Suffix Tree를 이용한 웹 문서 클러스터의 제목 생성 방법 성능 비교

염기종 권영식
(동국대학교 산업공학과)

Performance Comparison of Keyword Extraction Methods for Web Document Cluster using Suffix Tree Clustering

Ki Jong Youm Young S. Kwon
(Dongguk University)

요약

최근 들어 인터넷 기술의 발달로 웹 상에 많은 자료들이 산재해 있습니다. 사용자가 원하는 정보를 검색하기 위해서 키워드 검색을 이용하고 있는데 이러한 키워드 검색은 사용자들이 입력한 단편적인 정보에 바탕 하여 검색하고 검색된 결과들을 자체적인 기준으로 순위를 매겨 나열식으로 제시하고 있다. 이러한 경우 사용자들의 생각과는 다르게 결과가 제시될 수 있다. 따라서 사용자들의 검색 시간을 줄이고 편리하게 검색하기 위한 환경의 필요성이 높아지고 있다. 본 논문에서는 Suffix Tree 알고리즘을 사용하여 관련 있는 문서들을 분류하고 각각의 분류된 클러스터에 제목을 생성하기 위하여 문서 빈도수, 단어 빈도수와 역문서 빈도수, 카이 검정, 공통 정보 , 엔트로피 방법을 비교 평가하여 제목을 생성하는데 어떠한 방법이 가장 효과적인지 알아보기 위해 비교 평가해본 결과 문서빈도수가 TF-IDF보다 약 10%정도 성능이 좋은 결과를 보여주었다.

Key words : Suffix Tree, Feature selection, 제목 생성

1. 서론

1.1 연구의 배경

인터넷(Internet)의 보급과 관련기술의 발달로 전자 문서는 급속하게 증가를 보이고 있다. 웹 상의 정보들은 대부분 잘 정돈된 것이 아니라 일관성 없이 산재해 있는 경우가 많아 사용자들이 원하는 정보를 검색하기 위해서는 많은 시간을 투자해야 한다. 따라서, 많은 사용자들이 야후와, 엠파스와 같은 정보 검색 서비스를 제공하는 곳을 이용해서 정보를 찾고 있다[1].

현재 대부분의 정보 검색 서비스는 사용자들로부터 찾고자 하는 정보에 대한 단편적인 정보, 즉 찾고자 하는 정보를 나타내는 몇 개의 단어만을 입력받아 정보들을 조합하고 검색해서 사용자에서 제시하는 방법을 사용하고 있다.

사용자들에게 주어지는 정보검색 결과는 문서의 적합도에 따라 순위를 산출하고 사용자에게 나열식으로 보여주게 된다. 검색 결과 리스트는 단어의 빈도수만 고려해서 보여주는 것이기 때문에 사용자의 생각과는 다르게 그 결과가 주어지는 경우가 있고 사용자가 원하지 않는 다른 분야의 문서들이 포함되는 경우도 있다. 그러면 사용자들은 나열식으로 보여지는 결과들을 하나씩 확인해보며 자신에게 필요한 정보인지 판단을 해야한다.

이는 상당한 시간을 필요로 하게 된다. 그러므로 검색 결과를 제시할 때 사용자들이 원하는 결과를 쉽게 찾을 수 있도록 하는 연구가 이루어 져야 한다.

검색 결과를 사용자들이 파악하기 쉬운 형태로 제공하기 위한 노력으로 다양한 연구가 이루어지고 있다. 문서 클러스터링(Document clustering)이 그중 하나의 방법이다. 이는 검색 결과집합 내에서도 유사한 문서들과 유사하지 않은 문서들을 유사도를 계산하여 문서들의 유사 정도에 따라 그룹단위로 결과를 제시하는 방법이다. 이러한 방법은 사용자가 전체적인 검색 결과를 쉽게 파악할 수 있다는 장점을 가지고 있다. 그러나, 클러스터링 결과로 만들어진 각 클러스터(cluster)가 어떠한 주제로 묶였는지를 적절히 표현하지 못할 경우 사용자가 클러스터를 이해하는데 노력을 필요로 하는 문제점이 있다[2].

1.2 연구의 목적 및 의미

검색결과를 관련 있는 문서들을 그룹 지어 제시하고 적절한 제목을 생성하게 되면 사용자들이 원하

는 정보를 쉽게 찾을 수 있다. 일반적으로 문서를 분류하고 적절한 제목을 생성하기 위해 단어 빈도수와 역문서 빈도수(TF-IDF) 방법을 사용하여 제목을 생성하였다. 하지만 단어의 빈도수를 고려하게 되면 클러스터내 하나의 문서에 주제와는 다른 단어의 빈도가 높게 되면 그 단어에 영향을 받아 정확한 제목을 생성하는데 영향을 받을 수 있다.

본 논문에서는 정보 검색 결과로 만들어진 문서 집합들을 클러스터링 기법을 이용해서 유사도가 높은 문서들을 하나의 그룹으로 분류하고 그룹화를 통해서 사용자가 검색 결과를 쉽게 파악할 수 있도록 적절한 제목을 붙여 주기 위해 속성 추출 방법인 공통정보(Mutual Information), 엔트로피(Entropy), 카이-검정, 문서 빈도수(Document Frequency), TF-IDF 방법을 적용하고 각각의 방법을 비교 분석하여 가장 효과적으로 클러스터 제목을 생성해 줄 수 있는 방법을 선택하는 것이다.

유사 문서들을 그룹 지어 주는 클러스터링 방법으로는 어미 트리(suffix tree) 알고리즘을 사용하여 클러스터링 하려고 한다. 어미 트리는 기존의 검색 서비스가 단어 몇 개의 입력을 받아서 검색 결과를 제시해 주던 단편적인 방식을 단어간의 순서를 고려하여 문장 구조를 이용하여 분류하는 방법이다.

제 2 장 클러스터링 기법

2.1 문서 클러스터링

클러스터링은 산재되어 있는 여러 데이터들에 대한 특징을 찾아서 그 특징에 부합되도록 데이터들을 분류하는 작업을 의미한다. 데이터 마이닝(Data-Mining) 분야에서는 정보들의 배치를 어떻게 하느냐가 관건이기 때문에 데이터들을 분류하기보다는 군집화 하는 쪽이 더 가깝다. 그에 반해서 웹 상에 산재되어 있는 여러 특징을 가진 문서들을 분류하는 웹 에이전트(Web Agent)들의 역할에서는 클러스터링을 군집화 보다는 분류에 더 초점을 맞추어 사용되고 있다. 이 두 가지로 볼 때 클러스터링의 정의는 다음과 같이 내려 볼 수 있다[2].

"클러스터링은 사물을 분류할 수 있는 패턴들을 찾기 위하여 미리 알려져 있지 않은 사물들의 특징을 인식하는 것이다."

다시 말하면, 클러스터링이란 유사한 특성을 가지는 데이터들을 함께 묶어 이들 데이터가 가지고 있는 공통적인 특징을 그 군집의 대표로 나타내어 전체에 산재되어 있는 데이터를 몇 가지의 특성 군으로 나누어주는 것이다.

이러한 클러스터링의 정의에 따라서 웹 문서 분류에 사용되는 웹 문서 클러스터링이란 특정 문서 집합 내에 있는 각 문서들간의 유사도를 측정하여 유사한 문서들을 그룹 지어 주는 것을 말한다. 문서들간의 유사도는 각 문서가 갖는 특징들을 비교하여 계산하게 되는데 일반적으로 문서에 포함되어 있는 단어의 빈도수를 그 특징으로 사용한다. 문서 클러스터링 기법은 그 대상이 되는 집합에 따라 검색 전 클러스터링과 검색 결과 클러스터링으로 나누어 볼 수 있고, 그 방법에 따라 비계층적 클러스터링과 계층적 클러스터링으로 나누어 볼 수 있다[1].

2.1.1 검색전 클러스터링과 검색결과 클러스터링

검색 전 클러스터링이란, 웹 검색엔진의 문서 집합•전체를 대상으로 하여 수행되는 클러스터링으로 검색엔진의 성능을 향상시키고자 하는 목적으로 이루어지는 문서 클러스터링을 말한다. 이 경우 문서 클러스터링이 사용자의 검색요구가 있기 전에 미리 이루어질 수 있어, 사용자들이 클러스터링 결과가 반영된 검색 결과를 얻기까지 걸리는 시간을 줄일 수 있다는 장점을 가지고 있다. 그러나, 대상 집합 내 각 문서간의 유사도를 결정하는데 있어서 사용자 관심 밖의 문서를 많이 포함하게 되므로 사용자가 원하는 결과와는 다른 결과를 얻게 될 수 있다는 문제가 있다[1, 10].

검색 결과 클러스터링은 사용자가 검색엔진에 질의를 던진 결과로 얻게 되는 검색결과 문서들을 대상으로 하는 클러스터링을 말한다. 이 경우 사용자의 관심사항이 반영된 문서들을 대상으로 하여 비교가 이루어지므로 검색전 클러스터링에 비해 좋은 결과를 얻을 수 있다. 그러나, 사용자가 검색질의를 던진 후 문서 클러스터링을 통해 최종 결과를 얻기까지 사용자가 추가적인 시간이 소모될 수도 있다는 단점이 있다[1, 9].

2.1.2 비계층적 클러스터링(Non-hierarchical clustering)

비계층적 클러스터링은 임의로 선택된 초기 클러스터로부터 문서를 클러스터에 재배치하는 작업을 반복적으로 수행하여 최종 클러스터를 형성하는 방법으로, 계층적 클러스터링에 비해 시간은 빠르지만 검색 효율이 떨어지고 문서의 입력 순서에 따라 클러스터링 결과가 달라진다는 단점을 가지고 있다. 비계층적 클러스터링은 선형시간 알고리즘들을 주로 사용한다[9, 10].

비계층적 클러스터링 알고리즘에는 K-Means clustering, single-pass clustering, Buckshot and Fractionation이 있다.

K-Means clustering 알고리즘은 초기에 클러스터링 해야 할 중심들의 개수를 임의로 선택하여야 하고 반복적으로 클러스터의 중심값을 계산하면서 중심값의 변동이 없는 시점이 학습의 종료를 의미하고 최종적인 클러스터들을 얻는 방법이다. K-Means clustering 방법은 $O(nkT)$의 시간 복잡도를 가진다. 요구되는 클러스터의 수 k와 반복 회수 T에 대해 속도가 비례한다. 이러한 방법은 클러스터 중심의 초기 값에 따라 클러스터링이 수행된 결과의 수렴성이 달라지는 단점이 있다[1, 2].

single-pass clustering 방법은 클러스터링의 대상이 되는 각 문서를 하나의 클러스터에만 할당하면서 점진적으로 클러스터들을 만들어 가는 알고리즘이다. 처음에 하나의 문서를 선택하여 클러스터로 지정한

다. 새롭게 들어오는 모든 문서를 기존에 존재하는 모든 클러스터와 비교하여 가장 유사한 클러스터에 임계치를 만족하는 경우에는 해당 문서를 할당하고 그렇지 않은 경우는 새로운 클러스터로 생성하고 이런 과정을 모든 문서에 적용한다. 이때, 기존의 클러스터와 새로운 문서간의 유사도를 계산하기 위해서 중심값을 사용하게 되고 새로운 문서가 할당되면 클러스터의 중심값을 갱신하여야 한다. 생성되는 클러스터의 수 K에 비례하여 시간 복잡도 $O(nk)$를 가진다[1, 2].

Buckshot & Fractionation은 빠른 클러스터링이 이루어진다는 장점을 가지고 있지만 문서 입력에 따른 점진적인 처리가 이루어지지 않는다.

Fractionation은 지역적으로 한정된 영역에서 두 개의 가장 가까운 클러스터들을 찾는 알고리즘으로 계층적 클러스터링과 같이 임의의 정지 기준을 가진다. 검색 대상의 중심이 되는 문서들과 동떨어진 문서를 많이 포함하고 있는 도메인에서 낮은 성능을 보인다는 단점을 가지고 있다.

Buckshot은 K-Means 알고리즘의 일종으로 대상 문서집합에 있는 문서 샘플에 대해 계층적 클러스터링을 적용하여 초기 클러스터 중심들을 생성한다. 샘플링을 통해 초기 작업을 하기 때문에 샘플에 나타나 있지 않은 작은 클러스터들에 사용자의 관심이 집중되어 있을 경우 좋지 않은 결과를 얻게 된다는 단점이 있다[1, 2].

비계층적 클러스터링의 대부분은 단순히 단어들로 이루어진 집합으로 보고 클러스터링을 수행한다. 이 경우 문서 내의 유용한 정보인 단어들간의 인접정보를 잃게 된다. 따라서, 이러한 단어들간의 인접정보를 이용하여 클러스터링의 정확도를 향상시키기 위하여 구(pharse)를 이용한 클러스터링이 수행되어야 한다. 2.1.4절에서 살펴볼 어미 트리 클러스터링이 이러한 구를 사용하여 단어들간의 인접정보를 이용한 클러스터링 방법이다.

2.1.3 계층적 클러스터링(Hierarchical clustering)

계층적 클러스터링은 비계층적인 방법에 비해 클러스터링 시간이 느리지만 보다 정확한 클러스터링이 수행된다는 장점이 있다. 계층적 클러스터링은 이진 트리와 유사한 형태의 클러스터 구조를 생성해 내는 방법으로 클러스터 결과에 대한 서로 다른 레벨의 해상도를 제공할 수 있는 장점이 있다[8].

계층적 클러스터 방법은 클러스터와 문서들을 비교할 때 문서간의 유사도가 높은 것을 기준으로 하느냐 유사도가 낮을 것을 기준으로 하느냐에 따라서 단일 연결 클러스터링(single-link clustering)과 완전 연결 클러스터링(complete-link clustering)으로 나누어진다[2].

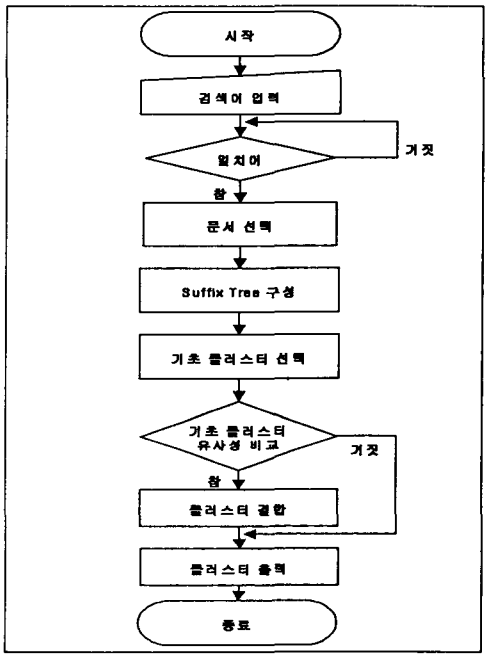
단일 연결 클러스터링은 이미 형성된 클러스터와 새로이 비교되는 문서간의 유사도를 계산하기 위해서 클러스터 안의 문서와 비교되는 문서간의 각각의 유사도 중에서 가장 큰 값을 선택하는 방법이다. 이 함수는 $O(n^2)$의 시간 복잡도를 가진다. 웹 문서를 대상으로 클러스터링을 수행할 경우 두 개의 대규모 클러스터와 여러 개의 아주 작은 클러스터로 나누어지는 경향이 있다.

완전 연결 클러스터링은 이미 형성된 클러스터와 새로이 비교되는 문서간의 유사도를 계산하기 위해서 클러스터 안의 문서와 비교되는 문서간의 각각의 유사도 중에서 가장 작은 값을 선택하는 방법이다. $O(n^3)$의 시간 복잡도를 가진다.

2.2 어미 트리 클러스터링(Suffix Tree Clustering: STC)

어미 트리 클러스터링(STC)은 클러스터링의 대상이 되는 문서들에 포함되어 있는 구들을 이용하여 공통구를 포함하는 문서들을 동일 클러스터에 할당하는 것을 기본으로 하고 있다. 어미 트리 클러스터링이 기존의 다른 클러스터링의 방식과 다른 점은 하나의 문서가 두 개 이상의 클러스터에 할당될 수 있는 구조를 가지고 있다. 그리고 필요한 클러스터의 수를 명시할 필요가 없고, 대신 기초 클러스터(base cluster)간의 유사도를 결정하기 위한 임계점(threshold)을 명시해 주어야 한다[1, 10].

[그림 1]은 어미 틀리 클러스터링의 단계를 순서도로 나타내어 보여주고 있다. 검색어를 입력받아 검색어에 부합하는 문서들을 선택하고 선택된 문서들을 대상으로 어미 트리를 구축하고 기초 클러스터를 찾아낸다. 마지막으로 찾아낸 기초 클러스터들의 유사도를 계산하여 유사도가 0.5 이상인 클러스터들은 합치게 된다.



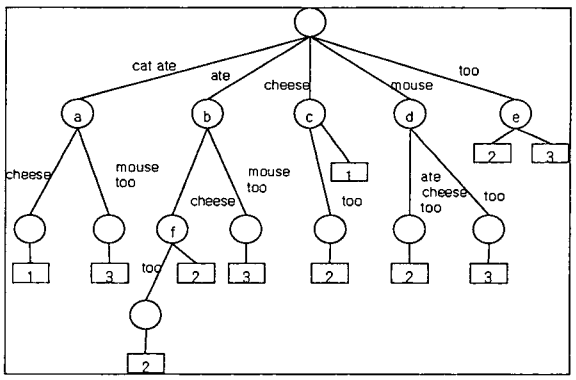
[그림 1] 어미 트리 순서도

어미 트리 클러스터링은 "문서처리", "기초 클러

스터 식별", "기초 클러스터 결합"의 세 단계로 이루어진다. 각 단계별 작업은 다음과 같다[1, 10].

① 문서 처리 단계(Document cleaning) : 형태소 분석과 불용어 제거를 통해서 각 문서를 표현하는 기본적인 문자열을 생성한다. 이 과정에서 단어들의 접미사는 제거되고, 복수형 단어는 단수형으로 변환된다. 또한 숫자나 HTML 태그, 구두점과 같은 비 단어 토큰들이 제거된다.

② 기초 클러스터 식별 단계(Identifying base cluster) : 문서집합의 구들에 대한 역인덱스를 생성하는 단계로서, 이를 위해 어미 트리를 이용한다. 다음 [그림 2]은 세 개의 문자열 "cat ate cheese", "mouse ate cheese too", "cat ate mouse too"에 대해 만들어지는 어미 트리의 구조를 도시화 한 것이다. 어미 트리는 문자열에 대한 모든 어미를 포함하는 컴팩트 트라이(compact trie)로써, 내부 노드는 적어도 두 개의 자식은 가지며, 루트(root)로부터 특정 노드에 이르는 경로에 있는 모든 간선의 라벨을 이어 붙이면 입력으로 주어진 문자열의 어미가 만들어진다.



[그림 2] 문자열 "cat ate cheese", "mouse ate cheese too", "cat ate mouse too"에 대한 어미 트리

클러스터 대상문서는 하나의 문자열로 간주되어 어미 트리의 입력으로 주어지게 되며, 그 결과 만들어진 어미 트리의 각 노드는 기초 클러스터(base cluster)가 된다.

③ 기초 클러스터 결합 단계(combining base clusters) : 어미 트리에 입력으로 주어진 문서들이 하나 이상의 구를 공유할 수 있기 때문에 클러스터가 중복되거나 동일하게 나타날 수 있다. 따라서, 유사한 클러스터들이 많이 생성되는 것을 피하기 위해 중복이 많이 일어나는 기초 클러스터들을 병합할 필요가 있다. 이를 위해 어미 트리 내의 모든 기초 클러스터들의 쌍에 대한 유사도를 정의하여 기초 클러스터의 유사도가 0.5보다 큰 경우에는 두 기초 클러스터를 병합한다.

어미 트리 클러스터링은 웹에서 검색결과 문서들을 실시간으로 해당 문서를 처리하여 그 결과를 어미 트리에 추가할 수 있고, 그 결과를 갱신하거나 새롭게 생성된 노드를 따로 기록함으로써, 관련된 기초 클러스터들만 갱신하고 그에 대한 유사도를 계산하는 방식을 이용할 수 있다[1, 9, 10].

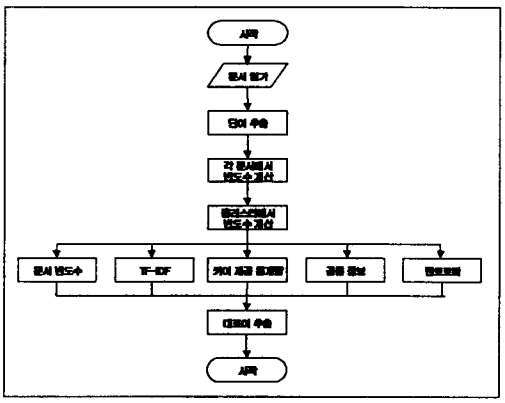
본 논문에서는 이러한 장점을 가지는 어미 트리 클러스터링을 기본 클러스터링 알고리즘으로 사용하여 웹 문서에 대한 검색 결과 집합에 대한 클러스터링을 수행하였다.

제 3 장 클러스터 제목 생성

검색 결과로 나온 문서들을 단순히 클러스터로만 구분을 한다면 사용자들은 각 클러스터들이 어떤 주제로 묶여 있는지 직접 확인해야 하는 노력을 기울여야 할 것이다. 사용자들의 노력을 최소화하기 위하여 분류된 각각의 클러스터들이 어떠한 공통 주제를 가지고 있는지를 알 수 있도록 각 클러스터들에 제목을 생성해 주어야 한다. 클러스터의 제목을 생성해 주기 위해서 클러스터 내에 있는 문서들의 공통된 정보를 이용하게 된다. TF-IDF 방법을 일반적으로 많이 사용하고 있다.

본 논문에서는 여러 가지 속성추출 방법 중에서 공통정보, 정보이익, 카이-검정, 문서 빈도수, TF-IDF로 분류된 클러스터의 제목을 생성하여 서로의 성능을 비교 평가하여 어떠한 속성추출 방법이 검색결과 웹 문서 클러스터의 제목을 가장 효과적으로 생성해 줄 수 있는지 평가하고자 한다[6].

[그림 3]은 분류된 클러스터의 문서들에서 텍스트 정보를 불러들여 공통정보, 정보이익, χ^2-통계량, 문서 빈도수, TF-IDF 각각의 방법들을 사용하여 대표어를 추출하는 방법에 대한 순서도 이다.



[그림 3] 대표어 추출 순서도

3.1 카이-검정

카이제곱 통계량은 여러 형태의 검정에 이용되는 검정통계량으로 적합하지만 특히 확률모형의 적합도 검정에 매우 유용하다. 카이제곱 통계량은 단어 t와 카테고리 c사이의 적합성을 측정한다. 만약 카이제곱 통계량이 큰 값을 가지게 되면 t와 c가 독립이라는 귀무가설을 기각하게 된다. 반대로 단어 t와 카테고리 c가 독립이라면 카이제곱 통계량이 0에 가까운 값을 가지게 된다. A는 c와 t가 동시에 발생한 수이고, B는 c없이 t만 발생한 수이다. C는 t없이 c만 발생한 수이다. D는 c와 t 모두에 대해서 발생하지 않

은 수이다, N은 문서의 총 수이다[6, 14].

단어-우수성 측정은 다음 식으로 정의할 수 있다[14].

$$\chi^2 = \frac{(AD-BC)^2 N}{(A+B)(A+C)(C+D)(B+D)} \quad (3.1)$$

따라서 식 (3.1)은 값이 크면 단어 t와 클래스 속성 C와 독립성을 기각하게 되므로 값이 클수록 단어가 클래스 속성과 연관이 있는 것을 의미한다.

3.2 공통 정보(Mutual information)

공통 정보는 목적 속성과 무관한 입력 속성을 제거하기 위해 많이 사용되어 왔다. 입력속성과 목적 속성과의 공통 정보 $I(X_i ; Y)$는 목적 속성 Y와 연관성 있는 입력 속성 부분집합 X_i을 선택하게 되면 $I(X_i ; Y)$ 측정값은 입력 속성 X_i의 지식으로 인한 Y의 불확실성 감소이다[6].

c를 목적 속성이라 하고 t를 하나의 단어라고 하면 A는 t와 c가 동시에 발생한 횟수, B는 c없이 t만 발생한 횟수, C는 t없이 c만 발생한 횟수, 그리고 N은 문서의 총 수를 말한다. 하나의 용어에 대한 관련성은 공통정보에 의해서 측정되어지며 식 (3.2)와 같이 정의되어 진다[6, 14].

$$MI(t, c) = \log \frac{p(t, c)}{p(t) \times p(c)} \quad (3.2)$$

식 (3.2)은 다음과 같이 추정될 수 있다[6].

$$MI(t, c) = \log \frac{A \times N}{(A+C) \times (A+B)} \quad (3.3)$$

공통정보 값이 커지게 되면 단어 t와 목적 속성 c 사이에 연관성이 커지게 된다. 목적 속성이 n개 값을 취하면 식 (3.4)를 이용하게 된다[6].

$$MI(t) = \sum_{i=1}^{n} p(c_i) MI(t, c_i) \quad (3.4)$$

3.3 엔트로피(Entropy)

엔트로피는 임의의 학습 셋에서 불순도를 측정하는 수단이 된다. 예를 들면 n개의 메시지가 각각 발생할 확률이 같다면 하나의 메시지가 발생할 확률(p)는 $p = \frac{1}{n}$이 된다. 이때 하나의 메시지를 구별하기 위해 필요한 정보량은 $-\log(p) = \log(n)$이 된다. 여기서 메시지를 구별하는데 필요한 정보량은 엔트로피라고 하는데 이것은 엔트로피가 클수록 얻을 수 있는 정보력이 적음을 의미한다[6, 14].

목적 속성값이 c개의 값을 갖는 일반적인 엔트로피는 식(3.5)와 같다[6].

$$H(s) \equiv -\sum_{i=1}^{c} -p_i \log_2 p_i \quad (3.5)$$

여기서 S는 학습 셋, p_i는 학습 셋 S에 속하는 클래스 i의 비율을 말한다. 따라서 목적 속성 값이 c개의 가능한 값을 취할 수 있기 때문에 엔트로피는 $\log_2 c$만큼 커질 수 있다.

3.4 문서 빈도 수(Document frequency)

문서 빈도수는 하나의 단어가 얼마나 많은 문서에 나타났는지를 말한다. 학습 셋에서 각각의 구별되는 단어에 대해서 문서의 빈도수를 계산한 후 정해 놓은 임계값보다 작은 값을 가지는 단어를 제거하게 된다. 문서 빈도수는 문서에 희박하게 나타나는 단어는 분류를 예측하는데 정보력을 가지고 있지 못하다는 기본적인 가정을 기반으로 이루어진다.

i번째 문서 $Dosc_i$에 단어 t가 존재한다면 1, t가 존재하지 않으면 값이 0이 되며 식 (3.6)과 같다[6].

$$DF(t) = \sum_{i=0}^{\text{문서수}} Dosc_i(t) \quad (3.6)$$

3.5 단어 빈도수와 역문서 빈도수 (TF-IDF)

단어 빈도수와 역문서 빈도수는 클러스터에 포함되어 있는 문서들에 나타난 용어들의 가중치를 이용하는 방안으로, 정보검색에서 가장 일반적으로 사용되는 방법이다[1, 3].

$$tf-idf_i = \frac{tf_i}{\max\ tf_i} \times \frac{af_i}{D} \quad (3.7)$$

N : 클러스터 내의 단어 개수
D : 클러스터 내 문서의 총 개
tf_i : i번째 단어의 클러스터 내 출현빈도
af_i : 클러스터 내에서 i번째 단어가 나타나는 문서의 수

분류된 클러스터 안의 각 단어의 빈도수와 각 단어가 나타나는 문서수를 계산하여 각 단어의 문서 내 중요도를 계산한다.

제 4 장 실험 방법

본 논문에서는 어미 트리를 이용한 웹 문서 분류와 각각의 분류된 문서들의 클러스터에 자동으로 제목을 생성하는데 사용되는 속성 추출 방법 중에서

공통정보, 엔트로피, 카이-검정, 문서 빈도수, TF-IDF를 이용하여 어느 방법이 제목을 자동으로 생성해주는데 효과적인지를 수행하여 비교하였다.

이 실험을 위해서 야후(http://kr.yahoo.com)의 웹 문서 300개를 대상으로 어미 트리 클러스터링을 수행하였다. 문서의 종류는 대학, 여행, 음악, 영화, 의료로 나누어진다.

웹 문서들을 대상으로 어미 트리 클러스터링을 수행하여 분류된 클러스터의 제목을 자동으로 생성하고 클러스터 제목의 자동 생성 방법을 서로 비교하여 어느 방법이 가장 효과적인지를 알아보기 위해 두 가지 실험을 수행하였다.

[실험 1] 어미 트리를 이용하여 분류된 웹 문서들의 클러스터에 대한 클러스터 제목 생성

[실험 2] "야후 디렉토리 사이트"에서 이미 분류되어 있는 문서집합에 대한 클러스터 제목 자동 생성을 통한 효과적인 방법 비교

4.1 실험 1

[실험 1]은 검색 결과 문서집합에 대해 어미 트리 클러스터링을 수행하여 분류된 문서들의 클러스터에 대해 클러스터 제목을 생성한 것이다. 이 실험을 위해 대학, 여행, 음악, 영화, 의료에 관한 문서 300개를 대상으로 어미 트리 클러스터링을 수행하였다.

[표 1]은 어미 트리 클러스터링의 결과로 만들어진 클러스터의 개수와 각 클러스터에 할당된 문서의 수를 보여준다. 어미 트리 클러스터링 기법은 하나의 클러스터에 하나의 문서만 할당하는 것이 아니라 하나의 문서가 두 개 이상의 클러스터에 할당될 수 있는 중복이 허용되는 특성 때문에 하나의 문서가 두 개 이상의 클러스터에 할당되는 것을 보여 주고 있다.

[표 1] 어미 트리 실행 결과

분류	생성된 클러스터의 수	클러스터에 할당된 문서 수	클러스터에 중복을 포함한 문서 수
대학	12	35	95
의료	9	23	40
영화	24	78	162
여행	20	55	89
음악	25	68	194

[표 1]과 같이 문서분류에 사용된 문서에 비해서 생성된 클러스터의 수가 많은 이유는 다른 분류방법과는 달리 어미 트리 클러스터링 방법은 단어의 순서, 즉 구를 이용해서 문서들을 분류하고 두 개 이상의 클러스터에 중복으로 할당되는 것을 허용하기 때문에 클러스터의 수가 많이 생성된다.

다시 말하면 같은 단어라도 문장의 위치와 인접 단어에 따라서 그 의미가 다를 수 있기 때문에 하나의 문서가 두 개 이상의 다른 클러스터에 분류가 될 수 있는 것이다. 이렇게 분류된 클러스터에 제목을 생성하기 위하여 다섯 가지 속성 추출 방법을 적용해본 결과, 생성된 각각의 클러스터들 중에 서로 유사한 제목을 가지고 있는 클러스터들을 볼 수 있었다.

[표 2]는 어미 트리를 이용해서 분류한 문서의 정확도를 측정한 것이다. 각각의 클러스터에 포함되어 있는 문서들이 클러스터가 나타내는 정보에 부합되는 문서가 정확히 분류된 것과 부정확하게 분류된 문서들의 비를 이용하였다. 정확도에 사용된 식은 다음과 같다.

$$\text{정확도} = \frac{\text{클러스터에\_정확히\_분류된\_문서.}}{\text{클러스터에 분류된 전체 문서}} \quad (4.1)$$

[표 2] 어미 트리를 이용한 문서 분류의 정확도

	정확히 분류된 문서(개)	오분류된 문서(개)	정확도 (%)
대학	65	30	0.684
의료	24	16	0.600
영화	108	54	0.667
여행	64	25	0.719
음악	128	66	0.660
전체	389	191	0.671

[실험 1]을 통해서는 문서 빈도수와 TF-IDF 방법이 어미 트리를 이용하여 분류한 클러스터의 제목을 대체로 잘 나타내 주고 있다는 것을 알 수 있었다. 하지만 분류하면서 하나의 문서가 중복으로 클러스터에 할당되는 특성 때문에 클러스터간의 유사성이 다소 높게 나오기 때문에 제목을 자동으로 생성하는 정확도를 정확히 평가하는데 한계가 있기 때문에 [실험 2]의 방법으로 다섯 가지 속성추출 방법을 이용하여 평가하였다.

4.2 실험 2

[실험 2]는 어미 트리를 이용하여 분류한 웹 문서들의 클러스터들에 대한 자동 제목 생성 방법 비교에 대한 성능만을 평가하기 위해 수행되었다. 이는 어미 트리를 이용하여 분류한 클러스터의 제목을 생성에 대한 평가가 객관적으로 이루어 질 수 없다고 판단되기 때문에 야후에서 이미 수작업으로 분류한 문서들을 대상으로 각각의 디렉토리를 하나의 클러스터로 간주하여 문서클러스터를 생성하고 야후의 제목과 각각의 속성 추출방법들의 정확도를 기준으로 비교 평가하였다.

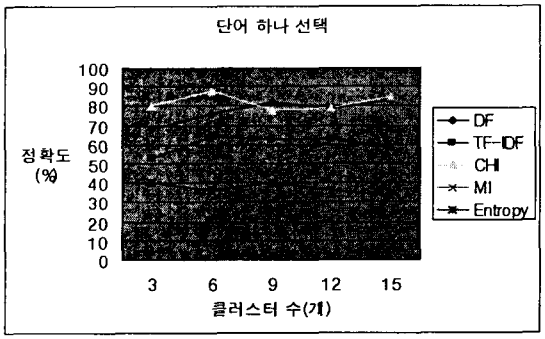
이 실험에 사용된 자료들은 대학, 의료, 음악, 여행, 영화의 5개 분야에 대한 웹 문서 300개 문서를 대상으로 5회 반복 실험하였다.

정확도 계산은 다음 식과 같이 계산하였다.

$$\text{정확도} = \frac{\text{일치하는\_단어\_수}}{\text{선택된 단어 수}} \quad (4.2)$$

각각의 실험은 단어 하나일 경우와 두 개, 그리고 다섯 개를 선택할 경우로 구분해서 실험하였다.

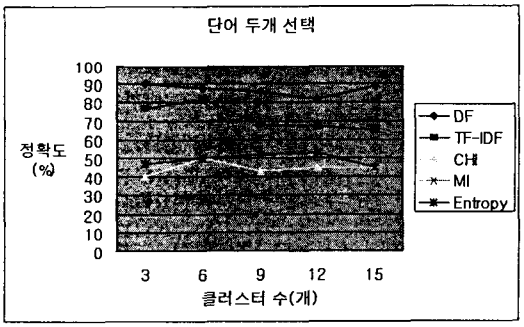
[그림 8]는 단어 하나만을 고려했을 경우의 속성 추출 방법들의 정확도를 클러스터의 수에 따른 변화를 보여주는 것이다. 대체적으로 카이-검정이 클러스터 수에 관계없이 전반적으로 좋은 성능을 보여주고 있다.



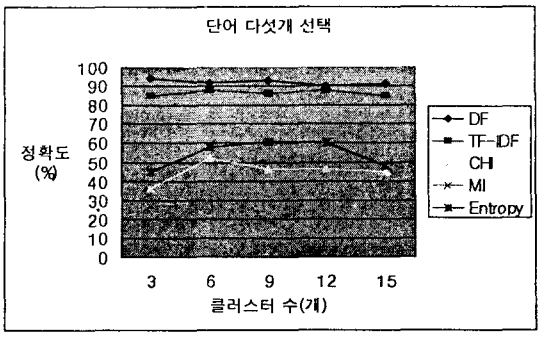
[그림 8] 단어 하나를 고려했을 때 제목생성

[그림 9]와 [그림 10]은 단어 두 개와 다섯 개를 고려했을 경우의 속성 추출 방법들을 비교한 것이다. 여기서는 단어 하나만 고려했을 경우와는 달리 문서 빈도수가 가장 좋은 성능을 보여주는 것을 알 수 있다.

이는 카이-검정의 경우에는 각각의 클래스의 특성을 나타낼 수 있고 다른 클래스와는 구별되는 단어들이 클러스터를 대표할 수 있는 단어들이 높은 값을 가지게 되는데 웹 문서의 제목들을 보면 클러스터 별로 구분할 수 있는 단어와 검색분야를 공통적으로 표현하는 단어가 같이 사용되고 있다. 이러한 웹 문서 제목의 특성 때문에 하나의 단어만을 고려했을 경우에는 카이-검정이 분류된 클러스터에 공통적으로 나타나는 단어들이 낮은 중요도를 갖기 때문에 가장 우수한 성능을 보여주고 있다. 하지만 두 개와 다섯 개의 단어를 고려했을 경우에는 문서 빈도수와 TF-IDF에 비해 정확도 측면에서 많은 차이를 보이는 것을 볼 수 있다.



[그림 9]단어 두개를 고려했을 때 제목생성



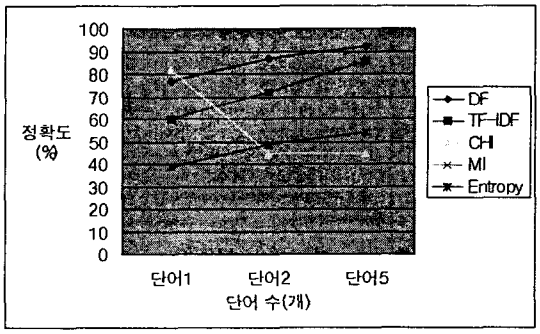
[그림 10] 단어 다섯 개를 고려했을 때 제목생성

엔트로피, 공통 정보는 희박한 단어에서 더 높은 값을 갖는 경향이 있기 때문에 클러스터를 대표할 수 있는 단어들이 제목으로 생성되는데 어려움이 있는 것을 볼 수 있다.

[표 4] 클러스터 제목 생성 정확도

(단위 : %)

	DF	TF-IDF	카이 검정	공통 정보	엔트로피
단어1	77	60	82	54	39
단어2	87	72	45	26	49
단어5	92	86	45	28	54



[그림 11] 단어 수에 따른 제목생성 정확도

[표 4]과 [그림 11]은 클러스터의 수에 따른 결과를 종합해서 각각 방법들의 평균을 보여주고 있다. 단어 하나를 고려했을 경우에는 카이-검정이 가장 좋은 성능을 보이고 있고 두 개 이상의 단어를 고려해서 웹 문서 클러스터의 제목을 생성할 때는 문서 빈도수가 클러스터의 수에 상관없이 가장 우수한 성능을 보이는 것을 볼 수 있다.

이전까지 일반적인 대표어 생성 방법에서는 TF-IDF를 이용하는 방법을 사용하였지만 [실험 2]의 결과에서는 문서 빈도수가 클러스터의 제목을 생성할 때 TF-IDF 보다 10%정도 성능이 향상되는 것을 볼 수 있다.

5장 결론 및 향후 연구

본 논문에서는 웹상에 산재해 있는 전자 문서들을 사용자들이 쉽게 원하는 정보를 찾을 수 있도록 검색결과를 단순히 순위를 매겨 나열식으로 제공하는 것이 아니라 같은 주제를 가지는 문서들을 어미 트리 클러스터링 방법을 사용하여 제목을 생성하는 방법을 비교 분석하여 가장 적합한 방법을 선택하는 연구를 하였다.

문서 분류에 사용된 어미 트리 클러스터링은 다른 클러스터링 방법들과는 달리 하나의 문서를 두 개 이상의 클러스터에 분류하는 특징을 가지고 있다. 이러한 특징은 하나의 웹 문서가 꼭 하나의 분류에만 속하는 것이 아니라 두 개 이상의 분류에도 속할 수 있는 특성을 잘 표현할 수 있었다.

어미 트리를 이용해서 분류한 클러스터의 제목을 자동으로 생성하는 방법으로는 속성 추출 방법 중에서 공통 정보, 카이-검정, 엔트로피, 문서 빈도수, TF-IDF를 사용해서 각각의 성능을 비교하였다.

이 다섯 가지 속성 추출 방법들을 비교 분석해본 결과 단어 하나를 선택했을 경우에는 카이-검정의 성능이 우수하였다. 그러나 두 개와 다섯 개 단어를 선택했을 경우에는 문서빈도수가 다른 네 가지 방법들 보다 우수한 성능을 보여주고 있다. 기계학습 분야에서 많이 쓰이는 엔트로피와 공통 정보는 희박한 단어에서 더 높은 값을 가지는 경향이 있기 때문에 클러스터의 제목을 생성에는 성능이 빈약하게 나타났다고 판단된다. 그리고 카이-검정의 경우에는 다른 클러스터들과 비교를 해서 그 클러스터의 특징을 나타낼 수 있는 단어들에 중요도를 높게 나타내 주는데 일반적으로 웹 문서 클러스터의 제목으로는 그 클러스터를 다른 클러스터와 차별화 하는 단어 하나와 검색된 문서들에서 골고루 나타나는 단어 하나를 조합해서 나타내 주기 때문에 한 단어만을 고려했을 때는 우수한 성능을 보여주고 있지만 두 개 이상의 단어를 선택했을 때는 성능이 많이 나빠지는 것을 볼 수 있었다.

보편적으로 웹 문서 클러스터의 경우 두 개 이상의 단어들을 제목으로 하고 있기 때문에 문서 빈도수가 일반적으로 사용되는 TF-IDF보다 10%정도 성능이 우수한 결과를 보여주고 있다.

참고 문헌

[1] 김태현, "계층구조를 이용한 문서 클러스터 제목의 자동생성 기법" , 석사학위논문, 2000

[2] 신진섭, "웹 문서 분류를 위한 단어의 연관성 모델과 클러스터링 모델", 박사학위논문, 2000.

[3] 윤보현, 강현규, 고형대, "자동 문서 클러스터링을 위한 디스크립터 추출 방안", 석사학위논문, 2000.

[4] 윤종식, "배깅과 부스팅을 이용한 나이브 베이지안 이메일 분류기의 성능 향상", 석사학위논문, 2001

[5] 장동현, " 문장 클러스터링을 통한 텍스트 자동요약에 관한 연구", 박사학위 논문, 2002

[6] 황호순, "프론트 엔드 e-CRM을 위한 전자메일 분류기 개발", 석사학위논문, 2001.

[7] Arne Anderson, N. Jesper Lasson, and Kurt Swanson. " Suffix trees on Words", In Combinational Pattern Matching, 1996.

[8] Oren Zamir, "Fast and Intuitive Clustering of Web Documents", Qual's Paper, University of Washington, 1997.

[9] Oren Zamir and Oren Etzioni, "Grouper : A dynamic Clustering Interface to Web Search Results", WWW8 Conference Refereed Papers, 1999.

[10] Oren Zamir and Oren Etzioni, "Web Document Clustering :A Feasible Demonstration", Proc.of ACM SIGIR'98

[11] Tom M. Mitchell, "Machine Learning", The McGraw-Hill Company, 1997.

[12] W.W.Cohen and Y.singer, "Context-sensitive learning methods for text categorization", J.ACM, 1999.

[13] Yiming Yang, "An evaluation of statistical approaches to text categorization", Journal of Information Retrieval, 1999.

[14] Yiming Yang and Jan O. Pedersen, "A comparative Study on Feature Selection in Text Categorization" ICML, 1997.

[15] Yiming Yang and X.Liu, "A re-examination of text categorization methods", In 17th annual International ACM SIGIR conference on Research and Development on Information Retrieval, 1999.

[16] http://kr.yahoo.com

[17] http://www.vivisimo.co.kr

[18] http://www.wisenut.co.kr