www.calsec.or.kr
http://dx.doi.org/10.7838/jsebs.2014.19.2.095

제품 특징화를 위한 오피니언 문서의 클러스터링 기법

An Opinion Document Clustering Technique for Product Characterization

장재영(Jae-Young Chang)*

초 록

오피니언 마이닝은 문서로부터 의견을 추출하는 텍스트 마이닝의 응용분야로 현재 활발한 연구가 진행되고 있다. 대부분의 관련 연구는 특정 제품군에 대해서 주어진 특징별로 긍정과 부정 평가를 나누는 감성분류에 초점을 맞추고 있다. 하지만 제품별로 강조되는 특성들을 구별해내는 연구는 거의 이루어지고 있지 않다. 본 논문에서는 특성별로 오피니언 문서들을 분류하고, 이를 이용하여 특정 제품군에 대해서 제품별로 강조되는 특성들을 선별하는 기법을 제안한다. 제안된 기법에서는 텍스트 클러스터링을 활용하였으며, 새로운 유사도 계산 방식을 사용하였다. 또한 실험을 통하여 제안된 방법의 유용성을 증명하였다.

ABSTRACT

Opinion Mining is one of the application domains of text mining which extracting opinions from documents, and much researches are currently underway. Most of related researches focused on the sentiment classification which classifies the documents into positive/negative opinions. However, there is a little interest in extracting the features characterizing the individual product. In this paper, we propose the technique classifying the opinion documents according to the product features, and selecting the those features characterizing each product. In the proposed method, we utilize the document clustering technique and develope a new algorithm for evaluating the similarity between documents. In addition, through experiments, we prove the usefulness of proposed method.

키워드 : 오피니언 마이닝, 클러스터링, 특성, 감성분류, 유사도
Opinion Mining, Clustering, Feature, Sentiment Classification, Similarity

---

이 논문은 2011년도 정부(교육부)의 재원으로 한국연구재단의 기초연구사업 지원을 받아 수행된 것임(과제번호 : NRF-2011-0022445).

* Dept. of Computer Engineering, Hansung University(jychang@hansung.ac.kr)

2014년 03월 14일 접수, 2014년 04월 18일 심사완료 후 2014년 04월 24일 게재확정.

1. 서 론

오피니언 마이닝(opinion mining)은 주관적 문서로부터 작성자의 감정(sentiment)을 추출해 내는 기술로서, 2000년대 이후 많은 연구가 이루어지고 있다[1-4]. 오피니언 마이닝에서의 가장 핵심적인 요소는 오피니언 문서(opinion document)가 해당 객체에 대해서 긍정(positive) 혹은 부정(negative)적인 감정을 갖고 있는지 판단하는 감성 분석(sentiment analysis) 기술이다. 감성 분석은 문서전체에 대해 전반적인 극성(polarity)을 결정할 수도 있고, 특성(features)별로 세분화하여 특성별 극성을 판단할 수도 있다. 이외에도 문서로부터 오피니언의 특성들을 자동적으로 추출하거나[5] 오피니언 문서에 대한 검색, 요약 등에 관한 연구도 이루어지고 있다[6]. 최근에는 트위터와 같은 SNS 환경에서의 오피니언 마이닝에 관한 연구도 활발히 진행되고 있다[7].

상품평(product review)과 같은 오피니언 문서는 제품군(product category) 마다 미리 정해진 특성들의 집합을 정의할 수 있다. 예를 들어 카메라의 경우 렌즈, 화질, 디자인, 가격 등이 특성이 될 수 있으며, 영화의 경우에는 연기력, 연출, CG, 스토리 등을 특성으로 정의할 수 있다. 앞서 설명한 바와 같이 기존 연구에서는 오피니언 문서에서 이러한 특성들에 대해 부정 혹은 긍정인지를 판단하는데 초점을 맞추어 진행하였다. 그러나 실제로 동일한 제품군에 대해서 개별적인 제품에 대해 평가하는 오피니언 문서를 보면 각 제품별로 강조되는 특성들이 다를 수 있다. 예를 들어 카메라의 경우 A 제품은 렌즈나 화질이 우수하여 이에 대한 평가가 다수를 차지할 수 있으며, B 제품의 경우 상대적으로 디자인에 대한 평들이 많을 수 있다. 영화의 경우에는 더욱 이러한 현상이 두드러진다. 예를 들어 SF영화의 경우에는 영상, CG 등에 대한 평이 다수를 차지하는 반면, 드라마의 경우 스토리, 연기 등에 대한 평이 상대적으로 많을 수 있다. 이와 같이 동일한 제품군에 속한다 할지라고 각 제품별로는 특성들에 대한 중요도가 다르게 취급될 수가 있다.

각각의 오피니언 문서가 강조하는 특성에 따라 문서들을 분류하고, 그 중요도에 따라 각 카테고리(category)의 우선순위를 결정할 수 있다면 문서의 검색(search)이나 요약(summarization)에 유용하게 활용될 수 있다. 예를 들어 f_1, f_2, f_3의 특성을 갖는 제품군에 대해서, $r(f_i)\,(1 \le i \le 3)$가 각각 특성 f_i가 강조된 상품평의 집합이라고 가정하자. 이때 $|r(f_1)| > |r(f_2)| > |r(f_3)|$라면, f_1을 강조한 상품평의 비율이 가장 크며, f_3을 강조한 상품평의 비율이 가장 낮다는 것을 나타낸다. 다시 말해서 이 제품은 특성 f_1에 대해 특장점이 존재하며, 상대적으로 f_3에 대해서는 별다른 특징이 없다는 것을 의미한다. 따라서 오피니언 문서를 검색하는 사용자에게 $r(f_1)$, $r(f_2)$, $r(f_3)$로 분류하여 그룹별로 검색 결과를 제공함과 동시에 각 그룹간의 중요도에 따른 우선순위를 제공한다면, 사용자는 이 제품의 전반적인 오피니언 경향을 쉽게 파악할 수 있게 된다.

이러한 일련의 문제를 해결하기 위해 본 논문에서는 특성에 따라 오피니언 문서들을 분류하고, 각 카테고리에 대해 중요도에 따른 우선순위를 결정하는 기법을 제안한다. 특성에 따라 문서들을 분류하기 위해서는 우선 제

품군별로 특성들을 정의해야하는데, 본 논문에서는 특성들의 유사성을 고려한 특성 그래프(feature graph)를 정의하였다. 이를 기반으로 상호 유사한 특성들을 탐색하여 각 오피니언 문서들이 어떠한 특성들에 대해 집중적으로 표현하고 있는지를 결정하는데 이용한다. 이러한 특성 그래프는 오피니언 문서의 유사도(similarity)를 측정하는데 유용하게 활용될 수 있다. 기존의 코사인 유사도(cosine similarity)[8]와 같은 방식은 유사한 용어지만 서로 다른 용어를 사용한 경우 이를 판단할 수 없는 문제를 갖고 있다. 따라서 유사한 특성이지만 서로 다른 용어를 사용한 경우에는 이를 반영할 수 있는 새로운 유사도 측정 기법이 필요한데 본 논문에서는 특성 그래프를 이용하여 이 문제를 해결하였다.

오피니언 문서를 특성별로 분류하기 위해서 본 논문에서는 클러스터링(clustering) 기법을 활용하였다. 클러스터링은 비감독형(unsupervised) 분류 기법으로 학습문서(training document)가 없는 환경에서 상호 유사한 문서들을 분류하는데 유용하게 활용할 수 있다[8]. 또한 클러스터링을 통해 각 클러스터의 크기를 이용하여 클러스터간의 상대적 중요도를 판단할 수 있는데, 이를 이용하여 각 카테고리의 우선순위를 결정할 수 있다. 오피니언 문서의 표현을 위해서 본 논문에서는 벡터 공간 모델(vector space model)을 가정하였다. 단, 문제를 단순화하기 위해서 오피니언 문서에서 나타난 모든 단어(term)들을 모델의 특징(feature)으로 취급하지 않고, 특성 그래프에 존재하는 단어만을 추출하여 해당 문서의 특징으로 취급하였다.

본 논문에서 제안한 제품 특징화 방법의 유용성을 평가하기 위해서 실험을 실시하였다. 실험은 영화평을 대상으로 하였으며, 각 영화평에 대해서 언급된 주요 특성들을 기준으로 클러스터링을 하여 그 결과를 평가하였다. 평가는 정량적 방법과 정성적 방법을 모두 사용하였으며, 실험 결과 본 논문에서 제안된 방법이 오피니언 문서의 분류나 검색 등에 유용하게 활용될 수 있음을 입증하였다.

본 논문의 구성은 다음과 같다. 제 2장에서는 관련연구에 대해서 논하고 제 3장에서는 특성 그래프를 정의한다. 제 4장에서는 문서간의 유사도를 측정하기 위한 방법을 제시하고 제 5장에서는 클러스터링 절차와 방법론에 대해서 설명한다. 제 6장에서는 실험결과를 제시하고 마지막으로 제 7장에서는 결론을 맺는다.

2. 관련 연구

오피니언 마이닝과 관련된 연구는 대부분 오피니언 문서의 분류에 초점을 맞추어 진행되었다. 오피니언 문서를 분류하는 방법으로는 크게 자연어 처리기법과 통계학적 접근법이 있다. [1]에서는 기계 학습(machine learning) 및 자연어 처리 기술을 활용하여, 상품평 데이터에 대한 감성분석 및 분석결과 요약 기법을 제시하고 있으며, 결과물로서 연구목적의 Opinion Observer라는 명칭의 시스템을 개발하였다. 미국 카네기멜론 대학교에서는 RedOpal 시스템을 개발한 사례가 있으며[2], 이는 상품평 데이터와 사용자 평가점수를 활용하여 요약 보고서를 생성하는 기법을 제안하였다. 이 연구에서는 상품 속성과 평가

점수에 대하여 다차원 분석 결과를 보여주고 있지만, 주관적 긍정/부정 평가를 수행하지는 않고 있다. Xiaowen and Bing[3]에서는 문장 구조와 문장 사이의 관계, 문장성분의 패턴 정보 등의 언어 규칙을 이용한 통계학적 방법으로 오피니언 마이닝에 접근하고 있으며, Courses and Surveys[4]에서는 워드넷(WordNet)을 활용하여 어휘의 긍정이나 부정적 의미를 판단하고, 이를 센티워드넷(SentiwordNet)으로 응용하여 감정의 폭을 정량화하는 방법을 제시하고 있다.

오피니언 문서에 대한 클러스터링 연구는 감성분석에 비해 많이 이루어지지 않았다. 이와 관련된 연구들은 대부분 문서 자체의 클러스터링 보다는 관련된 특성들에 대한 클러스터링이 주를 이루고 있다[9, 10]. Zhai et al. [9]에서는 주어진 특성 집합에 대해서 유사하거나 동일한 의미를 갖는 특성 표현들을 EM (Expectation Maximization) 알고리즘을 이용하여 클러스터링하는 방법을 이용하였다. Ahmad [10]에서도 이와 유사한 연구를 진행하였는데, 여기서는 특성과 연관된 감성 단어를 이용하여 미리 정해진 특성들을 k-평균(k-means) 알고리즘으로 클러스터링하는 방법을 제안하였다. 하지만 이러한 연구들은 오피니언 문서 자체가 아닌 유사 특성들을 클러스터링하는 기법으로, 특성에 따라 오피니언 문서 자체를 클러스터링하는 본 연구의 목적과는 거리가 있다고 하겠다.

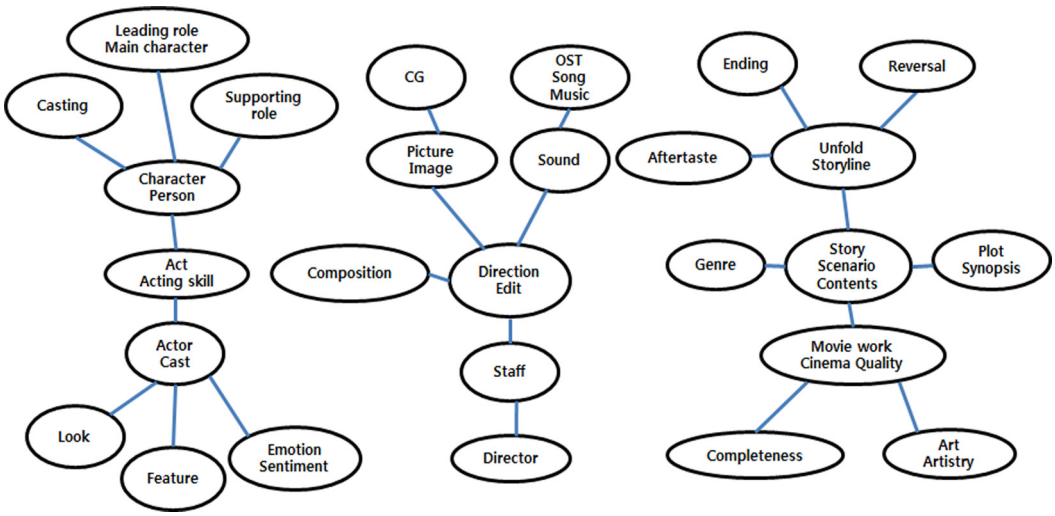
지금까지 살펴본 바와 같이 기존의 연구에서는 본 논문과 같이 오피니언 문서 자체를 특성별로 클러스터링하는 연구는 거의 찾아볼 수 없었다. 또한 특성의 중요도에 따라 우선순위를 결정하는 방법도 기존의 연구에서는 없는 최초의 시도로 볼 수 있다.

3. 특성 그래프

오피니언 마이닝에서 제품에 대한 사용자들의 구체적인 성향을 파악하기 위해서는 제품들의 특성을 정의하는 것이 매우 중요하다. 그러나 오피니언 문서를 작성하는 사용자들은 다양한 방법으로 특성을 표현하므로 제품군별 특성들을 완벽히 파악하는 것은 사실상 불가능하다. 이러한 이유로 오피니언 마이닝이 연구되었던 초기부터 오피니언 문서로부터 특성들을 추출하기 위한 다양한 연구들이 진행되어 왔다[5, 11]. 일단 본 논문에서는 특정 제품군에 대해서 이러한 특성집합들이 사전에 정해져있고 이들 간의 유사성을 비롯한 관계들도 정의되어 있다고 가정한다.

본 논문에서는 특성간의 관련성과 유사성을 표현하기 위해 특성 그래프를 정의한다. 그래프는 $G = \{V, E\}$와 같이 정의할 수 있는데, 여기서 V는 노드(node)의 집합으로 각 노드는 하나의 이상의 특성으로 구성된다. 그리고 E는 두 노드를 연결하는 간선의 집합으로, 각 간선은 연결된 두 노드의 연관성(유사성)을 나타낸다. 예를 들어 <Figure 1>은 영화 도메인들에 대한 특성 그래프의 일부를 보여주고 있다. 이 그래프에서 같은 노드에 속한 특성들은 동일한 의미를 가지며, 두 특성 간 거리가 더 가까울수록 유사한 특성임을 의미한다. 또한 서로 관련이 없는 특성들은 간선으로 연결되지 않도록 정의하였다.

이 그래프를 이용하면 특성간의 유사한 정도를 정량적으로 표현할 수 있다. 간단한 방법



〈Figure 1〉 An Example of Feature Graph in Movie Domain

으로 주어진 두 개의 특성 f_1과 f_2에 대해서 두 특성간의 거리 $d(f_1, f_2)$는 특성 그래프에서 두 특성이 연결된 간선의 수로 정의할 수 있다. 예를 들어 <Figure 1>의 특성 그래프에서 $d(act,\ character)$은 1이며, $d(act,\ leading\ role)$은 2가 된다. $d(act,\ acting\ skill)$은 두 특성이 동일한 노드에 있으므로 0이 된다. 또한 $d(act, edit)$는 두 특성이 간선으로 연결되어 있지 않으므로 무한대의 값을 갖는다. 따라서 두 개의 특성 f_1과 f_2에 대해서 $d(f_1, f_2)$이 작을수록 서로 유사하며, 반대로 클수록 유사정도가 작다고 할 수 있다. 만약 무한대의 값을 가지면 두 특성은 서로 관련 없음을 의미한다. 특성 그래프에서 특성간의 거리는 다음 장에서 설명할 두 오피니언 문서간의 유사도 측정에 활용된다.

주어진 오피니언 문서 집합으로부터 특성 그래프를 생성하는 방법도 하나의 연구 이슈가 될 수 있다. 제 2장의 관련 연구에서 언급한 [9, 10]에서의 연구도 관련된 특성들의 연관성을 알아내기 위한 노력들 중의 하나라고 볼 수 있다. 또한 특성간의 연관성을 정량적으로 계산하기 위해서는 특성 그래프를 가정하지 않더라도 이미 구축된 다양한 온톨로지(ontology)를 활용할 수도 있다. 예를 들어 본 논문의 실험에서 가정한 영화의 경우 영화 온톨로지를 구축한 사례가 있으며[13], 이외 다양한 도메인에 대해서 온톨로지를 구축하거나 구축중인 사례를 찾아볼 수 있다[14]. 또한 워드넷과 위키피디아(Wikipedia)와 같이 범용적으로 사용가능한 온톨로지를 활용할 수도 있다[15]. 그러나 이 이슈는 본 논문의 범위를 벗어나며, 본 논문에서는 특성 그래프가 이미 주어졌다는 가정 하에 오피니언 문서의 클러스터링 기법에 이를 활용한다.

4. 오피니언 문서의 유사도 측정방식

문서 분류나 클러스터링을 위해서는 문서

간의 유사성을 정량적으로 측정하는 방식이 필요하다. 현재까지 데이터마이닝에서 사용되는 대표적인 유사도 측정 방식으로는 유클리디언 거리(Euclidean distance), 자카드 계수(Jaccard coefficient), 코사인 유사도(cosine similarity) 등이 있다[8]. 특히 코사인 유사도는 문서간의 유사도를 측정하는데 많이 쓰인다. 그러나 이러한 유사도들은 특징(문서일 경우 단어)들이 정확히 일치하는 경우에만 계산이 가능하다. 예를 들어 영화 도메인에서 두 문서 d_1과 d_2에서 추출된 특징들이 다음과 같다고 가정하자.

$$d_1 = (act,\ director,\ person,\ ending)$$
$$d_2 = (acting\ skill,\ story,\ character,\ cinematic\ \ quality)$$

이 예에서 두 문서는 동일한 단어를 포함하고 있지 않으므로 유클리디언 거리나 코사인 유사도에서는 0의 값을 갖게 된다. 그러나 'act'와 'acting skill'은 서로 유사한 특징이며, 'person'과 'character'도 유사하다. 'ending'과 'story'는 위의 예들보다는 유사성이 떨어지나 어느 정도는 관련성 있다고 볼 수 있다. 이와 같이 특정 도메인에서 서로 관련 있거나 유사한 특징들의 관계를 유사도에 반영하기 위해서는 새로운 유사도 계산방식이 필요하다. 본 논문에서는 다음과 같은 유사도 측정방식을 활용하였다. 두 개의 특징 벡터 $A = (A_1, \cdots, A_n)$과 $B = (B_1, \cdots, B_n)$에 대해서 A와 B의 유사도 $Fsim(A, B)$는 다음과 같이 계산된다.

$$Fsim(A, B) = \sum_{i=1}^{n} \left(\max\left(\max_{B_j \in B} \frac{A_i \times B_j}{d(i,j)+1}, \max_{A_j \in A} \frac{B_i \times A_j}{d(i,j)+1} \right) \right) \quad (1)$$

이 식에서 $d(i,j)$는 제 3장에서 설명한 특성 그래프에서 정의한 두 특징간의 거리를 의미한다. 특징간의 거리가 멀수록 유사성이 떨어지므로 이 값의 역수로 유사 정도를 반영하게 된다. A_i와 B_j가 특성 그래프에서 서로 연결되지 않으면 무한대의 값을 가지므로 $\frac{A_i \times B_j}{d(i,j)+1}$는 0이 된다. 이 식은 A와 B의 각 A_i와 $B_i (1 \le i \le n)$에 대해서, 이 특징이 두 문서에서 공유되지 않더라도 의미적으로 유사한 다른 특징 있다면, 이를 유사도에 반영하도록 설계한 것이다. 즉, A_i와 B_i에 대해서, A_i의 경우에는 $\frac{A_i \times B_j}{d(i,j)+1}$가 최대가 되는 $B_j (1 \le j \le n)$를 찾고, B_i의 경우에는 $\frac{A_j \times B_i}{d(j,i)+1}$가 최대가 되는 $A_j (1 \le j \le n)$를 찾아 이들 중 더 큰 값을 A_i와 B_i의 유사도로 결정한다. 이러한 과정을 모든 A_i와 $B_i (1 \le i \le n)$에 대해서 수행하게 되면, 최종적으로 이들의 합이 A와 B에 대한 유사도가 된다.

예를 들어 특징들이 (act, director, person, ending, acting skill, story, character, cinematic quality)로 정의되어 있고 두 문서 d_1과 d_2에 포함된 단어들이 위의 예와 같다면 이들에 대한 특징 벡터 $v(d_1)$과 $v(d_2)$는 각각 다음과 같이 표현된다. 편의상 두 문서의 각 특징에 대한 빈도수는 1이라고 가정한다.

$$v(d_1) = (1, 1, 1, 1, 0, 0, 0, 0)$$
$$v(d_2) = (0, 0, 0, 0, 1, 1, 1, 1)$$

이 경우 특징 'act'와 'ending'에 대한 유사도는 <Figure 1>의 특성 그래프와 식 (1)에 의해 각각 다음과 같이 계산된다.

'act'의 유사도 = $\max(\max(0, 0, 0, 0, 1, 0, \frac{1}{2}, 0), \max(0, 0, 0, 0, 0, 0, 0, 0))$

'ending'의 유사도 = $\max(\max(0, 0, 0, 0, 0, \frac{1}{3}, 0, \frac{1}{4}), \max(0, 0, 0, 0, 0, 0, 0, 0))$

따라서 'act'에 대한 유사도는 1이 되며, 'ending'에 대한 유사도는 1/3이 된다. 이러한 방법으로 각 특징에 대한 유사도를 모두 계산하면 다음과 같다.

$$(1, 0, 1, \frac{1}{3}, 1, \frac{1}{3}, 1, \frac{1}{4})$$

마지막으로 이들 값을 모두 더한 값인 4.92가 두 문서에 대한 최종 유사도가 된다.

식 (1)의 유사도 계산 방식을 알고리즘으로 표현하면 <Figure 2>와 같다. 이 알고리즘에서 개별 특징 A_i와 B_i에 대한 유사도는 변수 max에 저장된다. A_i에 대해서는 모든 $B_j(1 \le j \le n)$와의 유사도 중 최댓값을 구하고, B_i에 대해서는 모든 $A_j(1 \le j \le n)$와의 유사도 중 최댓값을 계산한 후, 더 큰 값을 max에 저장한다. 마지막으로 A_i와 B_i에 대한 유사도를 모두 더한 값인 TotalSim이 두 문서 A와 B의 최종적인 유사도가 된다.

5. 상품 특징화를 위한 클러스터링 절차

본 장에서는 지금까지 설명한 특성 그래프와 유사도 계산방식을 이용하여 상품 특성별로 오피니언 문서를 클러스터링하고 각 클러

```
Algorithm FSIM(A, B)
Input  Feature vectors A(A1, …, An), B(B1, …, Bn)
output Similarity between A and B

        TotalSim = 0
        For each i(1 ≤ i ≤ n)
                max = 0
                for each Bj(1 ≤ j ≤ n)
                        partialsim= (Ai × Bj)/(d(i, j)+1)
                        if max < partialsim
                        max = partialsim
                for each Aj(1 ≤ j ≤ n)
                        partialsim= (Bi × Aj)/(d(i, j)+1)
                        if max < partialsim
                        max = partialsim
                TotalSim = TotalSim + max
        return TotalSim
```

〈Figure 2〉 An Algorithm for Similarity Calculation

스터의 우선순위를 결정하기 위한 방법과 절차를 설명한다. 그 과정은 다음과 같이 크게 6단계로 나누어진다.

단계 1 : 오피니언 문서 수집

영화평이나 상품평과 같이 온라인상에 존재하는 오피니언 문서는 주로 웹 크롤링(crawling)을 통해 수집된다. 본 논문에서는 네이버 영화평을 실험대상으로 정하였다. 네이버에서 분석을 위한 영화를 선정한 후, 크롤링 엔진을 이용하여 영화평들을 텍스트 형태로 수집한 후 데이터베이스에 저장하였다.

단계 2 : 제품 특성으로 구성된 특징 벡터 구성

수집된 각 오피니언 문서에 대해서 특징 벡터를 구성하기 위해서는 형태소 분석기를 이용하여 단어들로 구성된 bag of words를 구성해야한다. 일반적으로 문서 분류나 검색을 위한 bag of words는 일부 불용어를 제외한 단어를 모두 포함하나, 본 논문에서는 제품의 특성들이 클러스터링의 주된 수단이므로 3장에서 정의한 특성들만으로 구성된 특징 벡터를 생성한다. 네이버 영화의 경우 140자 이하의 단문으로 구성된 영화평이므로 사전에 정의된 특성 단어가 포함되지 않는 영화평들이 다수 존재할 수 있다. 이때에는 해당 영화평들을 분석 대상에서 제외한다.

단계 3 : 문서간의 유사도 계산

클러스터링 알고리즘을 적용하기 위한 사전 수단으로 문서간의 유사도를 측정한다. 유사도는 제 4장에서의 식 (1)을 이용하여 모든 오피니언 문서 쌍에 대해서 상호 유사도를 계산한다.

단계 4 : 군집화 알고리즘 수행

단계 3에서 생성된 모든 문서 쌍들에 대한 유사도 정보를 이용하여 클러스터링 알고리즘을 수행한다. 본 논문에서는 클러스터링 알고리즘 중에서 많이 사용되고 있는 계층 클러스터링(hierarchical clustering) 알고리즘을 이용하였다[8].

단계 5 : 클러스터간 우선순위 결정

본 논문의 목적이 각 상품의 주요 특성간의 우선순위를 찾는 것이고, 특성별로 클러스터링을 수행하였으므로 클러스터간의 우선순위를 결정해야한다. 그러나 클러스터링의 본래 목적은 유사한 객체들을 그룹핑하기 위한 수단으로 개발된 것으로, 클러스터간의 중요도나 우선순위를 찾는 목적으로는 사용되지를 않는다. 하지만 각 클러스터의 여러 특징을 분석하여 우선순위를 부여할 수 있다. 클러스터간의 특징으로는 응집도(cohesion)나 클러스터의 크기 등을 예로 들 수 있는데 본 논문에서는 단순히 클러스터의 크기로 우선순위를 결정하였다. 그 이유는 클러스터의 크기가 클수록 해당 클러스터를 대표하는 제품특성에 대한 오피니언들이 많다는 것을 의미하기 때문이다.

단계 6 : 클러스터를 대표하는 오피니언 특성 선택

마지막 단계로 각 클러스터를 대표하는 특성들을 선택해야한다. 하나의 클러스터에는 다수의 특성들이 존재할 수 있어 이들 모두를 사용자에게 제공할 수 없으므로, 중요도에 따

라 특성들을 선별할 필요가 있다. 클러스터의 대표 특성 선택에는 다양한 방법이 존재할 수 있다. 예를 들어 클러스터 내에서 응집도가 강한 부분들을 탐색하여 이들을 중심으로 상위 N개의 특성을 선택할 수 있고, 클러스터의 중심점(centroid)과 가까운 상위 N개의 특성을 선택할 수도 있다. 또한 각 클러스터에 언급된 특성들의 출현 빈도순으로 선택할 수도 있다. 본 논문의 실험에서는 중심점과 가까운 문서에 속한 상위 N개의 특성들을 대표 특성으로 선택하였다.

6. 실 험

6.1 실험 환경

본 논문에서 제안한 클러스터링 기법의 성능을 평가하기 위해서 실험을 수행하였다. 실험은 영화 도메인을 이용하였으며, 네이버 영화평을 대상으로 실시하였다. 네이버 영화평 중에서 최근에 흥행에 성공한 '변호인(The Attorney)'과 '겨울왕국(Frozen)' 등 2개의 영화를 선정하였다. 영화에 관한 특성은 122개를 선정하여 수작업으로 특성 그래프를 정의하였다. 또한 클러스터링을 위한 특징 벡터는 122개의 영화 특성만으로 구성되었다. 따라서 각 영화평에 대해서 122개의 특성중 하나라도 언급이 안 된 영화평들은 실험대상에 제외하였다. 최종적으로 두 영화에 대해서 각각 1,000개의 영화평을 클러스터링 대상으로 선정하였다. 클러스터링 알고리즘으로는 계층 클러스터링 알고리즘을 이용하였다. <Figure 1>에서 보는 바와 같이 특성들은 크게 3개의 그룹으로 나누어져 있으므로 클러스터의 수는 3개로 고정하였다.

6.2 실험 결과

실험은 우선 각 클러스터에 포함된 문서의 수로 클러스터들의 우선순위를 정하여, 그 결과가 영화의 분야에 따른 각 특성들의 상대적 중요도를 반영하는지를 판단하였다. <Table 1>은 본 논문이 제안한 유사도 계산방식을 사용했을 경우 각 클러스터별 대표 특성들과 각 클러스터의 크기를 나타내고 있다. 이 표에서 보는 바와 같이 영화 '변호인(The Attorney)'의 경우 '연기(act)'나 '연기력(acting skill)', '배우(actor)'에 관련된 특성을 언급한 클러스터 3이 가장 크고, '이야기(story)'나 '내용(contents)'에 관한 특성을 갖고 있는 클러스터 1이 그 다음을 차지하고 있다. 여기서 '연기(act)', '연기력(acting skill)', '배우(actor)'는 <Figure 1>에서 보는 바와 같이 상호 유사한 특성이므로 이들이 하나의 클러스터로 묶이는 것은 매우 자연스러운 현상이라고 볼 수 있다. 또한 클러스터 1의 대표 특성들인 '이야기(story)', '내용(contents)', '여운(aftertaste)'들도 상호 유사한 특성들임을 알 수 있다. 반면에 클러스터 2가 가장 작은데, 이 클러스터는 '이야기(story)'에 관련된 것과 '연기(act)'에 관련된 영화평들이 혼합되어 있으며, 기타 드물게 등장하는 특성들이 이 클러스터에 속하고 있다. 따라서 영화 '변호인(The Attorney)'은 '연기(act)'가 가장 중요한 특성으로 볼 수 있고, 그 다음이 '이야기(story)'가 주요하게 다뤄지는 특성으로 파악할 수 있다. 영화 '겨울왕국(Frozen)'의 경우는 '노래(song)'나 '영상(picture)' 혹은 이와

〈Table 1〉 Clustering Results with the Proposed Method(FSIM)

Title	Cluster number	Main features	Cluster size
The Attorney	1	story, contents, aftertaste, movie work	377
	2	story, act, actor, movie work, ending, aftertaste, director	132
	3	act, acting skill, actor	491
Frozen	1	song, picture	517
	2	story, contents, song	215
	3	synopsis, story, movie work	268

〈Table 2〉 Clustering Results with Cosine Similarity

Title	Cluster number	Main features	Cluster size
The Attorney	1	story	83
	2	actor, act, acting skill, aftertaste, contents, movie work, story	523
	3	act	394
Frozen	1	song	375
	2	movie work, aftertaste, picture, music, character	470
	3	story	155

관련된 특성들을 언급한 클러스터 1이 가장 크며, '줄거리(synopsis)'나 '이야기(story)'와 관련된 클러스터 3이 그 다음으로 큰 것을 확인할 수 있다. 마지막으로 가장 작은 클러스터 2는 '이야기(story)'와 '노래(song)'가 혼합된 형태를 갖고 있다. 따라서 영화 '겨울왕국(Frozen)'의 경우는 '노래(song)'나 '영상(picture)'에 관련된 특성들이 영화평의 주류를 이루고 있다는 것을 확인할 수 있다.

참고로 <Table 2>는 코사인 유사도를 이용했을 경우의 결과를 보여준다. 이 표에서 보는 바와 같이 영화 '변호인(The Attorney)'의 경우 클러스터 2가 가장 큰 것으로 나타났는데, 이 클러스터는 다양한 특성들이 혼합된 형태를 갖고 있다. 반면에 클러스터 1과 클러스터 3은 각각 '이야기(story)'와 '연기(act)'란 단어가 포함된 영화평들만으로 포함하고 있으며, 그 수가 클러스터 2에 비해 상대적으로 작다. 따라서 이 결과만으로는 어떠한 특성들이 이 영화를 대표하는 지를 파악하기가 쉽지 않다. 이러한 현상은 영화 '겨울왕국(Frozen)'에서도 동일하게 나타난다. 이러한 현상이 발생되는 이유는 코사인 유사도를 이용하였을 경우는 상호 유사한 특성들의 관계를 유사도 계산에 반영할 수 없어 반드시 동일한 이름을 갖는 특성들만으로 클러스터링하는 경향이 있기 때문이다. 따라서 본 논문이 제안한 유사도 계산 방식이 오피니언 문서들을 클러스터링하는데 더 효율적임을 알 수 있다.

다음으로 제안된 유사도 계산방식과 코사

인 유사도 계산방식에 따른 클러스터링의 정확도를 비교 평가하였다. 클러스터링의 정확도를 평가하기 위해서는 정확한 분류가 이루어진 참조 집합(gold standard)이 존재해야 한다. 이를 위해 <Figure 1>의 특성 그래프에 나타난 3개의 그룹을 각 클러스터를 대표하는 특성들로 규정하고, 이를 바탕으로 각 클러스터링 결과를 평가하였다.

클러스터링 결과의 평가 지표는 순도(purity)와 엔트로피(entropy)를 이용하였다[12]. 순도는 클러스터들의 순수성 정도를 나타내는 것으로 0에서 1의 값을 가지며 1에 가까울수록 성능이 좋다는 것을 의미한다. 실험 결과로 나온 클러스터 집합을 $C=\{C_1, \cdots, C_n\}$이라 하고, 참조 집합을 $G=\{G_1, \cdots, G_n\}$이라 하자. 그리고 $P_i(G_j)$를 C_i의 문서 중에서 G_j의 비율이라고 가정하자. 그러면 순도는 다음의 수식으로 계산된다.

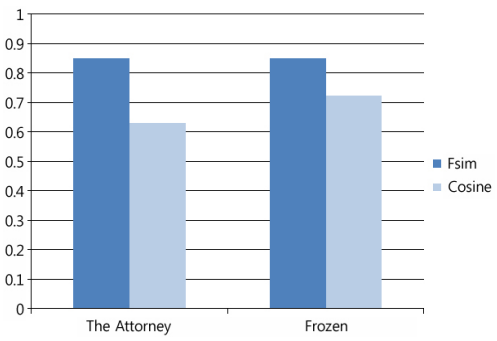
$$purity(C) = \sum_{i=1}^{n} \frac{|C_i|}{|C|} max_j P_i(G_j) \qquad (2)$$

여기서 $\max_j P_i(G_j)$는 C_i의 문서에서 G_j의 비율이 가장 큰 값을 나타내며, 모든 클러스터에 대해 이를 계산한 후 각 클러스터의 크기비율에 따라 평균을 구한다. 반면에 엔트로피는 불순도를 나타내는 것으로 순도와는 달리 0에 가까울수록 성능이 좋다는 것을 의미한다. 엔트로피는 다음의 수식으로 계산된다.

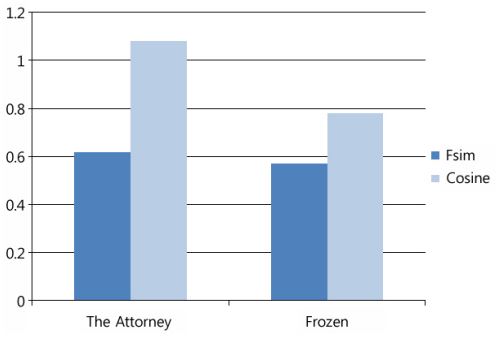
$$entropy(C_i) = -\sum_{j=1}^{n} P_i(G_j) \log_2 P_i(G_j) \qquad (3)$$

$$entropy(C) = \sum_{i=1}^{n} \frac{|C_i|}{|C|} entropy(C_i) \qquad (4)$$

식 (4)에서 보는 바와 같이 엔트로피도 식 (3)을 이용하여 각 클러스터에 대한 엔트로피를 먼저 구한 후에 각 클러스터의 크기비율에 따라 평균을 구한다. 성능평가 결과는 <Figure 3>과 같다. <Figure 3>(a)와 <Figure 3>(b)는 각각 순도와 엔트로피 값을 비교한 결과이다. Fsim은 본 논문이 제안한 유사도 계산방식을 의미하며, Cosine은 코사인 유사도 계산 방식을 의미한다. <Figure 3>(a)에서 보는 바와 같이 영화 '변호인(The Attorney)'에 대한 순도값은 Fsim은 0.85인 반면 Cosine은 0.63으로 Fsim이 더 좋은 성능을 보이고 있다. 영화 '겨울왕국(Frozen)'의 경우에도 Fsim과 Cosine이 각각 0.85, 0.72로 Fsim이 더 좋은 성능을 보인다. 엔트로피 값을 비교한 <Figure 3>(b)도 동일한 결과를 보이고 있다. '변호인(The Attorney)'의 경우 Fsim은 0.62인 반면 Cosine은 1.08이고, '겨울왕국(Frozen)'의 경우 Fsim은 0.57인 반면 Cosine은 0.78이다. '변호인(The Attorney)'와 '겨울왕국(Frozen)'을 비교해봤을 때는 큰 차이점을 보이지 않고 있다. 따라서 영화 장르에 관계없이 본 논문이 제안한 방식이 기존의 코사인 유사도에 비해서 더 좋은 클러스터링 성능을 보인다고 결론을 내릴 수 있다. 그 이유는 앞서 설명한 바와 같이 코사인 유사도의 경우 정확히 일치하는 특성들만 동일한 클러스터에 속하게 되는 반면, 본 논문이 제안한 유사도 계산방식은 같은 용어를 사용한 특성이 아니더라도 서로 유사성이 존재하면 그 값을 정량적으로 평가하여 동일한 클러스터로 그룹화 할 것인가를 결정하기 때문이다.



(a) purity



(b) entropy

〈Figure 3〉 Purity and Entropy of Clustering Results

7. 결 론

본 논문에서는 클러스터링 기법을 활용하여 특정 제품군에 대해서 제품별로 강조되는 특성들을 추출하는 방법을 제안하였다. 이를 위해 특성들의 관계를 표현하는 특성 그래프를 정의하였으며, 오피니언 문서간의 유사도를 정량적으로 계산하기 위한 새로운 방식을 제안하였다. 실험에서는 네이버 영화평에서 장르별로 강조되는 특성들을 선별하였으며, 이 결과로부터 본 논문이 제안한 방법의 유용성을 증명하였다. 다만 본 논문에서 실험한 결과는 제 3장에서 설명한 특성 그래프를 어떻게 정의하느냐에 따라 민감하게 변화할 수 있다. 따라서 본 논문이 제안한 방법의 실용성을 보다 효과적으로 검증하기 위해서는 워드넷, 위키피디아, 영화 온톨로지[13]와 같은 검증된 온톨로지를 활용한 방법도 고려해볼 수 있다. 이 문제는 향후에 관련 연구를 계속 진행할 예정이다.

본 논문에서 제안한 방법은 오피니언 문서의 검색이나 요약에 유용하게 응용될 수 있다. 문서 검색의 경우 클러스터링 결과를 이용하여 검색 결과를 특성별로 범주화(categorization)하여 제공할 수 있다. 또한 각 클러스터에 대한 대표적인 문서들을 선택하여 검색결과를 제공하는데 활용할 수도 있다. 요약의 경우에도 클러스터별로 특징화하여 요약된 결과를 제공할 수 있으며, 요약 정보를 생성하는데 있어서도 각 클러스터의 대표적인 문서를 이용한다면 보다 정확한 결과를 생성하는데 도움을 줄 수 있다.

References

[1] Liu, B., Hu, M., and Cheng, J., “Opinion observer : analyzing and comparing opinions on the Web,” Proceedings of the 14th international conference on WWW, pp. 10-14, 2005.

[2] Scaffdi, C., Bierhoff, K., Chang, E., Felker, M., Ng, H., and Jin, C., “Red Opal : Product-Feature Scoring from Reviews,” Proceedings of the 8th ACM conference on

Electronic commerce, pp. 11-15, 2007.

[3] Xiaowen Ding, and Bing Lui, "The Utility of Linguistic Rules in Opinion Mining," SIGIR 2007, pp. 811-812, 2007.

[4] Courses, E. and Surveys, T., "Using SentiWordNet for multilingual sentiment analysis," IEEE 24th International Conference on Data Engineering Workshop, ICDEW 2008, 2008.

[5] Popescu, A. O., "Extracting product features and opinions from reviews," Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing, pp. 339-396, 2005.

[6] Liu, J., Cao, Y., Lin, C., Huang, Y., and Zhou, M., "Low-Quality Product Review Detection in Opinion Summarization," Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp. 334-342, 2007.

[7] Pak, A. and Paroubek, P., "Twitter as a Corpus for Sentiment Analysis and Opinion Mining," Proceedings of The International Conference on Language Resources and Evaluation, pp. 1320-1326, 2010.

[8] Tan, P., Steinbach, M., and Kumar, V., Introduction to Data Mining, Addision-Wesley, 2006.

[9] Zhai, Z., Liu, B., Xu, H., and Jia, P., "Clustering Product Features for Opinion Mining," Proceedings of the fourth ACM international conference on Web search and data mining, pp. 347-354, 2011.

[10] Ahmad, T., "Clustering Technique for Feature Segregation in Opinion Analysis," International Journal of Computer Applications, Vol. 76, No. 17, pp. 43-49, 2013.

[11] Hu, M. and Liu, B., "Mining opinion features in customer reviews," Proceedings of the 19th national conference on Artificial intelligence, pp. 755-760.

[12] Liu, B., Web Data Mining : Exploring hyperlinks, contents, and usage data, Springer, 2006.

[13] Mo-The movie ontology, http://www.movieontology.org/.

[14] Unified Medical Language System, http://www.nlm.nih.gov/research/umls/.

[15] Rho, J.-H., Kim, H., and Chang, J.-Y., Improving Hypertext Classification Systems through WordNet-based Feature Abstraction, The Journal of Society for e-Business Studies, Vol. 18, No. 2, 2013.

저 자 소 개

장재영	(E-mail : jychang@hansung.ac.kr)
1992년	서울대학교 계산통계학과 (학사)
1994년	서울대학교 계산통계학과 전산과학전공 대학원 (석사)
1999년	서울대학교 계산통계학과 전산과학전공 대학원 (박사)
2000년~현재	한성대학교 컴퓨터공학과 교수
관심분야	데이터베이스, 데이터마이닝