Journal of The Korea Society of Computer and Information
Vol. 17 No. 9, September 2012
www.ksci.re.kr
http://dx.doi.org/10.9708/jksci/2012.17.9.039

의도된 의견 대상의 추출을 위한 경험적 방법

소 윤 규*, 김 한 우**, 정 성 훈***, 김 동 주****

A Heuristic Method for Extracting True Opinion Targets

Yun-Kyu Soh*, Han-Woo Kim*, Sung-Hun Jung*, Dong-Ju Kim**

요 약

일반적으로 사람들은 특정 상품에 관한 의견을 표현할 때 그 상품이 갖는 개별속성에 대해 긍부정 성향을 표시한다. 어떤 경우에는 상품이 갖는 동질의 개별 속성에 대해 포괄적으로 긍부정 성향을 표현하거나 상품 자체에 대해 표현하기도 한다. 따라서 의견검색 분야에서 추출 대상이 되는 의견 속성명에는 상품의 개별 속성명, 이 개별 속성들을 포함하는 전체어, 그리고 상품명이 존재한다. 그러나 의견 대상을 상품명이나 전체어로 표현할 때, 경우에 따라 의견문장 표면에 나타나는 속성명과 의견 작성자가 의도한 실제 대상이 일치하지 않을 수도 있다. 본 논문에서는 의견문장으로부터 의견 대상을 추출하는 방법을 제시한다. 무엇보다 우리는 의도한 대상과 일치하지 않는 속성명으로부터 의도한 대상을 추출하기 위한 새로운 방법을 제안한다. 제시하는 방법에서는 단어간 의존관계를 이용하여 의견속성 후보쌍을 추출하고, 추출된 후보쌍들 중 의견 대상과 일반적으로 빈번히 불일치하는 속성명을 선택한다. 선택된 속성명을 작성자가 의도한 개별속성으로 변경한 뒤, 이를 포함한 전체 의견속성 후보쌍들로부터 적합한 의견속성을 추출하기 위해 사람들이 관심 있어할만한 순으로 재배열하게 된다.

▸ Keywords : 오피니언 마이닝, 의견속성 추출, 의미 분석, 상품속성

Abstract

The opinion of user on a certain product is expressed in positive/negative sentiments for specific features of it. In some cases, they are expressed for a holistic part of homogeneous specific features, or expressed for product itself. Therefore, in the area of opinion mining, name of opinion features to be extracted are specific feature names, holonyms for theses specific features, and product names. However, when the opinion target is described with product name or holonym, sometimes it may not match feature name of opinion sentence to true opinion target intended by

• 제1저자 : 소윤규 • 교신저자 : 김한우 • 책임저자 : 정성훈 • 제4저자 : 김동주
• 투고일 : 2012. 09. 11, 심사일 : 2012. 09. 14, 게재확정일 : 2012. 09. 18.
* 한양대학교 컴퓨터공학과(Dept. of Computer Science and Engineering, Hanyang University)
** 안양대학교 교양대학(College of Liberal Arts, Anyang University)

the reviewer. In this paper, we present a method to extract opinion targets from opinion sentences. Most importantly, we propose a method to extract true target from the feature names mismatched to a intended target. First, we extract candidate opinion pairs using dependency relation between words, and then select feature names frequently mismatched to opinion target. Each selected opinion feature name is replaced to a specific feature intended by the reviewer. Finally, in order to extract relevant opinion features from the whole candidate opinion pairs including modified opinion feature names, candidate opinion pairs are rearranged by the order of user's interest.

▸ Keywords : opinion mining, opinion feature extraction, sentiment analysis, product feature

I. 서 론

우리가 사용하는 인터넷은 많은 정보를 제공한다. 이러한 정보 중에서 사용자의 의견 분석 결과는 판매자에게 판매 전략을 계획하는데 중요한 정보가 되고, 구매자에게 상품을 구입하는데 중요한 정보가 된다. 이처럼 유용한 사용자의 의견을 추출하고 분석하기 위한 연구 분야로서 최근 의견검색 분야가 각광받고 있다.

의견검색은 크게 세 가지의 중요한 이슈가 있다. 첫 번째는 방대한 양의 웹 문서에서 다양하게 표현되어 분포하고 있는 의견문장들을 추출하는 것이다. 일반적으로 리뷰에는 대상에 대한 객관적인 사실을 서술한 부분과 의견 작성자의 주관적인 의견을 서술한 부분이 같이 존재하는데, 이 중 의견분석 대상이 될 수 있는 의견이 포함된 문장을 추출 한다. 두 번째는 추출한 의견문장에서 의견속성을 추출하는 것이다. 의견속성은 의견대상이 되는 상품의 특징을 의미하며 상품명, 상품의 동질 속성을 포괄하는 전체어, 상품의 개별속성명으로 나뉜다. 의견검색에서는 특정 상품의 전체 평가뿐만 아니라 상품의 개별속성들 각각에 대한 평가가 의견분석에 유용하게 사용되기 때문에 중요하게 다뤄지고 있다. 세 번째는 의견 극성 판별이다. 의견 극성 판별은 사용자의 의견이 긍정성향인지 부정성향인지를 판별하는 것으로써 의견 대상 평가에 필수적인 과정이다.

의견검색 시스템은 사용자에게 의견속성별로 각각의 의견분석 결과를 나타내기 때문에 적절한 의견속성의 선정은 매우 중요하다. 대부분의 기존 의견검색 시스템에서는 시스템 설계자의 주관적인 기준 또는 사용자들의 설문에 의해 의견속성을 선정해왔다. 하지만 응용분야에 따라 전체적인 의견이나 특정 상품 그 자체에 대한 의견을 중요하게 생각 할 수도 있고 반대로 상품의 부분적인 개별속성에 대한 의견을 중요하게 생각할 수 있기 때문에 사용자가 원하는 의견속성의 범주를 고려하여 사용자 요구에 적합한 의견속성 추출 방법이 필요하다. 그리고 기존의 연구는 의견문장에서 표면정보와 품사정보만을 사용하여 의견속성을 추출하고, 그 결과 의견단어가 표현하는 대상을 정확하게 찾아 내지 못하는 문제점을 갖고 있다. 또한 기존의 연구는 대용어, 상품명, 상품의 동질 속성을 포괄하는 전체어는 의견대상이 될 수 없다고 판단하여 대용어, 상품명, 전체어가 포함된 의견문장은 의견을 포함하고 있음에도 불구하고 분석대상에서 제외되는 문제점이 있었다.

본 논문에서는 이와 같은 문제점을 해결하기 위해 사용자에게 객관적이고 원하는 범주의 의견속성을 선택하는 방법을 제안하고, 의견단어가 표현하는 대상을 정확하게 찾기 위해 의존 관계를 사용하는 방법과 정보 손실을 막기 위해 대용어, 상품명, 전체어가 포함된 영문 의견문장에서 의견 작성자가 의도한 개별속성을 추출하는 방법을 제안한다.

이후 논문은 2장에서 기존에 존재하는 의견속성 추출 연구에 대해 살펴본 후, 3장에서 우리가 제안한 방법에서 의견속성 추출이 진행되는 과정을 설명한다. 4장에서 실험을 통해 우리가 제안한 방법을 평가하고 5장에서 결론을 내린다.

II. 관련 연구

2000년대부터 의견속성 추출연구가 활발히 진행되었다. Hu. M[1-2]은 문장에서 나타나는 모든 명사와 명사구를 의견속성 후보라고 정의 하였다. 비록 일부 불필요한 후보를 제거하는 과정을 거치지만 무분별하게 의견속성 후보를 생성한다. 그 결과 의견속성 추출 정확도가 낮아지게 되었다. A. Qadir[3]와 G. Qiu[4]는 의견속성 추출의 정확도를 높이기 위해 문장에서 단어들의 의존 관계를 사용하여 추출한 의견속

성과 의견단어를 이용해 의견문장을 추출하는 방법을 제안하였다. 그 결과 불필요한 의견속성 후보의 생성을 최소화 하였다. 그러나 이와 같은 방법으로는 대용어, 상품명, 전체어로 의견속성 후보가 표현된 의견문장에서 개별 의견속성 후보 추출이 불가능하였고 그 결과 의견 분석에 활용되지 못하였다. Mahesh. Joshi[5]와 Ellen. Riloff[6]는 단어들의 의존 관계정보에 추가로 N-gram 기법을 사용하여 의견문장 내에서 의견속성이 출현하는 패턴을 추출하였다. 그러나 추출된 의견속성 패턴은 제한적이기 때문에 다양한 형태의 의견문장에 적용하기 힘들었다. 그 결과 의견을 포함하고 있지만 의견 분석에서 제외되는 문장이 많았다.

위의 연구들은 공통적으로, 의견속성을 추출하기 위한 하나의 방법을 제공하지만 응용분야에 따라서 의견속성의 의미적 구체성에 대한 다양한 방법이 존재 할 수 있다는 점을 고려하지 못하였다. 우리는 이 같은 기존 연구들의 단점을 해결하기 위해 의존 관계를 사용한 의견속성 추출 규칙을 제안하고 대용어, 상품평, 전체어로 표현된 의견문장을 최대한 포함하여 분석할 수 있는 방법을 제안한다.

III. 본 론

본 논문에서 제안하는 의견속성 추출 시스템의 구조는 그림 1과 같다.

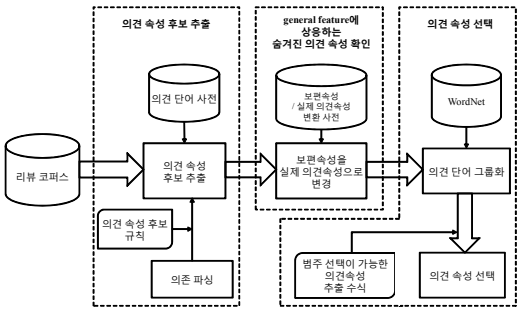

그림 1. 의견속성 추출 시스템 흐름도
Fig 1. The overview of proposed opinion feature extraction system

본 논문에서 제안하는 의견속성 추출 과정은 크게 4단계로 나눈다. 첫 번째 단계는 의견문장을 추출하는 단계에서 사용할 의견단어 사전과 보편속성을 개별속성으로 변환하는 단계에서 사용할 목록을 구축한다. 보편속성은 의견단어에 의해 서술되는 대상을 개별속성으로 표현하지 않고 대용어, 상품명, 상품의 동질 속성을 포괄하는 전체어와 같이 함축적으로 표현한 것을 의미한다. 두 번째 단계는 의견을 포함한 문장으로부터 의견속성 후보들을 추출한다. 세 번째 단계는 추출한 의견속성 후보들 중에서 보편속성을 앞서 구축한 변환목록을 사용하여 개별속성으로 바꿔준다. 마지막 단계는 본 논문에서 제안하는 전체어, 개별속성의 특징을 강조 할 수 있는 의견속성 추출방법을 사용하여 전체 의견문장들을 대표하는 의견속성들을 결정한다.

1. 의견단어 사전과 변환목록 구축

본 논문에서는 두 종류의 사전 및 변환목록을 구축한다. 하나는 의견단어 사전이고 다른 하나는 보편속성을 개별속성으로 변환하는 목록이다.

1.1 의견단어 사전 구축

의견을 포함하지 않은 문장은 의견속성 추출에 필요하지 않기 때문에 본 논문에서는 의견을 포함한 문장만을 사용한다. 이 단계에서는 의견을 포함한 문장을 찾기 위해 의견단어 사전을 SentiWordNet[7]을 활용하여 구축한다.

SentiWordNet에 속한 각 단어들은 positive, negative, objective 점수를 가지고 있는데, 세 점수의 합은 1이고 각 점수는 단어가 가진 긍정, 부정의 강도와 1에서 그 둘을 뺀 나머지를 나타낸다. 단어의 objective 점수가 0에 가까울수록 긍정 또는 부정 점수가 더 높으므로 의견이 더 명확하게 들어있다고 볼 수 있다. 본 논문에서는 명백하게 긍정 또는 부정의 의견을 담고 있는 단어만을 사용하기 위하여 objective 점수가 0.75 이하인 단어만을 사용하였다.

1.2 변환목록 구축

사람들은 자신의 의견을 글로 표현할 때, 개별속성을 사용하여 표현하는 대신 대용어, 상품명, 전체어와 같은 보편속성을 사용하여 표현하는 경우가 종종 있다.

자동차에 대한 200개의 리뷰를 구성하는 1,504개 문장을 수동으로 분석한 결과, 보편속성들은 전체 의견속성 후보쌍에서 10.9%를 차지하는 것을 확인하였다. 기존 연구[8-9]에서와 같이 분석대상에서 보편속성들을 제외한다면 많은 의미 있는 정보를 잃어버릴 수 있고 정확한 의견속성 후보를 추출하지 못할 수도 있다.

1) a. my car is good
   b. my car is beautiful

1-a)는 "car" 자체에 대한 의견을 가지고 있지만 1-b)의

경우는 “design”에 대한 의견을 가지고 있다. 이와 같이 보편속성을 사용하여 표현된 문장들의 일부는 보편속성 그 자체에 대한 의견을 담고 있고 또 일부는 개별 의견속성에 대한 의견을 갖고 있다. 개별 의견속성에 대한 의견을 갖고 있지만 보편속성을 사용하여 표현한 의견문장들은 의견단어가 가리키는 개별 의견속성을 알 수 없기 때문에 의견속성 선택 과정에서 제외되어 의견 오분석의 원인이 된다. 이러한 문제점을 해결하기 위해 이 단계에서는 의견속성 후보가 개별속성을 내포하고 있는 보편속성인 경우 보편속성이 실제 의미하는 개별 의견속성으로의 변환을 위해 변환목록을 구축한다. 아래 표 1은 구축한 변환목록의 일부이다.

표 1. 변환목록
Table 5. General feature / true opinion feature dictionary

도메인	의견단어	개별속성
car	fast, quick, rapid, speedy, swift, slow	speed
car	large, big, extensive, roomy, narrow, cramped	space
car, cellular phone	pretty, beautiful, cute, adorable	design
car, cellular phone	cheap, inexpensive, expensive, costly	price
movie	interesting, sad, fun	contents

표 1에서 의견속성들은 제약적 경향을 가진 단어들로 표현된다. 표 1에 나온 각각의 의견단어들은 특정 속성들만 표현하는 경향이 있다. 일반적으로 특정 의견단어들은 특정 유형의 속성만을 표현하는 경향이 있기 때문에, 이러한 경향을 반영하여 보편속성에 대한 변환목록을 구축한다.

2. 의견속성 후보 추출

상품 리뷰 코퍼스로부터 의견속성 후보들을 추출하는 단계는 두 개의 하위 단계로 나눠진다. 첫 번째 단계는 상품 리뷰 코퍼스에서 의견이 포함된 문장을 추출하는 것이고 두 번째 단계는 의견이 포함된 문장에서 의견속성 후보들을 추출하는 것이다.

2.1 의견문장 추출

이 단계에서는 상품 리뷰 코퍼스로부터 의견이 포함된 문장을 찾는다. 문장에 의견이 포함되어있는지 여부를 구별하기 위하여 1.1에서 구축한 의견단어 사전을 사용한다. 상품 리뷰 코퍼스에서 추출된 의견이 포함된 문장은 아래 예제 2)와 같다.

2) a. Beautiful design, great ride, exceptionally quiet, very good performance, loaded with amenities.
b. Interior is very roomy and comfortable.
c. I looked into buying an inexpensive dvd player that had more than the standard set of features and this item seemed to be the best in that category
d. it is a fantastic camera and well worth the price

2)의 문장에서 밑줄 그은 단어들은 의견단어 사전에 포함된 단어들로 의견을 담고 있다. 이와 같이 한 개 이상의 의견단어가 포함된 문장들을 의견문장으로 간주한다. 우리는 2)와 같은 의견문장들만 이후 분석 과정의 대상으로 한다.

2.2 의견속성 후보 추출

의견속성 후보 추출 단계에서는 이전 단계에서 추출된 의견문장에 포함된 의견단어가 표현하는 의견속성 후보를 찾는다. 이를 위해 스탠포드 의존 분석기(Stanford dependency parser)[10]와 의견속성 후보 추출 규칙을 사용한다. 우리는 스탠포드 의존 분석기를 사용하여 품사 정보, 문장의 구조 정보와 의존 관계(dependency relation) 정보를 얻는다. 의존관계 정보는 의견단어와 그에 상응하는 의견속성 후보를 찾는데 사용한다. 스탠포드 의존 분석기의 의존 관계 정보는 basic, collapsed dependencies, collapsed dependencies with propagation of conjunct dependencies 등의 정보를 지원하는데, 본 논문에서는 collapsed dependency 정보를 사용한다. collapsed dependency은 어떤 두 단어가 전치사나 연결어 등을 사이에 두고 의존관계가 성립되어있는 경우, 해당 전치사/연결사를 무시하고 두 단어를 직접적인 의존관계로 연결해주는 방식이다.

3) Other than that pictures taken in the dark are not as nice as I'd like them.

예를 들어, 3)에서 prep(taken-5, in-6,) pobj(in-6, dark-8) 라는 의존관계가 있다면, taken과 dark는 in이라는 전치사를 두고 의존관계가 성립되어있다. collapsed dependency에서는 이 둘이 하나로 합쳐져서 prep_in(taken-5, dark-8)이 된다. collapsed dependency을 사용함으로써, 만약 의견단어와 의견속성 사이에 전치사/연결사가 포함된 다

른 의존 관계가 존재하더라도 의견단어와 의견속성 후보 사이의 직접적인 의존 관계를 찾을 수 있다. 이러한 의존 관계를 활용하여 의견속성 후보를 추출하는 규칙은 규칙 1과 같다.

4)는 규칙 1을 적용한 예제이다. 4-a)의 문장에서는 규칙 1-a)에 의해 amod(interior-5, nice-4) 의존관계의 "interior"가 의견속성 후보로 추출된다. 4-b)에서는 규칙 1-b)에 의해, 4-c)에서는 규칙 1-c)에 의해 각각 nsubj(good-8, Braking-1), cop(good-8, is-6) 의존관계에서 "Braking"이, prep_of(plenty-14, room-16) 의존관계에서 "room"이 의견속성 후보로 추출된다.

규칙 1. 의견속성 후보 추출 규칙
Rule 1. Candidate opinion feature extraction rule

a. 문장을 의존 분석(dependency parsing)한 결과에 amod 또는 advmod 관계가 포함되어 있고, 의견단어가 해당 관계의 종속자 위치에 있다면, 지배자 위치에 있는 단어는 의견속성 후보이다.
b. 문장을 의존 분석한 결과에 cop관계와 nsubj 관계가 포함되어 있고, 의견단어가 cop관계에 속해있다면, nsubj 관계의 종속자 위치에 있는 단어는 의견속성 후보이다.
c. 문장을 의존 분석한 결과에 prep_x 관계가 포함되어 있고, 의견단어가 해당 관계의 지배자 위치에 있다면, 종속자 위치에 있는 단어는 의견속성 후보이다.

4) a. It has a nice interior and rides quiet and smooth.
amod(interior-5, nice-4)
b. Braking with 4 wheel discs is extremely good.
nsubj(good-8, Braking-1), cop(good-8, is-6)
c. I can carry my 48" hard bowcase with other luggage and have plenty of room left over
prep_of(plenty-14, room-16)

위 규칙 1에 의해 추출된 의견속성 후보와 상응하는 의견단어를 아래의 5)와 같이 쌍으로 묶는다. 우리는 이것을 의견속성 후보쌍(candidate opinion pair)이라고 부른다.

5) a. ⟨engine, powerful⟩, ⟨engine, feeble⟩, ⟨engine, attractive⟩
b. ⟨price, good⟩, ⟨price, cheap⟩, ⟨price, economical⟩
c. ⟨design, luxurious⟩, ⟨design, lovely⟩, ⟨design, posh⟩

3. 보편속성에 상응하는 개별속성 확인

이 단계에서는 개별 의견속성에 대한 의견을 갖고 있지만 보편속성을 사용하여 표현한 의견문장들에서 생기는 오차를 해결하기 위해 보편속성이 실제 의미하는 개별 의견속성을 추출하는 작업을 한다.

6) a. "My car is very beautiful"
b. "Designs of my car is very beautiful"

6-a)에서 의견 작성자의 정확한 의도는 6-b)일 것이다. 하지만 사람들은 a의 표현을 쓰는 경우가 종종 있다. 이와 같은 경우 "car", "beautiful"이 후보쌍 ⟨car, beautiful⟩로 추출되지만 보편속성/개별속성 변환목록을 통해 ⟨design, beautiful⟩로 변환되어야 한다.

아래 7)은 이전 의견속성 후보 추출 단계에서 제안하는 세 가지 규칙에 의해 추출된 의견속성 후보쌍들 중에서 보편속성이 포함된 의견속성 후보쌍이고, 8)은 7)의 보편속성 중 개별속성을 내포하고 있는 보편속성을 개별속성으로 변환한 후의 의견속성 후보쌍이다.

7) a. ⟨car, good⟩, ⟨car, quiet⟩, ⟨car, beautiful⟩
b. ⟨Equinox, large⟩, ⟨TaurusX, inexpensive⟩
c. ⟨it, fast⟩
8) a. ⟨car, good⟩, ⟨noise, quiet⟩, ⟨design, beautiful⟩
b. ⟨space, large⟩, ⟨price, inexpensive⟩
c. ⟨speed, fast⟩

7-a) 의견속성 후보쌍에는 "car"라는 전체어가 의견속성 후보이다. 7-b)의 의견속성 후보는 상품명이고. 7-c)의 경우는 대용어이다. 7-a)에서 ⟨car, good⟩의 경우 "good"이라는 의견단어는 "car" 라는 보편속성에 대한 의견을 가지고 있지만 ⟨car, quiet⟩ 와 ⟨car, beautiful⟩의 경우 의견단어는 각각 "noise", "design"이라는 개별속성을 "car"라는 보편속성에 내포하고 있다.

이전 변환목록 구축 단계에서 구축한 보편속성/개별속성 변환목록을 사용하여 7)의 보편속성이 포함된 후보쌍에서 각 의견단어에 해당하는 의견속성을 변환한다. 그 결과 7)의 예제는 8)과 같이 바뀐다.

자동차 리뷰에서 조사한 결과 164개의 보편속성이 포함된 후보쌍 중 약 34.8%를 차지하는 57개를 오분석 없이 개별속성의 후보쌍으로 변환할 수 있으므로 의견속성 추출의 정확률

을 향상시키는데 기여할 수 있다.

4. 의견속성의 선택

추출된 의견속성 후보들 중에는 사용자에게 유용하지 않은 의견속성들도 속해있다. 이 단계에서는, 유용한 의견속성을 선택하기 위한 방법을 제안한다. 본 논문에서 제안하는 방법은 사람들의 성향을 반영한다. 많은 사람들이 관심을 가지는 의견속성 후보는 자주 발생하지만, 이를 표현하는 의견단어의 동종범주 개수는 적다. 이러한 특성을 반영하기 위해, 의견속성 빈도와 범주 빈도(Category Frequency : CF)를 적용한 점수계산 수식을 제안한다. 동종범주는 동종의 의견속성을 표현하는 의견단어의 집합을 뜻하며, 범주 빈도는 의견속성에 상응하는 동종범주의 개수를 의미한다. 자주 발생하는 의견속성은 유용한 의견속성라고 생각될 수 있기 때문에 제안한 수식에서 높은 점수를 주었다. 하지만, 점수계산을 할 때 의견속성 빈도만을 고려한다면 문제점이 생기게 된다. 의견단어가 가진 의견의 대상이 되기 때문에 앞서 보편속성을 개별속성으로 변환하는 단계에서 변환되지 않는 상품명 또는 품목과 같은 보편속성의 빈도수는 매우 높은 편이다. 또한, 보편속성 외에 상품의 전체적 특징을 표현하는 의견속성의 빈도수 역시 매우 높은 편이다. 반면에 상품의 세부적인 특징을 표현하는 개별속성들의 빈도수는 이러한 전체적 특징에 비해 상대적으로 낮다. 때문에 보편속성을 포함한 전체적 특징만이 높은 점수를 획득하게 되고 개별속성은 점수가 낮아 무시된다는 문제점이 생기게 된다. 이러한 문제점을 해결하기 위해, 제안한 수식에 CF를 적용하면 개별 의견속성이 선택되는데 긍정적인 영향을 줄 수 있다.

먼저 CF를 계산하기위해서 동종의 의견단어를 하나의 범주로 묶는 작업을 수행해야한다. WotdNet에서의 동의어와 반의어 관계에 기반으로 범주를 결정한다. 하나의 의견속성을 표현하는 의견단어는 긍정 또는 부정으로 표현 된다. 반의어와 동의어는 하나의 대상에 대하여 같은 의견을 나타내거나 또는 반대 의견을 나타내는 단어를 뜻하기 때문에, 반의어와 동의어는 하나의 범주로 묶는다. 그 예는 아래 9)와 같다.

9) 〈price, expensive〉, 〈price, cheap〉

9)의 "expensive"와 "cheap"은 서로 반의어 관계이다. 두 단어는 같은 의견속성 "price"와 상응하기 때문에, 동일한 범주에 속한다. 두 의견단어 X와 Y가 하나의 범주에 속하는 것을 결정하기 위해 X, Y의 유사도를 계산한다. 계산 수식은 아래 수식 (1)과 같다.

$$SCORE_{category\ silarity\ X,Y} = (W_1 + W_2) \times \frac{|A \cap B|}{|A \cup B|} \quad (1)$$

W_1은 두 단어사이의 유사도이고, W_2는 두 단어사이의 반의어 관계의 유사도이다. WotdNet에서 지원하는 의견거리(sementic distance)를 깊이로 표현할 때, W_1 경우 이 깊이를 그대로 사용하고, W_2의 경우는 X가 Y의 반의어라면 이 깊이를 1로 보고, X가 Z의 유의어이고 Z가 Y의 반의어라면 X와 Y의 깊이를 2로 보는 방식으로 깊이를 계산한다. W_1와 W_2 는 이 깊이의 역수로써 깊이가 작을수록 X와 Y 사이의 유사도가 높아진다. A와 B는 각각 의견단어 X와 Y에 상응하는 의견속성의 묶음이다. $|A \cap B|$ 는 A와 B에서 공통적으로 등장하는 의견속성의 개수이고 $|A \cup B|$ 는 A와 B의 전체 의견속성 개수이다. 두 의견단어에 대한 점수가 미리 정의된 임계값을 초과할 경우에만 두 의견단어는 같은 범주로 묶인다.

범주로 묶는 작업이 완료된 후, 의견속성이 유용한지 여부를 평가하기위한 점수를 계산한다. 점수를 계산하기 위한 방법은 아래의 수식 (2)를 사용한다.

$$SCORE_{OF} = \alpha \frac{\text{의견속성 빈도}}{\text{전체 의견속성 빈도}} + (1-\alpha)\left(1 - \frac{\log_n(CF)}{\text{전체 범주 개수}}\right) \quad (2)$$
$$(0 \le \alpha \le 1)$$

α는 전체적 특징과 개별속성에 대한 영향을 조절하는 가중치이다. $0 \le \alpha \le 1$ 조건은 정규화를 위한 것이다. 일반적으로 전체 그룹 개수는 전체 의견속성 빈도에 비해 매우 작은 값을 가지고 있고 CF와 큰 차이를 보이지 않기 때문에, 그대로 사용할 경우 가중치 값과 상관없이 수식 계산결과에 큰 영향을 줄 수 없다. 때문에 수식에서 가중치 값 사이의 균형을 위해 $\log_n(CF)$를 사용하였다. $\log_n(CF)$에서 n을 조절함으로써 가중치 값 사이의 균형에 영향을 줄 수 있다. 수식 (2)에서 α값이 커질수록 전체적 특징의 점수가 높아지고 작아질수록 개별속성의 점수가 높아진다.

사용자는 α와 n을 조절함으로써 전체적 특징과 개별 의견속성 중 자신이 원하는 범주를 선택할 수 있다. 최종적으로, 임계값을 넘어서는 높은 점수를 획득한 의견속성들을 추출한다.

IV. 실 험

1. 실험 데이터

우리는 3개의 도메인(자동차, 영화, 휴대폰)에 대한 상품 리뷰를 실험 데이터로 사용하여 실험하였다. 실험을 위해 세 개의 웹 사이트*에서 과도하게 생략된 문장, 통신어를 사용한 문장들은 실험에서 제외하고 각 상품에 대하여 200개씩 총 600개의 리뷰를 수집하였다. 그리고 평가를 위해, 수동으로 리뷰 코퍼스에서 의견속성 후보와 보편속성을 표시하였다.

2. 실험 평가 방법 및 결과

제안한 방법은 3가지 측면에 대하여 평가하였다. 첫 번째로, 의존관계를 이용하여 의견속성 후보를 추출하는 규칙의 정확도를 평가하였다. 두 번째로, 보편속성이 내포하고 있는 개별속성을 추출하는 방법의 정확도를 평가하였다. 마지막으로, 의견속성의 의미적 구체성을 조절하기위해 제안한 수식의 유효성을 평가하였다.

의견속성 후보 추출 규칙과 보편속성이 내포하고 있는 개별속성을 추출하는 방법의 정확도는 표 2와 같다.

표 2. 의견속성 후보 추출 규칙 (A)과 보편속성이 내포하고 있는 개별속성 후보를 추출하는 방법 (B)의 정확도
Table 7. The accuracy of the candidate opinion feature extraction rule (A) and the method of replacing general feature to true opinion feature (B)

도메인	car	movie	cellular phone
리뷰 개수	200	200	200
문장 개수	1504	840	1292
의견속성 후보 개수	1285	640	1079
보편속성 개수	164	68	116
(A)	0.866	0.814	0.885
(B)	0.890	0.823	0.842

표 2에서 (A)는 의견속성 후보 추출 규칙의 정확도를 나타내며 (B)는 보편속성/개별속성 변환목록을 통해 추출하는 방법의 정확도를 나타낸다.

그림 2는 자동차 리뷰 코퍼스에 속한 의견속성 후보의 점수를 계산한 결과를 보여준다. 실험에 앞서, 우리는 자동차 리뷰에서 58개의 의견속성 후보와 71개의 의견단어를 수동으로 찾아 제안한 그룹화 방법을 사용하였고, 15개의 그룹이 만들어졌다. 우리는 제안한 수식의 변수 n을 5로 정하고 가중치 α를 5가지 임의의 값으로 바꿔가면서 각 의견속성 후보의 점수를 계산하였다. 이 계산의 결과로 나온 점수를 기준으로 순위를 매겨 정리해본 결과, 제안한 수식은 α를 이용하여 추출되는 의견속성의 범주를 선택할 수 있음을 그림 2에서 보여준다. $\alpha=0.9$ 일 때, 높은 점수를 갖는 의견속성(design, feel, speed 등)는 낮은 점수를 갖는 의견속성(outline, comfort, acceleration 등) 보다 더 전체적 특징을 표현한다. 예를 들어, speed는 acceleration과 유사한 의미를 가지고 있지만 acceleration보다는 전체적인 특징을 표현하는 의견속성이다. 반대로, 가중치가 $\alpha=0.9$ 에서 $\alpha=0.1$ 로 바뀌면 점수에 의한 의견속성 순위가 대부분 뒤집어지고 개별속성이 전체적 특징에 비해 더 높은 점수를 갖게 된다. 순위가 바뀌는 것은 점선 화살표를 보면 확인 할 수 있다. 이 실험 결과는, 우리가 제안한 의견속성 추출 방법을 통해 전체적 특징과 개별속성 중 사용자가 중요하게 생각하는 범주의 의견속성을 추출할 수 있음을 보여준다.

V. 결 론

이 논문에서, 우리는 상품 리뷰에서 사용자 요구에 적합한 범주의 의견속성을 추출하는 방법을 제안하였다. 의견이 포함된 문장에서 의견단어의 의존 관계정보를 사용한 추출 규칙을 활용하여 기존의 연구보다 정확한 의견속성 후보를 추출하였고 개별 의견속성을 대신하여 표현된 보편속성을 실제 내포하고 있는 개별 의견속성으로 변환함으로써 의견정보 손실을 막아 의견속성 추출 정확도를 높였다. 그리고 전체적 특징 / 개별속성의 범주 선택이 가능한 수식을 이용하여 응용분야에 맞는 적절한 범주의 의견속성을 선정 할 수 있었다.

우리는 수집된 실제 데이터에 제안한 방법을 사용하여 의견속성 추출을 시도할 때, 분석하지 못하는 생략이 포함된 문장, 통신어, 약어를 사용한 문장들이 많은 것을 확인하였다. 이러한 문장들은 단어를 인식할 수 없거나 문법에 맞지 않아 분석 정확도가 20%이하로 상당히 낮았다. 이처럼 분석이 제대로 되지 않는 문장들에 대해서는 의견속성을 추출하지 못하는 문제가 있기에 이를 해결하기 위한 연구가 필요하다.

* http://www.amazon.com
http://autos.yahoo.com
http://www.mrge.com

no	α = 0.9				α = 0.7		α = 0.5		α = 0.3		α = 0.1	
	opinion feature	feature frequency	GF	score	opinion feature	score	opinion feature	score	opinion feature	score	opinion feature	score
1	mileage	110	5	0.1704	mileage	0.3399	mileage	0.5095	acceleration	0.7033	acceleration	0.9011
2	design	80	5	0.1494	design	0.3236	acceleration	0.5054	transmission	0.7033	transmission	0.9011
3	feel	68	5	0.1410	feel	0.3170	transmission	0.5054	drive	0.7030	drive	0.9010
4	price	66	5	0.1396	price	0.3160	drive	0.5051	weight	0.7028	weight	0.9009
5	body	64	6	0.1374	engine	0.3135	weight	0.5047	floor	0.7026	floor	0.9009
6	speed	56	5	0.1326	service	0.3132	floor	0.5043	condition	0.7026	condition	0.9009
7	interior	54	5	0.1312	body	0.3126	condition	0.5043	exhaust	0.7023	exhaust	0.9008
8	engine	50	3	0.1305	comfort	0.3110	exhaust	0.5039	preference	0.7023	preference	0.9008
9	performance	46	4	0.1265	speed	0.3105	preference	0.5039	quality	0.7023	quality	0.9008
10	service	40	2	0.1251	interior	0.3094	quality	0.5039	airbag	0.7023	airbag	0.9008
11	appearance	44	5	0.1242	performance	0.3078	airbag	0.5039	tax	0.7021	tax	0.9007
12	comfort	36	2	0.1223	acceleration	0.3076	tax	0.5035	roof	0.7021	roof	0.9007
13	handle	40	5	0.1213	transmission	0.3076	roof	0.5035	benefit	0.7021	benefit	0.9007
14	stability	30	3	0.1165	drive	0.3071	benefit	0.5035	mirror	0.7019	mirror	0.9006
15	chair	26	2	0.1156	weight	0.3065	mirror	0.5031	part	0.7019	part	0.9006
⋮												
22	acceleration	14	1	0.1098	airbag	0.3054	steep	0.5023	gear	0.7012	gear	0.9004
23	transmission	14	1	0.1098	tax	0.3049	gear	0.5019	button	0.7012	button	0.9004
24	drive	13	1	0.1091	roof	0.3049	button	0.5019	personality	0.7009	personality	0.9003
25	weight	12	1	0.1084	benefit	0.3049	personality	0.5016	outline	0.7009	outline	0.9003
26	space	16	2	0.1083	mirror	0.3044	outline	0.5016	movement	0.7007	movement	0.9002
27	Finishes	16	2	0.1083	part	0.3044	service	0.5012	service	0.6892	service	0.8773
28	reaction	16	2	0.1083	driving	0.3044	movement	0.5012	comfort	0.6883	comfort	0.8770
29	floor	11	1	0.1077	appearance	0.3040	comfort	0.4997	chair	0.6860	chair	0.8762
30	condition	11	1	0.1077	control	0.3038	design	0.4978	trunk	0.6848	trunk	0.8758
31	noise	15	2	0.1076	strongness	0.3033	engine	0.4967	color	0.6846	color	0.8757
32	insurance	15	2	0.1076	capacity	0.3033	chair	0.4958	logo	0.6846	logo	0.8757
33	exhaust	10	1	0.1070	locking	0.3033	trunk	0.4938	seat	0.6843	seat	0.8756
34	preference	10	1	0.1070	steep	0.3033	color	0.4934	space	0.6836	space	0.8754
35	quality	10	1	0.1070	trunk	0.3028	logo	0.4934	Finishes	0.6836	Finishes	0.8754
36	airbag	10	1	0.1070	gear	0.3027	feel	0.4931	reaction	0.6836	reaction	0.8754
⋮												
43	part	8	1	0.1056	handle	0.3018	insurance	0.4915	engine	0.6798	engine	0.8629
44	driving	8	1	0.1056	seat	0.3017	upkeep	0.4911	mileage	0.6790	stability	0.8614
45	control	7	1	0.1049	movement	0.3016	starting	0.4907	stability	0.6751	power	0.8611
46	strongness	6	1	0.1042	power	0.3011	noisy	0.4895	power	0.6744	wheel	0.8611
47	capacity	6	1	0.1042	wheel	0.3005	style	0.4895	wheel	0.6742	performance	0.8519
48	locking	6	1	0.1042	space	0.3001	performance	0.4892	design	0.6720	mileage	0.8486
49	steep	6	1	0.1042	Finishes	0.3001	stability	0.4889	performance	0.6705	design	0.8462
50	noisy	10	2	0.1041	reaction	0.3001	speed	0.4885	feel	0.6692	feel	0.8453
51	style	10	2	0.1041	noise	0.2996	body	0.4878	price	0.6687	price	0.8451
52	gear	5	1	0.1035	insurance	0.2996	power	0.4878	speed	0.6664	speed	0.8444
53	button	5	1	0.1035	upkeep	0.2990	interior	0.4877	interior	0.6659	interior	0.8442

그림 2. 가중치 조정에 따른 의견속성의 점수 변화
Fig 2. The change of score of opinion feature for controlling weight parameters

그리고 기존의 대다수 의견검색 연구들은 형용사만을 의견 단어로 정의하여 사용하지만 의견정보는 형용사뿐만 아니라 동상, 명사 등에도 존재하므로 의견 분석에 활용되어야 한다. 이와 같은 이슈들에 대해서 차후에 연구할 것이다.

참고문헌

〔1〕 Hu. M, and Liu. B, "Mining and summarizing customer reviews," In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining, Seattle, WA, USA, pp. 168-177, 2004.

〔2〕 Hu. M, and Liu. B, "Mining opinion features in customer reviews," In Proceedings of American Association for Artificial Intelligence, pp. 755-760, 2004.

〔3〕 A. Qadir, "Detecting opinion sentences specific to product features in customer reviews using typed dependency relations," eETTs Proceeding of the Workshop on Events in Emerging Text Types, Singapore, pp. 38-43, 2009.

〔4〕 G. Qiu, B. Liu, J. Bu, C. Chen, "Opinion word expansion and target extraction through double propagation," In Proceedings of the ACL, pp. 9-27, 2011.

〔5〕 Mahesh. Joshi, and Carolyn. Penstein-Rose, "Generalizing dependency features for opinion mining," In Proceedings of the ACL-IJCNLP 2009 Conference Short Paper," Suntec, Singapore, pp. 313-316, 2009.

〔6〕 Ellen. Riloff, Siddharth. Patwardhan, and Janyce. Wiebe, "Feature subsumption for opinion analysis," In Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing, Sydney, pp. 440-448, 2006.

〔7〕 A. Esuli, and F. Sebastiani, "SentiWordNet: A publicly available lexical resource for opinion mining," In Proceedings of LREC-06, the 5th Conference on Language Resources and Evaluation, Genova, IT, pp. 417-422, 2006.

〔8〕 N. Jakob, I. Gurevych, "w," In Proceedings of the ACL 2010 conference short papers, pp. 263-268, 2010.

〔9〕 AM. Popescu, and O. Etzioni, "Extracting product features and opinions from reviews," In Proceedings of Human Language Technology Conference on Emprirical Methods in Natural Language Processing, Vancouver, pp. 339-346, 2005.

〔10〕 Marie-Catherine Marneffe, and Christopher D. Manning, "Stanford typed dependencies manual," Stanford Parser, 2011.

저 자 소 개

소 윤 규

2011 : 한양대학교 컴퓨터공학과 공학사
현 재 : 한양대학교 컴퓨터공학과 석사과정
관심분야 : 자연어처리, 의견검색
Email : kim@gmail.com

김 한 우

1980 : 한양대학교 전자공학 공학박사
현 재 : 한양대학교 컴퓨터공학과 교수
관심분야 : 정보처리, 자연어처리, 기계번역
Email : kimhw@hanyang.ac.kr

정 성 훈

2009 : 한양대학교 컴퓨터공학과 공학석사
현 재 : 한양대학교 컴퓨터공학과 박사과정
관심분야 : 의견검색, 질의응답
Email : ishkkman@gmail.com

김 동 주

2007 : 한양대학교 컴퓨터공학과 공학박사
현 재 : 안양대학교 교양대학 교수
관심분야 : 맞춤법검사, 기계번역, 한국어정보처리, 의견검색, 감정인식
Email : djkim@anyang.ac.kr