한국지능시스템학회 논문지 2011, Vol. 21, No. 1, pp. 68-79
DOI : 10.5391/JKIIS.2011.21.1.68

웹2.0에서 의견정보의 실시간 모니터링을 위한 웹 콘텐츠 마이닝 시스템*

Web Contents Mining System for Real-Time Monitoring of Opinion Information based on Web 2.0

김영춘* · 주해종** · 최혜길*** · 조문택**** · 김영백***** · 이상용******+

Young-Choon Kim, Hae-Jong Joo, Hae-Gill Choi, Moon-Taek Cho, Young-Baek Kim and Sang-Yong Rhee

* 공주대학교 기계자동차공학부
** 동국대학교 산학협력중심대학
*** 경희사이버대학교 정보통신학과
**** 대원대학 전기전자계열
***** 경남대학교 컴퓨터공학부

요 약

본 연구에서 제안하는 시스템은 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 콘텐츠에서 사용자 의견 정보들을 자동 추출 및 분석함으로써, 긍정/부정 의견별로 검색 및 통계를 확인할 수 있는 의견 검색 서비스를 제공한다. 그 결과 의견 검색 사용자들은 특정 키워드에 대하여 다른 사용자들의 의견을 손쉽게 한눈에 검색 및 모니터링하는 시스템을 용이하게 사용할 수 있으며, 웹 콘텐츠에서의 의견 추출 및 분석하는 기능을 제공받는다.
제안한 기법들은 다른 기법들과의 비교 실험을 수행하여 실제 성능이 우수함을 증명하였다. 성능 평가는 긍정/부정 의견 정보를 추출하는 기능의 성능 평가, 다국어 정보 검색을 위한 동적 윈도우 기법과 토크나이저 기법을 적용한 성능 평가, 그리고 정확한 다국어 음차표기를 추출 기법에 성능 평가를 실시하였다. 그 적용 사례로 대표적인 영화 리뷰 문장과 위키디피아 실험 데이터를 대상으로 실험하고 그 결과를 분석하였다.

키워드 : 모니터링 검색 시스템, 콘텐츠 의견 정보 자동 추출, 웹 마이닝, 콘텐츠 의견 정보 모니터링

Abstract

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.
Proposing technique proved that the actual performance is excellent by comparison experiment with other techniques. Performance evaluation of function extracting positive/negative opinion information, the performance evaluation applying dynamic window technique and tokenizer technique for multilingual information retrieval, and the performance evaluation of technique extracting exact multilingual phonetic translation are carried out. The experiment with typical movie review sentence and Wikipedia experiment data as object as that applying example is carried out and the result is analyzed.

Key Words : Motoring Search System, Opinion Information Automatic Extraction, Web Contents Mining, Opinion Information Monitoring

1. 서 론

최근 많은 사람들이 인터넷의 블로그(Blog), 위키(Wiki)와 같은 매체를 통해서 자신의 의견을 표현하고 있으며[1, 특정한 정보의 가치를 평가할 때, 이러한 다른 사람들이 인터넷 상에 올려놓은 의견 정보를 참조하고자 하는 수요도 높아지고 있다.

예를 들면, 인터넷 상에는 상품 리뷰(Review)에서 영화 리뷰까지 다양한 사용자들의 의견이 존재한다. 이러한 각 사용자들의 의견들은 일반 사용자들이 물품을 구매하거나, 영화를 보기 전에 다른 사용자들의 의견을 보고자 하는 경우에도 이용될 수 있으며, 마케팅 담당자나 주식 매매자 등

접수일자 : 2011년 1월 27일
완료일자 : 2011년 2월 10일
본 연구는 2010년도 경남대학교 학술연구장려금 지원으로 이루어졌음
+ 교신저자

이 각 물품이나 회사에 대한 일반 사용자들의 다양한 의견을 알고자 하는 경우에도 사용될 수 있다. 특히, 일반 사용자들은 특정 물품을 구매하기 전에 다른 사용자들의 평가를 참조하는 경향이 크다. 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. 이러한 모든 웹사이트들을 사용자들이 모두 찾아보기 어려우며 일반 검색으로 다른 사용자들의 의견을 찾고자 하는 경우에는 의견이 있는 웹 문서, 긍정적인 의견이 있는 웹 문서, 부정적인 의견이 있는 웹 문서 등이 혼재하여 효과적으로 다른 사용자들의 의견을 찾아보기 어려운 문제점이 있다[1,9].

이러한 문제점을 해결하기 위하여 국・내외 학계를 중심으로 사용자 의견 추출 기술이 활발하게 연구되고 있으며, 정보 검색 분야에서도 2000년도 초반부터 크게 발전하여 다양한 기술이 연구되고 있다[1,3,5]. 그러나 기존의 정보 검색 기술은 단순히 키워드가 존재하는 정보를 기반으로 하는 검색만 제공해주고 있을 뿐이고, 각 키워드가 등장하는 문서나 문장에서 긍정적/부정적으로 평가된 내용을 기반으로 한 좀더 고차원적인 검색까지 제공해주고 있지 못하고 있다. 최근에 사용자 의견 추출 기술을 정보 검색에 적용하려는 시도가 진행되고 있으나 아직도 단순히 긍정, 부정 문서를 나누는 수준에만 머무르고 있는 실정이다.

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

이 논문의 구성은 다음과 같다. 2장에서는 본 논문의 이론적 고찰을 위해 기존 웹 마이닝 기법, 의견 추출 기법, 그리고 다국어 언어학 사전의 이론적 배경을 살펴보고 문제점을 살펴본다. 3장에서는 인터넷 상에서 의견 정보를 수집 및 분석하여 모니터링할 수 있는 웹 콘텐츠 마이닝 시스템의 설계와 구현 방법을 제안한다. 4장에서는 본 제안 시스템의 성능분석을 위하여 의견 정보추출 및 분석 기능의 성능평가, 동적 윈도우와 토크나이저 기법의 성능평가, 그리고 다국어 음차표기 추출 기법의 성능평가를 제시한다. 마지막으로 5장에서는 결론을 맺는다.

2. 관련 연구

2.1 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 기계학습 방법이 연구 되었다[7,8].

규칙 기반 방법에 기반하여 구 단위로 의견을 추출하는 연구로는 Nasukawa와 Zhongchao Fei의 연구가 있다[9,10]. 규칙기반에서는 각각의 단어를 그 단어의 품사 정보와 극성 정보를 합쳐서 태그를 부여한다[11,12].

기계학습 방법은 문장 내에서 긍정 또는 부정 표현 부분에 태깅된 코퍼스를 이용하여 기계학습을 수행한다. 의견 태깅 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 사용하였다.

규칙 기반 방법은 일치 하는 패턴에 대해서는 높은 정확도를 보이지만, 이미 구축한 패턴이 약간 변형된 형태로 나온 경우에 취약점을 보여서 재현율이 크게 떨어지는 단점과, 주변 문맥 정보를 학습에 반영하지 못하는 한계를 지니고 있다.

기계 학습 기반의 단어/구 단위의 의견 분류는 주변 문맥 정보와 여러 자질을 학습에 반영하여 학습 코퍼스 표현의 다양한 변형에 대해서도 의견 분류가 가능하고, 이에 따라서 높은 재현율을 지니는 장점을 가지고 있다. 문제점은 학습을 위해서는 문장 내 의견 표현 부분이 수동으로 태깅된 학습 코퍼스가 필요한데, 현재 가용한 학습 코퍼스는 영어권에 소량의 MPQA(Multi-Perspective Question Answering) 데이터만 존재할 뿐이어서, 실제로 기계 학습 기반의 방법을 적용하기 위해서는 다른 도메인과 다른 언어권에 대하여 의견 표현이 태깅된 학습 코퍼스가 필요하다. 하지만 이 학습 코퍼스를 수동으로 구축하는 일은 규칙 기반에서 의견 패턴을 구축하는 일과 마찬가지로 많은 인력과 시간이 필요한 작업이다.

2.2 음차표기 모델

기계번역과 교차언어 정보검색 등과 같은 자연언어 응용에서 사용되는 번역지식을 자동으로 구축하는 연구에서, 음차표기된 단어들은 새로운 개념을 나타내는 신조어가 많기 때문에 사전에 등재되어 있지 않은 경우가 많다. 따라서 효과적인 번역지식 구축을 위해서는 이러한 음차표기 번역지식을 자동으로 획득하는 것은 매우 중요하다. 주어진 영어 단어에 대한 음차표기 대역어를 획득하는 연구로는 자동 음차표기와 음차표기 대역쌍 추출 등이 있다. 자동 음차표기는 주어진 영어 단어를 비영어권의 언어의 단어로 음차 표기하는 기법이다[7].

음차표기 대역쌍 추출은 이중언어 문서 (bilingual corpora)에서 영어와 영어에 대응되는 음차표기된 단어를 자동으로 추출하는 기법이다[13]. 이들 연구는 번역사전의 적용범위를 높이기 위하여 단어의 번역지식을 이중언어 문서로부터 자동으로 추출하는 연구로서, 번역지식은 음차표기 대역쌍으로 한정하여 수행하였다.

중국어 음차표기 대역쌍 추출에서는 일반적으로 중국어 한자의 로마표기법, 즉 병음을 사용하여 영어와 비교한다[15]. 통계기반 음차 표기 모델에서는 E는 영어, C는 중국어, TU(Transliteration Unit)는 음차표기 단위로 가정한다. 그러면 조건확률P(C|E)는 P(중국어|영어)로 치환되어 P(C|E)확률을 구하는 문제로 전환할 수 있다. 또한 영어는 유니그램(Unigram), 바이그램(Bigram), 트라이그램(Trigram)을, 중국어는 병음의 첫 음절, 마지막 음절 혹은 병음 전체를 TU로 사용하고 있다.

본 논문에서도 발음사전 없이 EM(Expectation Maximization) 알고리즘[13]을 적용하여 파라미터를 자동으로 추정하는 방법을 사용하였으며, 음차표기 모델에 매치타입 정보를 추가하였다. P(C|E) 식에 매치타입(M) 정보를 추가하

면 식 1과 같다.

$$P(C|E) \fallingdotseq \max P(C|M,E)P(M|E) \fallingdotseq \max P(C|M,E)P(M) \quad (1)$$

3. 모니터링 검색엔진을 위한 웹 콘텐츠 마이닝 시스템

모니터링 검색엔진을 위한 웹 콘텐츠 마이닝 시스템은 웹 콘텐츠에서 의견 정보를 자동추출 및 분석하기 위한 시스템으로, 그 플랫폼은 그림 1과 같다. 제안 시스템은 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 문서에서 사용자 의견 정보들을 자동 추출 및 분석함으로써, 긍정/부정 의견별로 검색 및 통계를 확인할 수 있는 의견 정보 검색 서비스를 제공하는 시스템을 제안한다. 제안한 시스템은 의견 정보 검색 사용자들에게 특정 키워드에 대하여 다른 사용자들의 의견 정보를 손쉽게 한눈에 검색 및 모니터링할 수 있는 기능을 제공하기 위한 구조이다.

제안한 의견정보 모니터링 시스템은, 사용자가 모니터링 키워드를 등록하면 한국어, 일본어, 중국어, 영어로 등록 키워드를 변환하여, 인터넷의 의견정보를 자동 모니터링 수집한다. 수집된 모니터링 정보는 다시 어떤 정보의 종류 군을 자동 분류하고, 분류된 정보에서 긍정의견과 부정의견을 자동 추출한다.

즉, 이러한 의견정보 모니터링 시스템을 사용함으로써 사용자는 온라인 의견 정보를 실시간 모니터링 하여, 추출된 긍정정보와 부정정보를 쉽게 파악한 후, 각 구분 정보에 따라 효율적인 실시간 대처를 할 수 있게 된다. 그에 따라, 비용절감, 시간단축, 기회정보에 따른 기회창출 효과, 유해정보 조기대응에 대한 경제적 피해 손실을 줄일 수 있는 "미래시그널 예측 툴"이라고 정의할 수 있다.

제안된 시스템은 크게 데이터 수집 처리, 의견/비의견 자동 구축, 의견 정보 자원, 인덱싱 처리, 의견 인덱싱 정보 자원, 의견표현 기계 학습, 다국어 언어학 사전 자동 등록, 다국어 의견 정보 자원, 의견 검색 처리 및 사용자 단말 등을 포함하여 이루어진다.

제안 시스템의 목적을 달성하기 위하여 인터넷 상에 존재하는 웹 문서 데이터를 수집하여 문장 단위로 분리하고, 분리된 각 문장에 대해 언어처리를 수행하여 언어적인 자질들을 추출한다. 또한 추출된 각 문장의 언어적인 자질들을 이용하여 의견/비의견 문장을 구분하여 구분된 의견 문장의 언어적인 자질들에 대해 긍정/부정 의견표현을 구분함과 동시에 의견 문장의 언어적인 자질별로 해다 웹 문서의 의견정보들이 저장되도록 인덱싱 처리를 수행한다. 그리고 다국어 검색 지원을 위해 다국어 언어학 사전을 자동 등록하게 되면 의견 정보 자동 추출 웹 마이닝 시스템을 위한 플랫폼이 준비되게 된다.

3.1 데이터 수집 처리

데이터 수집 처리는 인터넷 상에 존재하는 다양한 웹 콘텐츠들을 수집하는 기능을 수행한다. 즉, 데이터 수집 처리는 인터넷 상에 존재하는 각 웹사이트들의 HTML 정보를 실시간으로 다운로드 받게 된다. 또한, 데이터 수집처리는 상기와 같이 다운로드받은 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.

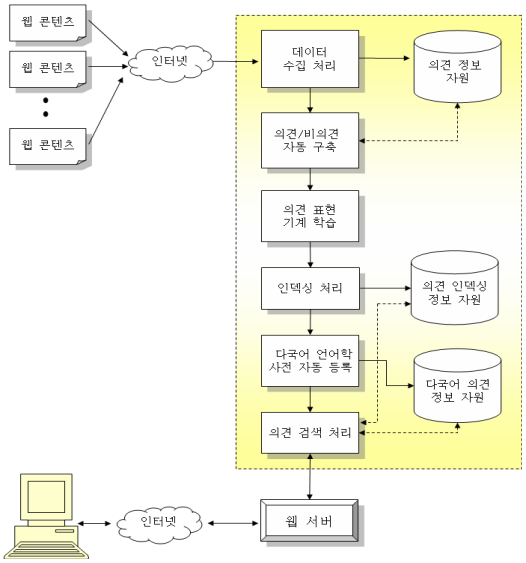


그림 1. 의견 정보 자동추출 웹 마이닝 시스템 플랫폼
Fig. 1. Opinion information automatic extraction web mining system platform

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. 이때, 상기 의견정보 데이터를 포함하는 웹 콘텐츠들만을 선별적으로 수집하는 방법으로는, 의견정보 데이터를 포함하는 특정의 웹 콘텐츠를 선별하고, 후술하는 기계학습 알고리즘(예컨대, SVM, K-NN, Bayseian 등)을 사용하여 웹 콘텐츠 선별 모델을 생성한 후, 상기 생성된 웹 콘텐츠 선별 모델을 사용하여 전체 인터넷 웹 페이지에서 의견정보 데이터가 포함된 웹 콘텐츠들만을 선별적으로 수집할 수 있게 된다.

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.

이를 구체적으로 설명하면, 본 논문의 실시 예에서 사용하는 점수 범위가 a~b 라고 하였을 때에 수집한 데이터의 점수범위가 c~d 라고 한다면, 해당 수집 점수 x는 식 2와 같이 변화한다.

$$PolarityScore(x) = (a-1) + \frac{x-c+1}{d-c+1} \times (b-a+1) \quad (2)$$

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 3과 같이 계산한다.

$$PolarityScore(2) = (1-1) + \frac{2-1+1}{5-1+1} \times (10-1+1) = 4 \quad (3)$$

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 사랑이야기 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

데이터 수집 처리에 의해 수집된 데이터는 표 1과 같이 본 연구에서 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 설명, 그리고 샘플 데이터의 표현 형식을 나타내고 있다.

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

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	…
3040343	hjjoo007	2009-08-03	영화	이 영화는 최고의 쓰레기 영화이다.	7	…

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 2와 표 2의 의견 정보 데이터 구조를 이용하여 인터넷에서 쉽게 구할 수 있는 문장 단위 긍정/부정 의견 정보가 표현된 데이터를 수집한다.

3.2 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 보간법(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.

3.2.1 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-Gram에 대한 점수를 구하게 된다.

$Freq(W_j, S_i)$는 단어 W_j 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 4와 같다.

$$Score(\text{영화}) = \log\left[\frac{Freq(\text{영화}, s_{10})+1}{Freq(\text{영화}, s_1)+1}\right] = \log\left[\frac{9+1}{1+1}\right] = 1.60 \quad (4)$$

식 4는 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 같이 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 5는 절대적인 빈도수를 상대적인 확률로 변환시키기 위한 수식이다.

$$Freq(w_j, s_i) \rightarrow \frac{Freq(w_j, s_i)}{\sum_{w_k \in W} Freq(w_k, s_i)} = P(w_j | s_i) \quad (5)$$

식 5에서 $Freq(W_j, S_i)$는 단어 W_j 가 점수 집합 Si에서 나타나는 횟수를 나타내며, $\sum_{w_k \in W} Freq(W_k, S_i)$는 모든 점수 집합에서 단어 W_k가 나타나는 횟수를 더한 값으로서 결국 전체 데이터에서 W_k가 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 6과

같다.

$$PolarityScore(w_j) = \frac{\sum_{s_i \subset S} [i \times P(w_j|s_i)]}{\sum_{s_i \subset S} P(w_j|s_i)} \quad (6)$$

식 6에 따라서 이전에 절대 값의 평균을 취했던 부분을 정규화된 값들의 평균으로 바꿈으로 특정 점수 집합에 치우치지 않은 정규화된 의견 점수를 얻을 수 있게 된다.

3.2.2 주관적 점수를 이용한 산출 방법

앞에서 제안한 단어의 극성 점수 산출 방법만으로 의견 단어 자원을 구축하게 되는 경우에 의견을 나타내지 않는 특정 배우의 이름이 한 점수 집합에서 많이 나왔다는 이유로 긍정/부정 단어 자원으로 구축될 위험이 존재한다. 즉 단어 자원을 구축할 때에는 극성 점수뿐만 아니라 단어의 주관성도 계산을 해야 한다. 이때 주관성에 영향을 미치는 요소를 극성 점수를 계산하는 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 7은 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 계산한다.

$$\text{주관적점수}(pos_j) = \frac{\sum_{s_i \subset S} [i \times P(pos_j|s_i)]}{\sum_{s_i \subset S} P(pos_j|s_i)} \quad (7)$$

식 8은 본 논문의 의견 정보 단어를 자동 구축하는 제안 방법으로, 극성 점수와 주관적 점수 두 가지 점수를 이용하여 보간법을 사용하여 단어의 의견점수(OpinionScore)를 구하게 된다. 이 의견점수는 그 단어가 얼마나 의견 단어인지를 나타내는 점수로서, 특정 점수 이하의 단어는 의견 단어로 사용하지 않게 된다.

$$OpinionScore(w_j) = \left| PoarityScore(w_j) - \frac{1}{2} \times \max(S) \right| \times \alpha + \left[SujectiveScore(pos_j) - \frac{1}{2} \times \max(S) \right] \times (1-\alpha) \quad (8)$$

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

3.2.3 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 뿐 만 아니라, 명사나 특정 대상도 비유적인 표현으로 의견을 나타내기 위해서 쓰임을 볼 수 있다.

특히 부정 의견쪽 단어들은 대부분 영화 도메인에 의존적인 표현임을 볼 수 있다. “푹/MAG 자/VV” 같은 경우 다른 도메인에서는 의견을 나타내지 않는 단어이지만, 영화 도메인에서는 “영화 보는 내내 푹 자다가 왔다” 와 같이 부정적인 표현으로 사용될 수 있다.

이처럼 의견 표현은 다양한 비유, 반어, 비교 표현이 존재하기 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. Opinion Word Automatic Construction Result

	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

의견 단어 자원이 구축된 후에 이 단어 자원의 극성으로 바로 의견 문장에 긍정/부정 표현을 태깅할 수도 있다. 하지만 “이 영화는 최고의 쓰레기 영화이다” 문장은 부정 문장 집합에 속하는 한 문장인데, “최고의” 라는 단어가 긍정 점수 수치를 가지고 있다고 해서 해당 문장에서 “최고의”를 긍정으로 태깅하면 잘못된 태깅이 된다. 그러한 문장에서는 부정적인 의미로 사용되었기 때문이다. 이와 같은 문제점 때문에 본 논문에서는 그 단어의 극성을 바로 사용하지 않고, 일정한 기준 점수(7점) 이상의 긍정적인 단어와, 일정한 기준 점수(4점) 이하의 부정적인 단어를 주관적인 단어로 선별하였고, 이 주관적인 단어를 그 단어의 극성이 아니라 그 문장의 극성에 따라서 의견 표현 태깅을 하였다. 실제 실험 결과에서도 단어의 극성으로 태깅한 것 보다 그 문장의 극성에 따라서 태깅한 것이 더 좋은 성능을 보였다.

3.3 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].

□평가 문제(Evaluation Problem)

관찰된 심볼의 시퀀스 $O = O_1 O_2 \cdots O_T$와 모델

$\lambda=(A, B, \pi)$가 주어졌을 때, 그 모델에서 관찰된 데이터 O의 확률 $P(O\lambda)$를 어떻게 구할 것인가 하는 문제이다.

□디코딩 문제(Decoding Problem)

관찰된 심볼의 시퀀스 $O=O_1O_2\cdots O_T$와 모델 $\lambda=(A, B, \pi)$가 주어졌을 때 최적의 상태 전이 시퀀스 $Q=q_1q_2...q_t$는 무엇인가 하는 문제이다.

□ 학습 문제(Estimation Problem)

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 연구에서는 모델 파라미터인 상태 전이 확률, 관찰 확률, 초기 상태 확률은 의견/비의견 자동 구축 모듈에서 수집한 태깅 코퍼스로부터 얻는다.

3.4 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 데이터베이스(DB)화하여 저장되는 기능을 수행한다.

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

id	comm entct	date	snippet	data	pola rity	topic	url	…
big seri-al (PK)	char varying (50)	bi-gint	char varying (200)	char vary-ing (2000)	char vary-ing (10)	char varyi ng (10)	char varyi ng (200)	…
문장 식별 자	콘텐츠 설명	문장 날짜	의견정 보	문장	산출 점수	분야	url 정보	…
3040 343	부정적 내용	2009-08-03	이/MM 영화/ NNG 는 /JX 최고 ...	이 영화는 최고의 쓰레기 영화.	4	영화	http: //mo vie.n aver. com	…

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 설명, 그리고 샘플 데이터의 표현 형식을 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 2는 대 분류 인덱스를 구성하는 그림이다.

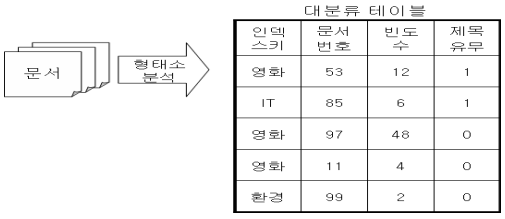


그림 2. 대분류 인덱싱
Fig. 2. Large Classification Indexing

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 3는 상세 분류 인덱싱을 구성하는 그림이다.

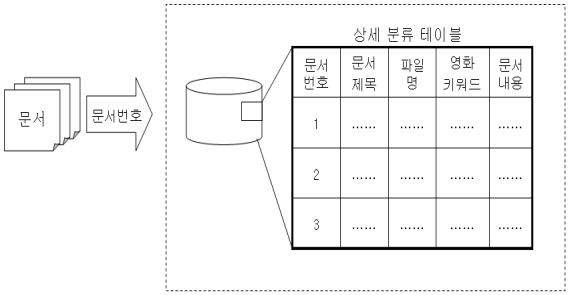


그림 3. 상세 분류 인덱싱
Fig. 3. Detailed Classification Indexing

3.5 다국어 언어학 사전 자동 등록

본 절에서 제안하는 것은 통계기반 음차표기 모델을 이용하여 웹 상의 이중언어 문서에서 다국어 언어학 사전을 자동으로 구축하는 시스템으로써, 기존의 수동적인 언어학 사전 구축방식에 비하여 인력을 대폭 절감할 수 있는 효과를 가진다. 즉, 병렬코퍼스에서의 음차표기 대역쌍 추출을 확장하여 대량의 비교 가능한 코퍼스에서의 음차표기 대역쌍 추출을 진행하였다. 비교 가능한 코퍼스에서는 추출된 영어단어가 음차표기 안 될 수도 있다. 또한, 음차표기에 대응되는 중국어 문서에 출현하지 않을 수도 있고, 한 문장에서 음차표기를 추출하는 것이 아니라 여러 개 문서에서 추출하기 때문에 노이즈도 병렬코퍼스에 비하여 훨씬 커지게 된다. 기존에는 비교 가능한 코퍼스에서의 음차표기 대역쌍 추출은 대부분 두 언어에서 각각 음차표기 후보를 추출하고 시간축에 따른 그것들의 빈도수 유사도를 계산하여 추출하였다.

이런 방법의 단점은 빈도수가 높은 단어에만 적용 가능하여 중국어와 같이 고유명사 인식이 잘 안되는 언어에서는 성능이 크게 떨어지는 것으로, 본 절에서는 병렬 코퍼스에 적용했던 동적 윈도우와 토크나이저 기법을 기반으로 음차

표기 빈도수와 음성적 유사도를 이용한 대량의 비교 가능한 코퍼스에서의 음차표기 대역쌍 추출 방법을 제안한다.

3.5.1 음차 표기 자동 추출 모델

본 절에서 제안한 영-중 음차표기 자동 추출 모델은 먼저 영-중 병렬 코퍼스의 영어 문장에 고유명사 인식 모듈을 적용하여 고유명사를 추출한 후, 그 중에서 음차표기 될 영어 단어만 선택하여 대응되는 중국어 문장에서 음차표기 단어를 추출하였다. 그림 4은 영-중 병렬 코퍼스에서 음차표기 대역쌍을 추출하는 과정을 보여주고 있으며, 병렬 코퍼스 데이터 구조는 표 5와 같다.

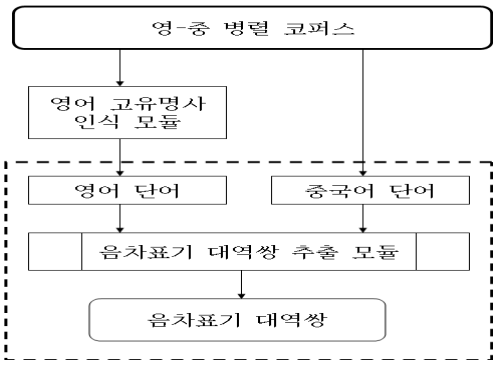


그림 4. 영-중 코퍼스에서 음차표기 대역쌍 추출 과정
Fig. 4. Phonetic Translation Interlinear Pair Extraction Process in English-Chinese Corpus

표 5. 다국어 의견정보 자원 데이터 구조
Table 5. Multilingual Opinion Information Resource Data Structure

id	from language	to language	from_text	to_text	…
bigserial (PK)	char vary-ing(50)	char vary-ing(55)	char vary-ing(2000)	char vary-ing(2000)	…
문장 식별자	원시 언어	목표 언어	원시 문장	번역 문장	…
3043201	China	English	杰西(JieXi)	Jacey	…

2장의 관련 연구에서 살펴본바와 같이, 통계기반 음차표기 모델을 적용하여 음차표기 대역쌍을 추출할 때, 만약 한 문장에 주어진 영어 단어와 발음상 비슷한 중국어 문자열이 여러 개 존재할 경우 오류가 자주 발생한다. 본 논문에서는 이런 오류를 해결하고자 동적 윈도우 기법과 토크나이저 기법을 제안한다.

3.5.2 동적 윈도우 기법

동적 윈도우 기법은 중국어 문장에 대하여 한번에 최적화된 경로를 찾는 것이 아니라 주어진 영어 단어에 근거하여 가능한 중국어 음차표기 단어크기의 범위를 설정하고, 그 범위 내의 윈도우를 각각 앞으로 이동하면서 음차표기를 찾는 기법이다. 만약 중국어 음차표기 단어의 실제 길이를 알 수 있고, 그것을 윈도우 크기로 설정하여 음차표기를 찾으면 아주 높은 성능을 낼 수 있다.

동적 윈도우 기법을 적용하면, 통계기반 음차표기 모델을 적용했을 때 생기는 대부분 오류들을 해결할 수 있다. 그러나 정확한 윈도우를 적용했을 때에도 여전히 오류가 생기는 일부 문장에 대해서는 해결할 수 없다. 또한 동적 윈도우만 적용할 경우 전체 문장에 대해 다양한 크기의 윈도우로 여러 번 정렬 과정을 거치야 하므로 시간 복잡도가 너무 커지게 되는 단점이 있다. 아래에 시간 복잡도를 크게 줄여주고 성능도 조금 더 향상 시켜주는 토크나이저 기법에 대하여 기술한다.

3.5.3 토크나이저(Tokenizer) 기법

토크나이저 기법은 중국어 음차표기에 전혀 사용되지 않는 문자를 기준으로 중국어 문장을 먼저 여러 부분으로 나누고, 각 부분에 대하여 통계기반 음차표기 모델을 적용하여 음차표기를 추출하는 기법이다. 문자열 토크나이저 클래스를 사용하면 응용프로그램이 문자열을 토큰으로 분리할 수 있다.

3.5.4 엔트로피를 이용한 비음차표기 인식 기법

본 논문의 제안에서는 먼저 인터넷을 이용하여 대용량의 영-중 비교 가능한 코퍼스를 구축하고, 이런 코퍼스에서 위에서 제안한 동적윈도우와 토크나이저 기법을 사용하여 음차표기쌍을 추출한다. 추출된 음차표기쌍은 기본적으로 음성적 유사도 값을 갖고 있고, 한 영어단어가 기본적으로 여러 문서에 나타날 수 있으므로 일정한 빈도수 값도 갖게 된다. 표 6은 비교 가능한 코퍼스에서 추출한 음차표기 쌍을 표시한다.

표 6. 비교가능한 코퍼스에서 추출한 영-중 음차표기 쌍
Table 6. English-Chinese Phonetic Translation Pair extracted from comparable Corpus

영어 단어	중국어 단어	음성적 유사도	빈도수
Clinton	克林頓	-4.29	37
Clinton	柯林頓	-4.29	6
Clinton	克林	-11.33	5
Clinton	阿灵頓	-7.44	3
Clinton	利福尼	-11.34	3
Clinton	林斯頓	-8.93	2
Clinton	里森	-10.33	2
Clinton	立運動	-10.49	2
Clinton	理論	-10.83	2
Clinton	倫敦	-10.98	2
Clinton	爾頓	-11.04	2
Clinton	安德魯	-11.27	2

기본적으로 대응되는 중국어 음차표기가 있는 영어 단어는 음성적 유사도와 빈도수 측면에서 크게 아래와 같은 두 가지 특징이 있다. 첫째, 영어단어와 정확한 중국어 음차표기 사이의 음성적 유사도는 영어 단어와 다른 중국어 문자열과의 음성적 유사도 보다 높다. 둘째, 음차표기가 있는 영어단어는 앞에서 제안한 동적윈도우 기법과 토크나이저 기법으로 추출한 것이기 때문에 대응되는 중국어 문서에 정확한 음차표기가 있을 경우 거의 99%정도 성능으로 그 음차표기를 찾아 준다. 따라서 영어 단어와 정확한 음차표기쌍

의 빈도수는 다른 단어쌍의 빈도수에 비해서 높게 된다. 반대로 중국어 음차표기가 없는 영어단어는 추출된 음차표기 쌍들이 음성적 유사도가 낮고 빈도수도 낮게 된다. 이런 특성을 이용하기 위하여 식 9와 같이 한 영어 단어 대하여 엔트로피를 구할 수 있다.

$$H = -\sum_{N} P_i \times \log(P_i) = -\sum_{N} \left(\frac{f_i \times s_i}{\sum_{N} f_i \times s_i}\right) \times \log\left(\frac{f_i \times s_i}{\sum_{N} f_i \times s_i}\right) \quad (9)$$

여기서 f_i와 s_i는 각각 빈도수와 음성적 유사도 값을 의미하고 N은 추출된 서로 다른 중국어 문자열의 개수를 의미한다. 그러나 엔트로피 값은 N이 커짐에 따라 커지는 경향이 있으므로, 본 논문에서는 특정 N에 대하여 엔트로피 최대값인 $\log(N)$으로 정규화 하여 엔트로피 값을 모두 0과 1사이로 조절하였다. 비음차표기는 음성적 유사도도 낮고, 음차표기가 불규칙적으로 뽑히므로 엔트로피 값이 커지게 된다. 그러므로 본 논문에서는 특정 임계값을 조절하여 그 임계값을 넘는 영어단어는 음차표기가 안되거나 음차표기 추출이 안됐다고 판단하여 걸러낸다.

3.5.5 음성적 유사도와 빈도수를 이용한 음차표기 추출 기법

3.5.4의 방법으로 비음차표기를 걸러낸 후 음차표기를 포함하고 있는 결과 중에서 정확한 음차표기만 추출하여야 한다. 정확한 음차표기는 특성상 음성적 유사도 값이 크고, 빈도수도 다른 중국어 문자열에 비하여 높기 때문에 음성적 유사도와 빈도수를 동시에 적용하여 특정 임계값을 넘는 음차표기만 추출한다. 즉, 먼저 (식 9)에서 $P_i = \frac{f_i \times s_i}{\sum_{N} f_i \times s_i}$를 이용하여 p_i를 내림차순으로 정렬한다. 여기서 음차표기가 아닌 중국어 문자열의 p_i는 음차표기에 비하여 음성적 유사도도 낮고, 빈도수도 작기 때문에 음차표기의 p_i보다 많이 낮게 된다. 이런 특성을 이용하여 최고 p_i인 p_0와 p_i의 나눈 값이 특정 임계값보다 높은 것만 추출한다.

3.6 의견 검색 처리

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

즉, 웹 서버에 전달되는 내용은 "키워드(Keyword) : 영화 제목, 타입(Type) : 긍정/부정/의견"이 될 수 있다. 여기서, 상기 타입 정보 중에서 "의견"이라 함은 긍정 및 부정 의견이 모두 함께 나타나는 검색 결과이며, "긍정"이라 함은 긍정 의견만 나오는 타입이다. "부정"이라 함은 부정 의견만 나오는 타입이다. 이와 같이 특정 의견검색 키워드와 타입을 의견 검색 모듈에 전달하게 되면, 인덱싱 서버 또는 인덱싱 정보 저장모듈에서 해당 특정 의견검색 키워드와 해당 타입에 해당되는 데이터를 읽어 와서 의견의 양이나 날짜 순서 등의 랭킹(Ranking)으로 검색된 결과를 다시 웹 서버에 전송해준다. 이때, 검색된 결과 정보는 예컨대, 제목, 링크(Link), 해당 사이트 제목, 긍정 개수, 부정 개수, 긍정 개수, 본문 내용, 본문 요약 내용, 긍정 표현 위치, 부정 표현 위치 등으로 이루어질 수 있다.

그림 5은 의견 정보를 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.

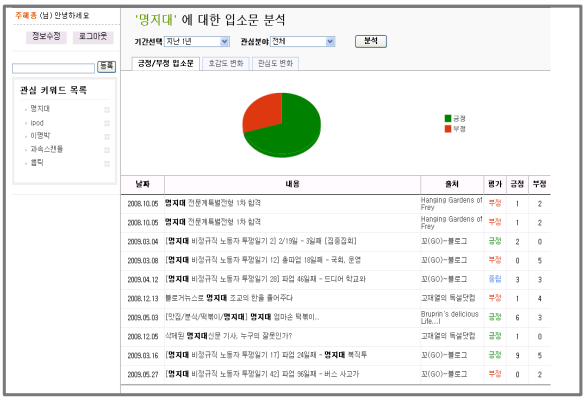


그림 5. 의견 정보 모니터링 메인 화면
Fig. 5. Opinion Information Monitoring Main Screen

4. 실험 및 성능 평가

본 장에서는 모니터링 검색엔진을 위한 웹 콘텐츠 마이닝 시스템의 성능을 평가하기 위하여 긍정/부정 의견 정보를 추출하는 기능과 다국어 정보 검색을 위한 동적 윈도우즈 기법과 토크나이저 기법을 적용하고, 정확한 다국어 음차표기를 추출 기법에 대한 실험과 성능 평가한다. 그 적용 사례로 대표적인 영화 리뷰 문장과 위키디피아 실험 데이터를 대상으로 실험하고 그 결과를 분석한다.

4.1 의견 정보 추출 및 분석 기능의 성능 평가

본 실험에서는 네이버 영화(http://movie.naver.com)의 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.

본 논문에서 성능 평가할 기준은 지도학습 방법의 HMM, CRF이다. HMM은 3.4절에 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 7. 자동 구축한 의견 구가 태깅된 코퍼스 성능 평가 결과
Table 7. Result of Performance Evaluation of Corpus which Automatically Constructed Opinion Phrase

	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 7에서 자동 구축한 의견 구가 태깅된 데이터의 성능을 살펴보면, 단어가 가지는 극성을 바로 적용한 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 결정하는데 더 큰 역할을 함을 알 수 있다.

4.2 동적 윈도우 기법과 토크나이저 기법의 성능 평가

본 절의 실험을 위하여 영-중 병렬 코퍼스에서 지명, 인명, 제품명을 등 각종 음차표기 대역쌍을 포함한 300개 문장을 선택하였다. 학습 데이터는 860개 영-중 음차표기 단어쌍을 사용하였다. 성능 평가를 위하여 본 논문에서 제안하는 기법과 기존연구와의 비교 실험을 수행하였다. 성능은 단어 정확률, 문자 정확률, 문자 재현율로 평가한다. 본 논문의 알고리즘은 한 번에 하나의 영어 단어에 대해서만 고려하기 때문에 단어와 문자의 정확률과 재현율은 그림 6과 같게 된다.

동적 윈도우 기법과 토크나이저 기법의 타당성을 증명하기 위하여 아래와 같은 실험을 수행하였다. 첫 번째는 기존 연구인 통계기반 음차표기 모델(STM)만 적용한 실험이고, 두 번째는 통계기반 음차표기 모델(STM)에 동적 윈도우(DW)와 토크나이저(TOK)를 각각 적용한 실험이고, 세 번째는 동적 윈도우와 토크나이저를 동시에 적용한 실험이다.

○ 단어 정확률$(wp) = \frac{\text{정확히 찾은 단어수}}{\text{정확한 단어수}}$

○ 문자 정확률$(cp) = \frac{\text{정확히 찾은 문자수}}{\text{찾은 문자수}}$

○ 문자 재현율$(cr) = \frac{\text{정확히 찾은 문자수}}{\text{정확한 문자수}}$

그림 6. 음차표기 성능평가를 위한 기준
Figure 6. Standard for Performance Evaluation of Phonetic Translation

신뢰성 평가를 실시한 표 8에서 보여주듯이 통계기반 음차표기 모델(STM)만 적용하여 대역쌍을 추출하면 약 75% 정도의 성능을 낼 수 있다. 실제로 통계기반 음차표기 모델은 짧은 문장에서는 비교적 좋은 성능을 낼 수 있으나, 문장 길이가 길어짐에 따라 노이즈도 많아지게 되어 성능도 많이 떨어진다. 반면에 통계기반 음차표기 모델에 동적 윈도우(STM+DW)를 적용하면, 현저한 성능 향상(약 21%)을 보일 뿐만 아니라 문장 길이가 길어져도 성능은 크게 떨어지지 않는다. 동적 윈도우 기법으로 도달할 수 있는 최고 성능을 측정하기 위하여 실험데이터에서 미리 중국어 음차표기 단어의 길이를 측정하고, 그 길이를 윈도우 크기로 설정하여 성능을 측정하였다. 그 결과, STM(baseline)에 비해 약 23%정도 성능이 향상되었다. 이는 동적 윈도우를 적용했을 때와 거의 비슷한 성능 향상이다. 즉 동적 윈도우를 적용하면 윈도우 기법으로 도달할 수 있는 최고 성능에 가까운 성능을 낼 수 있다. 그러나 동적 윈도우만 적용하면 시간적인 복잡도가 지나치게 커지는 단점이 있다.

표 8에서 토크나이저 기법(STM+TOK)을 적용했을 때, 통계기반 음차표기 모델(STM)에 비하여 3%정도 성능이 향상되었음을 알 수 있다. 비록 많은 성능향상을 가져오지는 못했지만 3%에는 대부분 동적 윈도우 기법에서 해결하지 못한 문제들을 포함한다. 그러므로 동적 윈도우와 토크나이저를 동시에 적용한 결과(STM+DW+TOK), 동적 윈도우(STM+DW)만 적용했을 때 보다 3%정도 성능이 더 향상되었다.

표 8. 신뢰성 평가를 위한 실험 결과 (%)
Table 8. Experiment Result for Liability Reliability Evaluation

	단어 정확률	문자 정확률	문자 재현율
STM (Baseline)	75.33	86.65	91.11
STM+DW	96.00	98.51	99.05
STM+TOK	78.66	85.24	86.94
STM+DW+TOK	**99.00**	**99.78**	**99.72**

시간 복잡도 평가를 실시한 표 9는 각 방법의 시간 복잡도에 대한 비교 실험이다. 표 9에서 동적 윈도우(STM+DW)만 적용했을 때, 시간 복잡도는 통계기반 음차표기 모델(STM)만 적용했을 때의 약 27배임을 알 수 있다. 그러나 동적 윈도우와 토크나이저(STM+DW+TOK)를 동시에 적용하면, 시간 복잡도가 원래의 1/5로 크게 줄어든다. 즉, 토크나이저 기법은 동적 윈도우 기법으로 해결하지 못하는 일부 문제를 해결함으로써, 성능을 조금 더 향상 시키고 동시에 시간 복잡도도 크게 줄여주는 역할을 한다.

표 9. 시간 복잡도 평가를 위한 실험 결과
Table 9. Experiment Result for Time Complexity Evaluation

	실행 시간
STM (Baseline)	5초 (5,751 밀리초)
STM+DW	2분 34초 (154,893 밀리초)
STM+TOK	4초 (4,574 밀리초)
STM+DW+TOK	32초 (32,751 밀리초)

4.3 다국어 음차표기 추출 기법의 성능 평가

음차표기가 있는 영어단어에 대하여 얼마나 효과적으로 정확한 음차표기를 추출하는지 성능을 측정하기 위하여 위에서 추출한 500개 단어 중에서 음차표기가 있는 영어단어 104개를 선택하여 실험을 하였다.

성능 평가는 크게 두 가지로 하였다. 하나는 여러 개 음차표기 후보들을 모두 찾아주는 성능이고, 다른 한 가지는 상위 N개를 추출하여 그 중에 정답이 있으면 정확한 것으로 간주하고 정확률만 측정하였다.

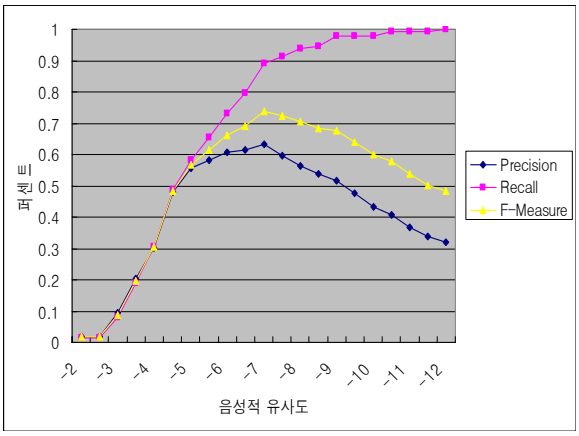


그림 7. 음성적 유사도를 이용한 정확한 음차표기 추출
Figure 7. Exact Phonetic Translation Extraction using Sound Similarity

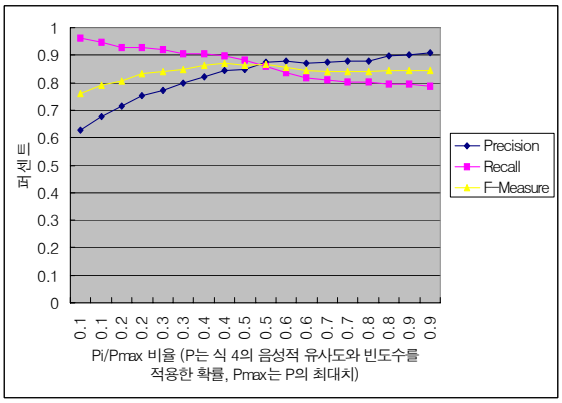


그림 8. 식(9)의 Pi확률을 이용한 음차표기 추출
Figure 8. Phonetic Translation Extraction using Pi Probability of (Formula 9)

먼저 정확한 음차표기 후보를 모두 추출하는 실험에 대하여 설명한다. 첫 번째 실험은 음성적 유사도만 이용하여 특정 임계값을 설정하고, 그 임계값보다 큰 음차표기를 모두 추출하여 성능을 측정하였다. 그림 7는 음성적 유사도의 임계값의 변화에 따른 성능 변화 그래프이다. 재현율은 임계값이 낮아질수록 꾸준히 올라가지만 정확률은 임계값이 -7.5보다 작아지면 떨어진다. 이는 임계값으로만 정확한 음차표기를 추출하였을 때 정확한 음차표기가 아닌 것도 같이 추출이 되기 때문이다.

두 번째 실험은 (식 9)의 P_i를 이용한 실험이다. P_i는 음성적 유사도와 빈도수를 동시에 고려한 확률이다. 정확한 음차표기 일수록 음성적 유사도가 높고 빈도수도 상대적으로 높기 때문에 정확한 음차표기의 확률이 다른 음차표기의 확률보다 높게 된다. 먼저 P_i로 각 음차표기에 대하여 정렬하고 P_i의 최대치 P_{max}를 설정하고 P_i/P_{max}의 값을 특정 임계값으로 설정하고 이 임계값보다 낮은 은 음차표기는 제거한다. 그림 8은 P_i/P_{max}의 값을 이용하여 정확한 음차표기를 추출하는 성능 그래프이다.

표 10. 음차표기 성능 비교
Table 10. Phonetic Translation Performance Comparison

방 법	정확률	재현율	F-Measure
음성적 유사도 이용	63.19%	88.97%	73.90%
(식 3.8)의 Pi 비율 이용	84.52%	89.76%	87.06%

표 10은 음성적 유사도와 (식 9)의 P_i비율을 이용한 정확한 음차표기 추출 성능을 비교한 것이다. 음성적 유사도만 이용했을 때에 비하여 P_i비율을 이용했을 때 정확률은 21% 가까이 향상되었으며 재현율도 1%가까이 향상 되었다. 그 이유는 음차표기가 아닌 것들은 정확한 음차표기에 비하여 음성적 유사도도 낮고 빈도수도 많이 낮기 때문이다.

5. 결 론

본 논문에서는 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 웹 콘텐츠에서 사용자 의견 정보들을 자동 추출 및 분석함으로써, 긍정/부정 의견별로 검색 및 통계를 확인할 수 있는 의견 검색 서비스를 제공할 수 있는 기법을 제안하였고, 제안한 기법을 이용하여 긍정/부정 의견 정보를 추출하는 기능의 성능 평가, 다국어 정보 검색을 위한 동적 윈도우 기법과 토크나이저 기법을 적용한 성능 평가, 그리고 정확한 다국어 음차표기를 추출 기법에 성능 평가를 실시하였다.

성능 평가 결과, 첫 번째로 의견 정보추출 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. 두 번째로 동적 윈도우 기법과 토크나이저 기법에서 신뢰성 평가를 위한 다양한 조합의 실험에서 동적 윈도우와 토크나이저를 동시에 적용(STM+DW+TOK)한 결과, 동적 윈도우(STM+DW)만 적용했을 때 보다 3%정도 성능이 더 향상되었고, 또한 시간 복잡도 평가를 위한 실험결과에서는 동적 윈도우와 토크나이저(STM+DW+TOK)를 동시에 적용하면, 시간 복잡도가 원래의 1/5로 크게 줄어들었다. 세 번째로 다국어 음차표기 추출기법에서 음성적 유사도만 이용했을 때에 비하여 Pi비율을 이용했을 때 정확률은 21%가까이 향상되었으며 재현율도 1%가까이 향상 되었다.

제안된 방법은 인터넷 상에 존재하는 여러 웹사이트들에 흩어져 있는 사용자 의견 정보들을 자동 추출 및 분석하여, 특정 키워드에 대한 다른 사용자들의 의견을 손쉽게 한눈에 검색 및 모니터링 할 수 있으며, 기존에 다른 사용자들의 의견을 검색하기 위해서 들였던 많은 시간을 크게 단축시킬 수 있다. 또한, 각 회사의 마케팅 담당자나 주식 투자자, 기업 가치 평가자 등은 방대한 인터넷 상에서 존재하는 해당 기업이나 물품에 대한 여러 사용자들의 의견을 한눈에 확인할 수 있으며, 기존에 사용자들의 의견을 알기 위해서 실시했던 설문조사나 컨설팅 회사에 들였던 비용을 대폭 줄일 수 있다.

참 고 문 헌

[1] 주해종・박영배, "모니터링 검색엔진을 위한 웹 콘텐츠 마이닝 시스템 설계," *한국통신학회 논문지*, 제34권 제2호 , pp.53-60, 2009,

[3] 장남식, 홍성완, 장재호, *데이터마이닝*, 대청, 2007.

[4] HaeJong Joo・YoungBae Park, "Design of Web Contents Mining System for Monitoring Search Engine", *KICS*, Vol. 34, No. 2, February, pp.53-60, 2009..

[5] NamSik Jang, SungWan Hong, JaeHo Jang, *Data Mining*, DaeChung, pp.32-56, 2007.

[6] S. Anand, D. Bell, J. Hughes, "The Role of Domain Knowledge in Data Mining," *CIKM95*, 1995.

[7] S. Anand, J. Hughes, "Hybrid Data Mining Systems: The Next Generation," *PAKDD '98*, Melbourne, Australia, pp. 13-24, 1998.

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

[9] J. Berry, G. Linoff, *Data Mining Techniques: For Marketing, Sales, and Customer Support*, John Wiley & Sons, 1997.

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

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

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

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

[14] Lee, J.S. and K.S.Choi, "Enflish to Korean Statistical transliteration for information retrieval," *Journal of Computer Processing of Oriental languages*, Vol. 12, No. 1, pp. 17-37, 1998.

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

[16] GotoI., N. Kato, N. Urataniand T. Ehara, "Transliteration Considering Context Information Based on the Maximum Entropy Method," *Proceedings of MT-Summit* IX, 2003.

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

[18] Virga Paola and Khudanpur, Transliteration of Proper Names in Cross-Lingual Information Retrieval, *ACL 2003's Workshop on Multilingual and Mixed-language Named Entity Recognition*, 2003.

[19] Dorre J., Gerstl, P., and Seiffert, R., "Text Mining:Finding Nuggets in Mountains of Textual Data," *Proceedings of the fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1999.

저 자 소 개

김영춘(Young-Choon Kim)
1987년 : 대전공업대학교 전기공학과
1989년 : 명지대학교 전기공학과(공학석사)
1997년 : 명지대학교 전기공학과(공학박사)
2006년～현재 : 공주대학교 기계자동차공학부 교수

관심분야 : 전기자동차전력변환, 전장제어, 하이브리드 자동차
Phone : 041 - 521 - 9274
E-mail : yckim59@kongju.ac.kr

주해종(Hae-Jong Joo)
2008년 : (美)Cumberland University (교육학박사)
2010년 : 명지대학교 컴퓨터공학과 (공학박사)
1997년～2005년 : 대원대학 멀티미디어과 조교수
2010년～현재 : 동국대학교 산학협력중심대학 교수

관심분야 : 데이터엔지니어링, IT융합기술, 유비쿼터스 비즈니스
Phone : 070 - 8680 - 7006
E-mail : hjjoo@dongguk.edu

최혜길(Hae-Gill Choi)
1980년 : 연세대학교 생화학과 졸업
1986년 : Northern Illinois Univ. (MS)
1999년 : 충남대학교 컴퓨터과학과 (이학박사)
2002년～현재 : 경희사이버대학교 정보통신학과 교수

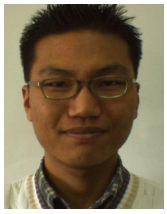
관심분야 : 멀티미디어DB, 소프트웨어공학, e-learning 표준화
Phone : 02 - 3299 - 8542
E-mail : hgchoi@khcu.ac.kr

조문택(Moong-Taek Cho)
1988년 : 명지대학교 전기공학과
1990년 : 명지대학교 전기공학과(공학석사)
1999년 : 명지대학교 전기공학과(공학박사)
2006년 ~ 현재 : 대원대학전기전자계열 부교수

관심분야 : 신재생에너지, 시뮬레이션
Phone : 043 - 649 - 3213
E-mail : mtcho@mail.daewon.ac.kr

이상용(Sang-Yong Rhee)
1982년 : 고려대 산업공학과 졸업
1984년 : 고려대 산업공학(공학석사)
1992년 : 포항공대 산업공학(공학박사)
2004년 ~ 현재 : 경남대학교 컴퓨터공학부 교수

관심분야 : 컴퓨터 비젼, 뉴로-퍼지, 지능로봇, 생체인식
Phone : 055-249-2706
E-mail : syrhee@kyungnam.ac.kr



김영백(Young-Baek Kim)
2005년 : 경남대학교 컴퓨터 공학부 졸업.
2007년 : 경남대 대학원 컴퓨터공학과 (공학석사)
2007 ~ 현재 : 경남대 대학원 컴퓨터공학과 박사과정

관심분야 : 영상처리, 컴퓨터 비젼, 증강현실
Phone : 010 - 5779 - 4783
E-mail : baroaleum@gmail.com