에 포함된 데이터 수이며, 셋째 항은 해당 클러스터의 평균 응집도를 나타낸 것이다. 그리고 넷째 항은 해당 클러스터의 키워드 중에서 가중치가 높은 순으로 10개를 표시한 것이다. 제시하는 방법으로 클러스터링을 수행하였을 경우 생성되는 6개의 클러스터 중 마지막에 생성된 한 개의 클러스터를 제외한 5개의 클러스터가 0.42 이상의 응집도를 유지하면서 20개 이상의 데이터를 포함하고 있음을 알 수 있고, 클러스터의 응집도가 높기 때문에 추출되는 키워드 또한 내용을 쉽게 예측할 수 있는 일관성 있는 단어들로 구성됨을 알 수 있다[표 2]. 반면 [표 3]이 나타내는 바와 같이 spherical k-means를 적용하였을 경우 클러스터의 응집도와 포함 데이터 수를 일정 수준 이상을 갖도록 조절할 수 없고, 제공하는 키워드도 그 의미상 제안하는 방법에 비해 모호한 것을 알 수 있다.
위의 실험을 통해 제시하는 방법에 의해 클러스터링을 할 경우 일정 응집도를 이루는 클러스터를 효과적으로 만들 수 있을 뿐 아니라 키워드도 효과적으로 검출할 수 있음을 알아보았다. 그러나 제안하는 방법을 수행하기 위해 연산시간이 상대적으로 많이 소요되는 것은 앞으로 연구해야할 점이다.

참고 문헌

[1] Dattola, R.T. 1971. "Experiments with a fast algorithm for automatic classification." In G. Salton. (Ed.), The SMART Retrieval System: Experiments in Automatic Document Processing (pp.265-297). Englewood Cliffs, NJ: Prentice-Hall
[2] Burgin, R. 1995. "The retrieval effectiveness of five clustering algorithms as a function of indexing exhaustivity." Journal of the American Society for Information Science, 46(8): 562-572
[3] Cutting, D.R., Karger, D.R., and Pedersen, J.O. 1993. "Constant interaction-time scatter/gather browsing of very large document collections." In Proceedings of the Sixteenth Annual ACM SIGIR Conference on Research and Development in Information Retrieval (pp. 126-134)
[4] Lee, Kyung-Soon, Park, Young-Chan, and Choi, Key-Sun. 2001. "Re-ranking model based on document clusters." Information Processing & Management, 37(1): 1-14
[5] Amo, P., Ferreras, F.L., Cruz, F., and Rosa, M. 2000. "Smoothing functions for automatic relevance feedback in information retrieval." In Proceedings of the 11th International Workshop on Database and Expert Systems Applications (pp. 115-119)
[6] Frakes W. B. and Baeza-Yates R., "Information Retrieval : Data Structures and Algorithms." Prentince Hall, Englewood Cliffs, New Jersey, 1992.
[7] Salton G., and Buckly C., "Term-weighting approaches in automatic text retrieval", Information Processing & Management, 4(5):513:523, 1988.
[8] Kolda T. G. and O'Leary D. P., "A Semi-Discrete Matrix Decomposition for Latent Semantic Indexing in Information Retrieval," ACM Transactions on Information Systems, 16, 322-346. 1998.
[9] Dhillon I. S. and Modha, D. S. "Concept Decomposition for Large Sparse Text Data using Clustering," Technical Report RJ 10147(9502), IBM Almaden Research Center, 1999
[10] Dhillon I. S., Fan J., and Guan Y., "Efficient Clustering of Very Large Document Collections" Data Mining for Scientific and Engineering Applications, Kluwer Academic Publishers, 2001. Available at http://www.cs.utexas.edu/users/jfan/dm/
[11] Available at http://www.dcs.gla.ac.uk/idom/ir_resources/test_collections/

[표 2] 제안하는 알고리즘에 의한 실험 결과 (ρ=0.42, σ=20)

클러스터	데이터 수	평균 응집도	키워드
1	*55*	*0.422*	*gaulle, france, europe, french, market, west, common, nato, nuclear, charles*
2	*32*	*0.543*	*viet, nam, diem, south, saigon, cong, buddhist, war, buddhists, vietnamese*
3	*21*	*0.570*	*nasser, arab, baath, syria, egypt, iraq, army, unity, cairo, yemen*
4	*39*	*0.446*	*khrushchev, soviet, moscow, nikita, red, russia, communist, chinese, time, peking*
5	*25*	*0.493*	*party, minister, government, labor, britain, election, macmillan, prime, tory, tories*
6	256	0.245	years, year, man, africa, british, country, president, people, premier, world

[표 3] spherical k-means에 의한 클러스터링 결과 (k=6)

클러스터	데이터 수	평균 응집도	키워드
1	125	0.305	government, south, viet, nam, diem, war, general, white, man, troops
2	67	0.373	soviet, moscow, red, chinese, russia, china, communist, peking, nikita, russian
3	76	0.337	party, minister, cent, years, time, india, year, labor, election, socialists
4	109	0.319	gaulle, britain, west, france, europe, germany, british, german, world, market
5	29	0.481	nasser, arab, egypt, baath, syria, yemen, iraq, army, unity, jordan
6	17	0.361	french, president, debre, senghor, africa, olympio, premier, assembly, african, mali