韓國컴퓨터情報學會 論文誌
第15卷 第1號, 2010. 1.

2010-15-1-4-1

의견정보 모니터링을 위한 웹 마이닝 시스템에 관한 연구

주 해 종*, 홍 봉 화**, 정 복 철***

A Study on Web Mining System for Real-Time Monitoring of Opinion Information Based on Web 2.0

Hae Jong Joo *, Bong Hwa Hong **, Bok Cheol Jeong ***

요 약

최근에 인터넷 사용이 점차 활발해 짐에 따라, 다른 사람들이 인터넷 상에 올려놓은 의견정보를 참조하고자 하는 수요가 높아지고 있다. 하지만, 이러한 인터넷 상에 존재하는 의견들은 개개의 웹사이트들에만 존재하여, 이러한 의견정보들을 사용하고자 할 경우에는 사용자가 일일이 이러한 개개의 모든 웹사이트를 수동으로 찾아보아야 하는 번거로움이 존재하는 문제점이 있다. 본 논문은 웹 콘텐츠에서의 통계기반 웹 마이닝(Web Mining)을 통한 의견 추출 및 분석 시스템에 관한 것으로, 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 문서에서 사용자 의견정보들을 자동으로 추출 및 분석한다. 또한, 긍정/부정 의견별로 실시간으로 검색 및 통계를 확인할 수 있는 의견정보 검색 서비스를 간편하게 제공할 수 있으며, 의견정보 검색 사용자들은 특정 키워드에 대하여 다른 사용자들의 의견정보를 손쉽게 실시간으로 검색 및 모니터링(Monitoring)할 수 있는 시스템이다. 제안한 기법들은 기존의 다른 기법들과의 비교 실험을 수행하여 실제 성능이 우수함을 증명하였다. 성능 평가는 긍정/부정 의견정보를 추출하는 기능의 성능 평가를 실시하였다. 그 적용 사례로 대표적인 영화 리뷰 문장 실험 데이터를 대상으로 실험하고 그 결과를 분석하였다.

Abstract

As the use of the Internet has recently increased, the demand for opinion information posted on the Internet has grown. However, such resources only exist on the website. People who want to search for information on the Internet find it inconvenient to visit each website. This paper focuses on the opinion information extraction and analysis system through Web mining that is based on statistics collected from Web contents. That is, users' opinion information which is scattered across several websites can be automatically analyzed and extracted. The system provides the opinion information search service that enables users to search for real-time positive and negative opinions and check their statistics. Also, users can do real-time search and monitoring about other opinion information by putting keywords in the system. Proposed technologies proved to have outstanding capabilities in comparison to existing ones through tests. The capabilities to extract positive and negative opinion information were assessed. Specifically, test movie review sentence testing data was tested and its results were analyzed.

• 제1저자 : 주해종
• 투고일 : 2009. 11. 06, 심사일 : 2009. 11. 11, 게재확정일 : 2010. 01. 26.
* 동국대학교 산학협력중심대학육성사업단 교수 ** 경희사이버대학교 정보통신학과 교수
*** 경희대학교 교양학부 교수

▸ Keyword : 웹 의견정보(Web opinion information) 모니터링(Monitering), 웹 의견정보 자동 추출(Web pinion information automatic extraction), 웹 마이닝(Web mining), 웹 의견정보 자동 분석(Web pinion information automatic analysis)

Ⅰ. 서 론

최근에 인터넷 사용이 점차 활발해 짐에 따라, 많은 사람들이 인터넷에서 예컨대, 블로그(Blog), 위키(Wiki)와 같은 매체를 통해서 자신의 의견을 표현하고 있는 추세이다[1]. 또한, 특정한 정보의 가치를 평가할 때, 이러한 다른 사람들이 인터넷 상에 올려놓은 의견 정보를 참조하고자 하는 수요가 높아지고 있다.

하지만, 이러한 인터넷 상에 존재하는 의견들은 개개의 웹사이트들에만 존재하여, 이러한 의견 정보들을 사용하고자 할 경우에는 사용자가 일일이 이러한 개개의 모든 웹사이트를 수동으로 찾아보아야 하는 번거로움이 존재한다.

본 논문 이러한 문제점을 해결하기 위하여 실시간 의견정보 모니터링을 위한 웹 콘텐츠 마이닝 시스템을 제안한다. 제안하는 시스템은 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 콘텐츠에서 사용자 의견정보들을 자동으로 추출 및 분석함으로써, 긍정/부정 의견별로 검색 및 통계를 확인할 수 있는 의견 정보 검색 서비스를 제공한다. 그 결과 의견 검색 사용자들은 특정 키워드에 대하여 다른 사용자들의 의견정보를 손쉽게 한눈에 검색 및 모니터링하는 시스템을 용이하게 사용할 수 있으며, 웹 콘텐츠에서의 실시간으로 의견정보를 자동으로 추출 및 분석하는 기능을 제공받는다.

이 논문의 구성은 다음과 같다. 2장에서는 본 논문의 이론적 고찰을 위해 기존 웹 마이닝 기법, 의견 추출 기법의 이론적 배경을 살펴보고 문제점을 살펴본다. 3장에서는 인터넷 상에서 의견정보를 수집 및 분석하여 모니터링할 수 있는 웹 콘텐츠 마이닝 시스템의 설계와 구현 방법을 제안한다. 4장에서는 본 제안 시스템의 성능분석을 위하여 의견정보 추출 및 분석 기능의 성능평가를 제시한다. 마지막으로 5장에서는 결론을 맺는다.

Ⅱ. 관련 연구

2.1 웹 마이닝 기법

웹 마이닝(Web Mining)은 웹에서 발생되거나, 웹상에 존재하는 모든 데이터를 대상으로 하며, 이러한 데이터를 기반으로 데이터마이닝 기법을 적용하여 유용한 정보를 추출, 분석하는 과정을 말한다. 즉, 데이터마이닝 기술을 웹이라는 거대한 데이터 집합에 적용한 어플리케이션이다[2,4,5]. 이러한 웹 마이닝은 웹 문서와 서비스로부터 자동으로 정보를 발견하고 추출하기 위해 데이터마이닝 기법을 이용하는 것이다. 즉, 웹 데이터로부터 미리 알려지지 않은 유용한 정보나 지식을 발견하는 과정이라고 정의할 수 있다.

웹 마이닝의 연구 분야는 정보 검색(Inform ation Retrieval) 혹은 정보 추출(Information Extraction)의 분야에서 연구하고 있는 많은 부분을 공유하고 있다[3,4,5].

2.2 의견 추출 기법

의견 분류를 문서나 문장 단위에서 좀더 세부적으로 분류하는 단위는 구나 단어 단위에서 의견을 분류하는 연구이다. 구나 단어 단위에서 의견을 분류하는 연구는 초기에는 규칙 기반의 방법으로 연구되었으며, 이후에 구나 단어의 주변 정보를 학습 하여 구나 단어의 극성을 판단하는 기계 학습 방법이 연구 되었다[7,8].

2.2.1 규칙 기반 모델

규칙 기반 방법에 기반하여 구 단위로 의견을 추출하는 연구로는 Nasukawa와 Zhongchao Fei의 연구가 이루어졌다[9,10]. 규칙기반에서는 각각의 단어를 그 단어의 품사 정보와 극성 정보를 합쳐서 태그를 부여하였다[11,12]. 극성 정보는 good, bad, neutral 이 3가지로 정했고, 대상이 되는 단어 품사는 형용사, 명사, 부사, 동사로 정하였다. 긍정적인 동사와 부정적인 동사에 대해서 태그를 달았다. 이때에는 해당 단어 하나에 대해서 태그를 부착하거나 그 단어와 함께 하나의 표현을 이루면서 특정 극성을 나타내는 구에 대해서도 태그를 부여한다.

2.2.2 기계 학습기반 모델

이전의 규칙 기반 방법에서는 의견 표현 자원을 수동으로 구축하는 방법에 주로 초점이 맞추어져 있었다. 기계학습 방법은 문장 내에서 긍정 또는 부정 표현 부분에 태깅된 코퍼스(Corpus)를 이용하여 기계학습을 수행한다.

의견 태깅된 코퍼스가 자동 구축된 후에는 이 코퍼스를 이용하여 단어/구 단위의 의견 분류를 위한 기계학습을 하게 된다. 의견 분류를 위한 기계학습으로는 HMM(Hidden Markov Model) [16]을 사용한다.

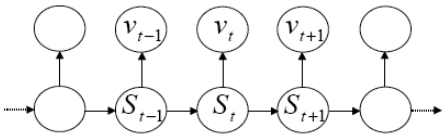

그림 1. HMM에서의 확률적인 Parameter 들
Fig. 1. Probability Parameters of HMM

2.2.3 이전 방법의 장단점

이전 방법에서 규칙 기반 방법은 일치 하는 패턴에 대해서는 높은 정확도를 보이는 장점을 가지고 있다. 그래서 의견 패턴만 정교하게 많은 양을 구축할 수 있다고 한다면 좋은 접근 방법이 될 수 있다. 하지만 규칙 기반 방법이 꾸준히 지적되어 온 문제와 같이, 여기서도 규칙 기반 방법으로 할 경우에는 이미 구축한 패턴이 약간 변형된 형태로 나온 경우에 취약점을 보여서 재현율이 크게 떨어지는 단점과, 주변 문맥 정보를 학습에 반영하지 못하는 한계를 지니고 있다. 이 의견 패턴을 다른 모든 도메인과 언어권에 대해서 구축하고 유지보수 하는 일은 수많은 인력과 시간이 필요한 어려운 작업이다.

단어/구 단위의 의견 분류를 위한 규칙 기반 방법과 기계학습 기반의 방법 모두 사람이 수동으로 의견 자원을 구축해야 하는 어려운 점이 있다. 이러한 의견 자원이 구축되어 있어야만 단어/구 단위의 의견 분류가 가능하기 때문에, 단어/구 단위의 의견 분류에서 가장 중요한 문제 중 하나가 의견 패턴 구축 또는 의견 표현이 태깅된 코퍼스 구축이라고 볼 수 있다.

III. 웹 의견정보 모니터링을 위한 웹 마이닝 시스템

3.1 시스템 개요

의견정보 실시간 모니터링을 위한 웹 콘텐츠 마이닝 시스템은 웹 콘텐츠에서 의견정보를 자동으로 추출 및 분석하기 위한 시스템으로, 그 플랫폼은 그림 2와 같다. 제안 시스템은 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 문서에서 사용자 의견정보들을 자동으로 추출 및 분석함으로써, 긍정/부정 의견별로 검색 및 통계를 확인할 수 있는 의견정보 검색 서비스를 제공하는 시스템이다.

그림 2의 제안 시스템은 크게 데이터 수집 처리, 의견/비의견 자동 구축, 의견 정보 자원, 인덱싱 처리, 의견 인덱싱 정보 자원, 의견표현 기계 학습, 의견 검색 처리 및 사용자 단말 등을 포함하여 이루어진다.

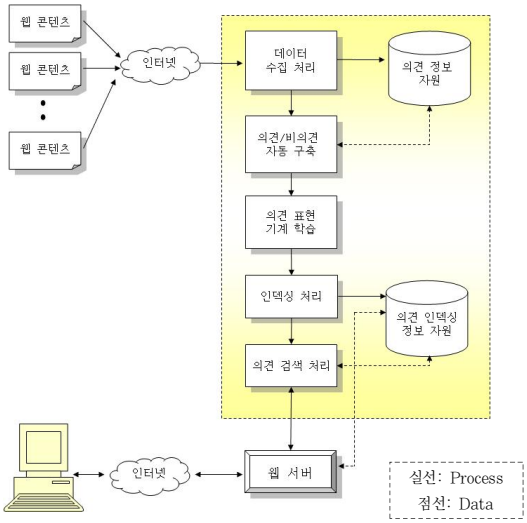

그림 2. 의견 정보 자동추출 웹 마이닝 시스템 플랫폼
Fig. 2. Web Mining System Platform for Opinion Information Extraction

3.2 데이터 수집 처리

데이터 수집 처리는 인터넷 상에 존재하는 다양한 웹 콘텐츠들을 수집하는 기능을 수행한다. 즉, 데이터 수집 처리는 인터넷 상에 존재하는 각 웹사이트(Web Site)들의 HTML(Hyper Text Markup Language) 정보를 실시간으로 다운로드 받게 된다. 또한, 데이터 수집처리는 상기와 같이 다운로드(Download) 받은 웹 콘텐츠에서 필요한 정보들 예컨대, 텍스트(Text), 이미지(Image) 또는 비디오(Video) 등의 정보들 중 적어도 어느 하나의 정보 데이터를 추출하여 별도의 데이터 저장 모듈에 저장시킬 수 있다.

데이터 수집 처리는 표 1과 같이, 의견정보 데이터(즉, 일반 문장/문서 데이터와 이에 대한 긍정/부정 평가가 매겨진 정보 데이터)를 포함하는 웹 콘텐츠들을 선별하여 수집할 수도 있다. 데이터 수집 처리를 통해 수집되는 대상 데이터는 표 1에 나타난 바와 같이, 의견정보 데이터 즉, 일반 문장/문서 데이터와 이에 대한 긍정/부정 평가가 매겨진 정보 데이터들이다. 이때, 상기 긍정/부정 평가는 일정 범위내의 점수로 표현되어지거나, 별표(★)나 기타 기호들을 이용하여 다양하게 평가될 수 있다. 본 논문에서는 이렇게 다양한 방식으로 표현되는 긍정/부정 평가는 모두 동일한 점수 범위로 재계산되어서 사용된다.

표 1. 의견 정보 데이터
Table 1. Opinion Information Data

표현	점수	의견 내용
★★★★★	10	재미있어 신고
★★★★★	10	'똑똑한' 사람들이 살아있는 이야기 신고
★★★★☆	8	현명한 사람들의 일상 뜯어고치기! 신고
★★★★★	9	삼촌의 매력에 흠뻑... 신고
★★★★☆	8	평범한 사람들의 이야기 신고
★★★★★	10	연기도 좋고 내용도 잼있고 가슴 훈훈해지는 사랑 이야기 신고
★★★★★	10	정말 감동할만한 이야기이었어요. 신고
★★★★★	10	보는 내내 가슴 따뜻해지는 영화였습니다. 재미도 있고요 신고
★★★☆☆	6	훈훈하고 코믹하고.. 영화 넘 짧은거 같은데.. 신고
★★☆☆☆	5	돌고 돌고 돌아 결국은 뻔한 이야기. 신고

이를 구체적으로 설명하면, 본 논문의 실시 예에서 사용하는 점수 범위가 a~b 라고 하였을 때에 수집한 데이터의 점수 범위가 c~d 라고 한다면, 해당 수집 점수 x는 식 1과 같이 변화된다.

$$PolarityScore(x) = (a-1) + \frac{x-c+1}{d-c+1} \times (b-a+1) \quad \text{(식 1)}$$

예를 들어, 본 논문은 1~10점 사이의 점수를 사용하고(10점에 가까울수록 긍정), 수집한 데이터는 1~5점 사이의 점수를 사용하는 경우에, 수집한 데이터가 2점이라고 한다면, 식 2와 같이 계산되어 진다.

$$PolarityScore(2) = (1-1) + \frac{2-1+1}{5-1+1} \times (10-1+1) = 4 \quad \text{(식 2)}$$

데이터 수집 처리에 의해 수집된 데이터는 표 1과 같이 본 논문에서 사용하는 점수로 변환된 의견 점수 집합 {(데이터, 점수), (데이터, 점수), ...(데이터, 점수)}으로 표현하기 위해, 표 2와 같은 의견 정보 자원 데이터 구조에 저장된다. 표 2에서는 의견 정보 자원 데이터 구조의 필드명과 해당 필드에 대한 데이터 타입, 필드에 대한 설명을 나타내고 있다.

표 2. 의견 정보 자원 데이터 구조
Table 2. Resource Data Structure of Opinion Information

id	user_id	date	topic	sentence	polarity	.....
bigserial (PK)	char varying (200)	bigint	char varying (200)	char varying (1000)	char varying (10)	.....
문장 식별자	사용자 식별자	문장 날짜	분류	문장	산출 점수	.....

본 논문에서는 의견 정보 "단어/구"가 태깅된 학습 코퍼스(Corpus)를 자동 구축하는 것을 그 목표로 한다. 이렇게 자동 구축한 코퍼스를 이용하여 기계 학습 방법을 통해 의견 정보 "단어/구"를 자동 분류하게 된다. 이때 데이터 수집 처리는 의견 정보 "단어/구" 가 태깅된 코퍼스를 자동 구축하기 위해서 식 1과 표 2의 의견 정보 데이터 구조를 이용하여 인터넷에서 쉽게 구할 수 있는 문장 단위 긍정/부정 의견 정보가 표현된 데이터를 수집하게 된다.

3.3 의견/비의견 자동 구축

식 2와 같이 규칙기반에 의해 단순히 긍정/부정 문서에 나오는 횟수를 이용하는 방법은 1~10 점과 같은 점수 형태의 데이터에는 부정확하게 된다. 또한 긍정문서와 부정문서의 개수가 다른 경우에 절대적인 등장 횟수를 사용하게 되는 경우 해당 데이터 셋 집합 크기가 큰 쪽으로 치우친 점수가 나오는 문제점이 있다.

본 절에서 의견/비의견 정보 자원을 자동으로 구축하는 방법은 가능한 의견 정보 표현을 모든 사전기반 N-Gram 분석기로 생성한 후에 여기서 사용할 후보를 그 단어의 긍정/부정 확률과 의견 문서에서 나올 확률을 복완법(interpolation)을 통하여 자동으로 구하는 특징을 지니고 있다. 긍정/부정 확률과 의견 문서에서 나올 확률을 구하는 과정에서 1~10점과 같이 여러 점수 집합의 의견 강도를 반영하였으며, 특정 점수 집합의 데이터 크기 자체가 커져서 점수 치우쳐 지는 문제를 해결하기 위해서 정규화 방법도 제안하였다.

3.3.1 제안 방법의 단어 점수 산출 방법

의견 정보 단어 자원을 자동 구축하기 위해서 본 절에서는 문장 단위로 의견이 표시된 데이터를 이용한다. 그 후, 형태소 단위로 문장을 나눈 후에는 각 형태소의 N-Gram 에 대한 점수를 구하게 된다.

$Freq(W_j, S_i)$는 단어 Wj 가 점수 집합 Si에서 나타나는 횟수를 나타낸다. 따라서 단어 "영화"가 10점 점수가 9번 ($Freq(\text{영화}, s_{10}) = 9$) 나왔고, 1점 점수가 1번 ($Freq(\text{영화}, s_1) = 1$) 나왔다고 가정할 때, 긍정문서와 부정문서에서의 "영화"단어에 대한 빈도수를 이용한 극점 점수는 식 3과 같다.

$$Score(\text{영화}) = \log\left[\frac{Freq(\text{영화}, s_{10})+1}{Freq(\text{영화}, s_1)+1}\right] = \log\left[\frac{9+1}{1+1}\right] = 1.60 \quad \text{(식 3)}$$

식 3은 2.2절의 이전 방법을 바로 적용했을 때 문제가 발생하는 예이다. 이전 연구에서는 긍정/부정 데이터의 크기가 같은 상황에서 사용하였다. 만약에 각 데이터 크기가 식 4와 같

이 계산하게 되면 문제가 발생한다. "영화" 라는 단어가 10점 점수 집합에서 9번, 1점 점수 집합에서 1번 나온 경우 단순히 위 수식으로 계산을 하게 되면 1.6 점이라는 아주 긍정적인 점수를 받게 된다. 하지만 위의 예를 보면, 10점 점수 집합 자체가 크기 때문에 그 점수 집합에서는 각 단어들이 더 많이 등장함을 알 수 있다. 즉, 단어의 극성 점수를 구할 때 각 점수 집합의 크기가 다른 경우에 문제가 발생한다.

위 예제처럼 영화 리뷰에서 "영화" 라는 단어는 모든 점수 집합에서 공통적으로 많이 나오는 단어이다. 이때 단순히 10점 점수대의 절대적인 크기 자체가 큰 상황에서 절대적인 값을 사용하는 경우에 그 점수가 큰 점수에 지나치게 편중되는 문제가 생긴다. 따라서 각 점수 집합의 크기가 다른 경우에는 단순히 점수의 평균을 취하지 말고, 적절한 정규화가 필요하게 된다. 식 4는 절대적인 빈도수를 상대적인 확률로 변환시키기 위한 수식이다.

$$Freq(w_j, s_i) \rightarrow \frac{Freq(w_j, s_i)}{\sum_{w_k \in W} Freq(w_k, s_i)} = P(w_j | s_i) \quad \text{(식 4)}$$

식 5에서 $Freq(W_j, S_i)$는 단어 Wj 가 점수 집합 Si에서 나타나는 횟수를 나타내며, $\sum_{w_k \in W} Freq(W_k, S_i)$는 모든 점수 집합에서 단어 Wk가 나타나는 횟수를 더한 값으로서 결국 전체 데이터에서 Wk가 나타나는 횟수를 의미한다. 따라서 $P(w_j | s_i)$는 절대적인 빈도수를 상대적인 확률로 변환한 값이 된다. 이를 기반으로 절대적인 빈도수를 상대적인 확률 $P(w_j | s_i)$를 가지는 상대적인 값으로 정규화 하면, 식 5와 같다.

$$PolarityScore(w_j) = \frac{\sum_{s_i \subset S} [i \times P(w_j | s_i)]}{\sum_{s_i \subset S} P(w_j | s_i)} \quad \text{(식 5)}$$

식 5에 따라서 이전에 절대 값의 평균을 취했던 부분을 정규화된 값들의 평균으로 바꿈으로 특정 점수 집합에 치우치지 않은 정규화된 의견 점수를 얻을 수 있게 된다.

3.3.2 주관적 점수를 이용한 산출 방법

단어 자원을 구축할 때에는 극성 점수뿐만 아니라 단어의 주관성도 계산을 해야 한다. 이때 주관성에 영향을 미치는 요소를 극성 점수를 계산 하는 경우와 같이 그 단어 자체의 생성 확률보다는 그 단어의 품사 정보의 생성 확률로 해야 한다. 단어 자체의 생성 확률로 할 경우에는 학습 데이터에서 나오지 않은 여러 고유대명사, 외래어 기타 단어들에 대해서는 취약하게 되고, 의견 표현에서는 잘 등장하는 특정 품사 조합이 존재하기 때문에 품사 정보가 유용하게 사용될 수 있기 때문이다. 단어의 주관성은 앞에서 제안한 단어의 점수를 계산 하는 방법을 이용하여 똑같이 계산 할 수 있으며, 다만 이때에는 계산하는 대상 데이터를 의견 데이터의 품사 데이터와 의견이 아닌 데이터의 품사 데이터를 각각 긍정(10점), 부정(1점) 데이터라고 보고 계산을 하게 되면 위와 같이 그 단어의 주관성 점수를 구할 수 있게 된다.

식 6은 주관적 점수 산출 방법으로서, pos_j는 단어 w_j의 품사 정보를 나타내고, s_i는 각 점수 집합을 나타낸다. 주관적 점수를 산출할 때 s_1은 의견이 포함되지 않은 데이터 집합, s_{10}은 의견을 포함하는 데이터 집합으로 보고 계산을 하게 된다.

$$\text{주관적점수}(pos_j) = \frac{\sum_{s_i \subset S} [i \times P(pos_j | s_i)]}{\sum_{s_i \subset S} P(pos_j | s_i)} \quad \text{(식 6)}$$

식 7은 본 논문의 의견 정보 단어를 자동 구축하는 제안 방법으로, 극성 점수와 주관적 점수 두 가지 점수를 이용하여 복완법을 사용하여 단어의 OpinionScore 를 구하게 된다. 이 OpinionScore 는 그 단어가 얼마나 의견 단어인지를 나타내는 점수로서, 특정 점수 이하의 단어는 의견 단어로 사용하지 않게 된다.

$$OpinionScore(w_j) = \left| PoarityScore(w_j) - \frac{1}{2} \times \max(S) \right| \times \alpha + \left[SujectiveScore(pos_j) - \frac{1}{2} \times \max(S) \right] \times (1-\alpha) \quad \text{(식 7)}$$

위 수식에서 max(S) 는 점수 집합에서 최대 점수 집합을 의미한다. $-\frac{1}{2} \times \max(S)$ 부분과 PolarityScore 부분에 절대값을 취한 이유는 PolarityScore에서 부정적인 단어들도 의견 점수를 높여주기 위함이다.

3.3.3 주관적 단어 선별

표 3은 Unigram, Bigram 으로 의견 단어 자원을 영화 리뷰에서 자동 구축한 예이다. 각 단어의 아래에 있는 수치는 그 단의 의견 점수로서 점수 범위는 1~10 점이면, 1점 쪽이 부정, 10점 쪽이 긍정에 가까운 점수이다. 결과를 보면, 실제 사람들이 쓰는 긍정, 부정 표현에 가까운 것을 볼 수 있다. 형용사 뿐만 아니라, 명사나 특정 대상도 비유적인 표현으로 의견을 나타내기 위해서 쓰임을 볼 수 있다.

이처럼 의견 표현은 다양한 비유, 반어, 비교 표현이 존재

하기 때문에 일반적인 단어의 긍정/부정 점수가 매겨져 있는 단어 자원으로는 부족한 점이 많다. 따라서 위와 같은 자동적으로 일반 적인 의견 표현뿐만 아니라 해당 도메인에서만 사용되는 각종 의견 표현까지 구축 하는 작업은 매우 중요함을 알 수 있다.

표 3. 의견 단어 자동 구축 결과
Table 3. Result of Opinion Automatic Construction

	긍정적 단어		부정적 단어	
Unigram	짱짱/XR 9.908	설레이/VV 9.757	제기랄/IC 1.566	할인/NNG 1.518
Bigram	완전/NNG 강추/NNP 9.904	최고/NNG 입니다/EF 9.911	할인/NNG 카드/NNG 1.021	푹/MAG 자/VV 1.286

3.4 의견 표현 기계 학습

의견/비의견 태깅 코퍼스가 자동 구축된 후에는 이 코퍼스를 이용하여 단어/구 단위의 의견 분류를 위한 기계학습을 하게 된다. 의견 분류를 위한 기계학습으로는 2장의 관련 연구에서 기술한 HMM을 사용한다. 그러나 HMM이 실제 적용되기 위해서 해결되어야 하는 평가 문제(Evaluation Problem), 디코딩 문제(Decoding problem), 그리고 학습 문제(Estimation problem)가 있다[16].

첫 번째로 해결해야 하는 평가 문제는 관찰된 심볼의 시퀀스 O=O1O2...Or와 모델 $\lambda=(A,B,\pi)$ 가 주어졌을 때, 그 모델에서 관찰된 데이터 O 의 확률 $P(O|\lambda)$ 를 어떻게 구할 것인가? 하는 문제이다. 두 번째로 해결해야 하는 디코딩 문제는 관찰된 심볼의 시퀀스 O=O1O2...Or와 모델 $\lambda=(A,B,\pi)$ 가 주어졌을 때 최적의 상태 전이 시퀀스 $Q=q_1q_2..q_t$ 는 무엇인가? 하는 문제이다. 세 번째로 해결해야 하는 학습문제는 가장 큰 $\pi=\{\pi_i\}$ 를 나타내는 모델 파라미터 $\lambda=(A,B,\pi)$ 를 결정하는 문제이다.

위의 세 가지 문제는 각각 Forward 알고리즘, Viterbi 알고리즘, Baum-Welch 알고리즘으로 해결이 가능하다. 본 논문에서는 모델 파라미터인 상태 전이 확률, 관찰 확률, 초기 상태 확률은 의견/비의견 자동 구축 모듈에서 수집한 태깅 코퍼스로부터 얻는다.

3.5 인덱싱 처리

인덱싱 처리는 의견/비의견 자동 구축으로부터 구분된 의견 문장의 언어적인 자질별로 해당 웹 콘텐츠의 의견 정보들이 의견 인덱싱 정보 자원에 저장되도록 인덱싱(Indexing)하는 기능을 수행한다. 여기서, 의견 인덱싱 정보 자원은 인덱싱 처리를 통해 인덱싱된 각 의견 문장의 언어적인 자질별 해당 의견 문장의 요약정보 및 해당 웹 콘텐츠의 기본 및 의견 정보들이 데이터베이스화하여 저장되는 기능을 수행한다.

표 4. 의견 인덱싱 정보 자원 데이터 구조
Table 4. Resource Data Structure of Opinion Indexing Information

id	comm entct	date	snip pet	data	pola rity	topic	url	...
big serial (PK)	char varying (50)	bigint	char varying (200)	char varying (2000)	char varying (10)	char varying (10)	char varying (200)	...
문장 식별자	콘텐츠 설명	문장 날짜	의견 정보	문장	산출 점수	분야	url 정보	...

의견/비의견 자동 구축으로부터 구분된 의견 문장의 언어적인 자질별로 해당 웹 콘텐츠의 의견 정보들이 의견 인덱싱 정보 자원에 저장하기 위한 의견 인덱싱 정보 자원은 표 4와 같은 데이터 구조에 저장된다. 표 4에서는 의견 인덱싱 정보 자원 데이터 구조의 필드명과 해당 필드에 대한 데이터 타입, 필드에 대한 설명을 나타내고 있다. 인덱싱 과정은 검색 속도를 개선하기 위하여 대 분류 인덱싱 과정과 각 문서에 대한 색인어와 내용 정보를 인덱싱하여 실제 정보 검색과정에서 사용하기 위한 상세 분류 인덱싱으로 구성된다. 대 분류 인덱싱은 전문 용어를 포함하는 문서를 나타낸다. 다음의 그림 3은 대 분류 인덱스의 구성을 나타낸다.

대분류 테이블

인덱스키	문서 번호	빈도 수	제목 유무
영화	53	12	1
IT	85	6	1
영화	97	48	0
영화	11	4	0
환경	99	2	0

그림 3. 대분류 인덱싱
Fig. 3. Large Grouping Indexing

상세 분류 인덱싱은 사용자가 제시한 검색어를 포함하는 실제 문서를 검색하기 위해 문서 테이블을 구성하는 과정이다. 문서 테이블에는 제목정보, 파일명(저장경로), 문서 내용과

문서 내용에 포함된 전공 관련 키워드와 같은 정보가 저장된다. 여기서 영화 관련 키워드는 형태소 분석 시 영화 리뷰 문장을 조회하여 영화 관련 전문 용어들을 추출하여 저장하게 된다. 다음의 그림 4는 상세 분류 인덱싱의 구성을 나타낸다.

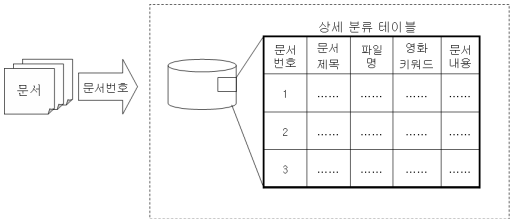

그림 4. 상세 분류 인덱싱
Fig. 4. Detail Grouping Indexing

3.6 의견정보 검색 처리

의견정보 검색 처리는 웹 서버를 통해 전송된 사용자의 특정 의견정보 검색 키워드 또는 타입(Type) 정보를 제공받아 인덱싱 처리 또는 의견 인덱싱 정보 저장 자원과 연동하여, 특정 의견검색 키워드 또는 타입 정보와 관련된 웹 문서의 인덱싱 정보들을 검색하여 해당 사용자 단말기로 전송되도록 웹 서버로 전달하는 기능을 수행한다.

그림 5는 의견정보를 실시간 모니터링할 수 있는 시스템의 메인 화면에서 키워드 "명지대"를 등록하여 얻은 긍정/부정 의견이다. 좌측의 부문은 키워드를 등록하는 창이고, 중단에 있는 것은 현재 인터넷 상에서 각 분야별로 이슈가 되는 대상들을 추출하여, 보여주는 화면이다. 등록된 키워드 검색은 기간별, 관심 분야별로 검색 가능하며, "긍정/부정 의견"과 "호감도 변화", 그리고 "관심도 변화"에 따라 내용을 볼 수 있다.

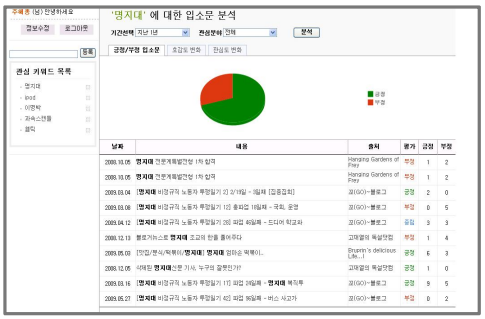

그림 5. 의견 정보 모니터링 메인 화면
Fig. 5. Main Screen for Opinion Information Monitoring

Ⅳ. 실험 및 성능 평가

4.1 실험 환경

본 장에서는 의견정보 실시간 모니터링을 위한 웹 콘텐츠 마이닝 시스템의 성능을 평가하기 위하여 긍정/부정 의견정보를 추출하는 기능에 대해 실험과 성능 평가를 실시한다. 그 적용 사례로 대표적인 영화 리뷰 문장 실험 데이터를 대상으로 실험하고 그 결과를 분석한다.

본 실험에서는 네이버 영화의 영화 리뷰 문장을 학습데이터로 사용한다. 점수 범위는 1, 2, 3점의 60,000 문장과 8, 9, 10의 60,000 문장으로 총 120,000 문장이다. 모두 영화에 대한 평가 문장들로 이루어졌다. 그리고 단어의 주관성 점수를 구하기 위해서 의견 포함되지 않은 데이터로도, 영화 줄거리 198,229 문장을 네이버 영화에서 수집하여 사용하였다.

4.2 성능 평가

본 논문에서 성능 평가할 기준은 지도(Super vised) 학습 방법의 HMM, CRF이다. HMM은 2.2절에 제시한 문제점을 해결한 의견 표현 기계 학습 방법이며, CRF(Conditional Random Fields)는 현재 단어를 중심으로 해서 이전 4개 단어와 이후 4개 단어를 자질로 반영한 모델이다[16].

지도 학습 방법을 사용하기 위해서는 문장에서 의견 표현이 태깅된 데이터가 필요한데 학습데이터로 사용되는 데이터는 문장 단위로만 긍정/부정이 표시된 데이터이다.

표 5. 코퍼스 성능 평가 결과
Table 5. Corpus Experiment's Result

	Precision(%) Exact/Overlap	Recall(%) Exact/Overlap	F-Measure(%) Exact/Overlap
단어 극성 적용	33.43/30.42	56.55/75.67	42.02/43.40
문장 극성 적용	**38.00/34.02**	**58.55/85.55**	**46.09/48.68**

표 5에서 자동 구축한 의견 구가 태깅된 데이터의 성능을 살펴보면, 기존의 연구방법인 단어가 가지는 극성을 바로 적

용한 경우에 비해서, 본 논문에서 제시한 바와 같이 문장이 가지는 극성으로 자동 구축한 코퍼스가 Precision과 F-Measure에서 "Exact"와 "Overlap"이 각각 4% 정도 더 높은 성능을 보였다. 그리고 Recall에서 "Overlap"이 약 10% 정도 더 높은 성능을 보였다. 이는 단어 자체가 가지는 극성보다, 그 문장이 가지는 극성이 문장 안에서 단어가 가지는 극성을 결정 하는데 더 큰 역할을 함을 알 수 있다.

V. 결 론

인터넷 상에 존재하는 의견들은 개개의 웹사이트들에만 존재하여, 이러한 의견정보들을 사용하고자 할 경우에는 사용자가 일일이 이러한 개개의 모든 웹사이트를 수동으로 찾아보아야 하는 번거로움이 존재한다. 이 논문에서는 이러한 문제점을 해결하기 위하여 의견정보를 실시간으로 모니터링하는 웹 콘텐츠 마이닝 시스템을 제안하였다. 제안한 시스템은 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 콘텐츠에서 사용자 의견정보들을 자동으로 추출 및 분석함으로써, 긍정/부정 의견별로 검색 및 통계를 확인할 수 있는 의견 검색 서비스를 제공한다.

제안하는 기법들은 다른 기법들과의 비교 실험을 수행하여 실제 성능이 우수함을 증명하였다. 성능 평가는 긍정/부정 의견정보를 추출하는 기능의 성능 평가를 실시하였다. 그 적용 사례로 대표적인 영화 리뷰 문장 실험 데이터를 대상으로 실험하고 그 결과를 분석하였다.

성능 평가 결과, 의견정보 추출 및 분석기능의 기법에서 본 논문에서 제시한 문장이 가지는 극성으로 자동 구축한 코퍼스가 Precision과 F-Measure에서 "Exact"와 "Overlap"이 각각 4% 정도 더 높은 성능을 보였으며, Recall에서 "Overlap"이 약 10정도 더 높은 성능을 보였다.

제안 시스템의 기대효과는 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 사용자 의견정보들을 자동으로 추출 및 분석하여 긍정/부정 의견별로 검색 및 통계를 확인할 수 있도록 의견 검색 서비스를 제공해 줌으로써, 사용자들은 특정 키워드에 대하여 다른 사용자들의 의견정보를 손쉽게 한눈에 검색 및 모니터링 할 수 있으며, 기존에 다른 사용자들의 의견을 검색하기 위해서 들였던 많은 시간을 크게 단축시킬 수 있다는 것이다.

향후의 과제로는 인터넷 상에서 언어장벽을 해소시켜 모국어로 다른 나라 정보를 모니터링 할 수 있게 하기위한 다국어(한, 중, 일, 영) 검색 및 기계번역 기능을 추가하여 완전한 모니터링 검색엔진을 위한 웹 컨텐츠 의견 검색 시스템을 만드는 것과 의견 모니터링에 대한 신뢰성을 검증하는 것이 필요하다.

참고문헌

[1] 주해종 · 박영배, "모니터링 검색엔진을 위한 웹 콘텐츠 마이닝 시스템 설계," 한국통신학회 논문지 제 34권, 제 2호, 53-60쪽, 2009년 2월.

[2] P. Adriaans, D. Zantinge, "*Data Mining*," Addison Wesley Longman, England, 1996.

[3] R. Kosala, H. Blockeel, "Web Mining Research: A Survey", ACM SIGKDD, July 2000.

[4] C. H. Lee, H. C. Yang, "A Web Text Mining Approach Base on Self-Organizing Map," In Proceedings of the 2nd International Workshop on Web Information and Data Management, WIDM'99, Kansas City, MO, USA, 1999, 59-62.

[5] M. Mulvenna, S. Anand, A. Büchner, "Personalization on the Net using Web Mining," Communications of the ACM, Vol. 43, No. 8, Aug. 2000.

[6] Dagan, I., Church, K.W., and Gale, "Robust bilingual word alignment for machine aided translation," In Proceedings of the workshop on Very Large Corpora, pp. 1-8, 1993

[7] Lee, J. S. and K. S. Choi, "English to Korean Statistical transliteration for information retrieval," Journal of Computer Processing of Oriental Languages, 12(1):17-37, 1998

[8] Kang B.J. and K-S. Choi, "Automatic Transliteration and Back-transliteration by Decision Tree Learning," In Proceedings of LREC'2000, 2000.

[9] Goto I., N. Kato, N. Uratani and T. Ehara, "Transliteration Considering Context Information Based on the Maximum Entropy Method," In Proceedings of MT-Summit IX, 2003

[10] Qu Yan, Gregory Grefenstette, David A. Evans, "Automatic transliteration for Japanese-to-English text retrieval," In Proceedings of ACM SIGIR'2003, pp. 353-360, 2003

[11] Dorre J., Gerstl, P., and Seiffert, R., "Text Mining:Finding Nuggets in Mountains of Textual Data," in Proceedings of the fifth ACM SIGKDD International

Conference on Knowledge Discovery and Data Mining, 1999.

〔12〕 Lee Hing Yan, "Text Mining-Knowledge Discovery from Text," Trend in Knowledge Discovery from Databases, 29th June 1999.

〔13〕 Jong-Hoon Oh, Sun-Mee Bae, Key-Sun Choi, "An Algorithm for extracting English-Korean Transliterationpairs using Automatic E-K Transliteration," In Proceddings of Korean Information Science Society, 2004.

〔14〕 C.J. Lee, J.S. Chang, J.S. Jang, "Extraction of transliteration pairs form parallel corpora using a statistical transliteration model," Information Science 176, 67-90, 2006

〔15〕 Chun-Jen Lee, Jason S. Chang, Jyh-Shing Roger Jang: "Alignment of bilingual named entities in parallel corpora using statistical models and multiple knowledge sources," ACM Trans. Asian Lang. Inf.Process. 5(2): 121-145, 2006

〔16〕 Satish L, Gururaj BI, "Use of hidden Markov models for partial discharge pattern classification," IEEE Transactions on Dielectrics and Electrical Insulation, April 2003.

저 자 소 개

주 해 종

명지대학교전자계산학과 공학사
명지대학교 전자계산학과 공학석사
(美)Cumberland University 교육학(컴퓨터)박사
(현)동국대학교 산학협력중심대학 육성사업단 교수

홍 봉 화

경희대학교 전자공학과 공학사
경희대학교 전자공학과 공학석사
경희대학교 전자공학과 공학박사
(현)경희사이버대학교 정보통신학과 부교수

정 복 철

경희대학교 법학 학사
경희대학교 도시·지역정책학 석사
경희대학교 정치외교학과 박사
(현)경희대학교 교양학부 조교수