블룸 필터를 이용한 감성 웹 문서 크롤링 알고리즘

나철원○, 온병원*
군산대학교 소프트웨어융합공학과
ncw0034@kunsan.ac.kr, bwon@kunsan.ac.kr

A Bloom filter-based Sentiment-aware Web Crawling Algorithm

Chul-Won Na○, Byung-Won On
Department of Software Convergence Engineering, Kunsan National University

요 약

최근 빅 데이터와 인공지능의 발달과 함께 감성 분석에 대한 연구가 활발해지고 있다. 더불어 감성 분석을 위한 긍/부정 어휘가 풍부한 텍스트 문서들에 대한 수집의 필요성도 높아지고 있다. 본 논문은 긍/부정 어휘가 풍부한 텍스트 문서들을 수집하는 기존의 수집 방법에 대한 문제점에 대하여 해결방안을 제시한다. 기존의 수집 방법으로 일단 모든 URL들을 저장하고 필터링 과정을 거쳐 긍/부정 어휘가 풍부한 텍스트 문서들을 수집하고자 한다면 불필요한 텍스트 문서 저장과 필터링 과정에서 메모리와 시간을 낭비하게 된다. 기존의 수집 방법에 블룸 필터라는 자료구조를 적용시켜 메모리와 시간을 낭비하게 되는 문제점을 해결하고자 한다.

주제어: 웹 크롤링, 블룸 필터, 감성분석

1. 서론

최근 빅 데이터와 인공지능의 발달과 더불어 사용자의 감성까지 읽는 감성 분석에 대한 연구도 활발해 지고 있다. 2018년 3월 휴마트컴퍼니가 출시한 머신러닝 기반 감성 분석 솔루션 감성스캐너는 300자 가량 고민을 입력하면 감성 상태를 자동 분석해준다. 위와 같은 연구를 진행하기 위해 긍/부정 어휘가 풍부한 텍스트 문서를 필요로 하는 경우가 있다. 대부분의 텍스트 문서들은 웹 페이지에 저장되어 있다. 웹 페이지에서 긍/부정 어휘가 풍부한 텍스트 문서를 수집하기 위한 기존의 방법은 다음과 같다. 특정 URL에 접근하여 웹 페이지에 있는 정보들을 원하는 형태로 가져오는 크롤링이라는 과정을 이행한다. 일단 URL을 크롤링하여 모든 텍스트 문서들을 저장한다. 저장된 텍스트 문서들 중 긍/부정 어휘가 풍부한 텍스트 문서들을 식별해낸다. 식별해 내기 위하여 저장된 전체 텍스트 문서들을 스캔한다. 스캔한 텍스트 문서의 단어에 대하여 감성 사전에 있는지 확인한다. 이러한 과정을 거치게 되면 불필요한 텍스트 문서도 저장이 되어 메모리낭비와 긍/부정 어휘가 풍부한 텍스트 문서들을 식별해 내는 과정에서 많은 시간이 소요가 되는 문제점이 발생한다.

본 논문에서는 위에서 언급한 문제점을 해결하기 위한 '블룸 필터를 이용한 감성 웹 문서 크롤링 알고리즘(A Bloom filter-based Sentiment-aware Web Crawling Algorithm)'을 제안한다. 기존의 수집 방법과 차이점은 크롤링을 할 때 버튼 하워드 블룸(Burton Howard Bloom)이 제안한 블룸 필터(Bloom filter)[1]라는 자료구조를 적용하여 긍/부정 어휘가 많다고 판단되는 텍스트 문서들만을 식별하여 저장한다는 것이다. 본 논문에서 제안하는 방안을 '감성분석 웹 크롤링'이라고 명명한다. '감성분석 웹 크롤링'을 하게 되면 텍스트 문서들을 저장하기 전에 식별하기 때문에 메모리 사용 공간을 아낄 수 있다. 더불어 모든 텍스트 문서들을 저장 한 후 따로 긍/부정 어휘가 풍부한 텍스트 문서들을 식별하는 과정을 할 필요가 없기 때문에 텍스트 문서를 스캔하는 시간과 스캔한 문서의 단어에 대하여 감성 사전에 있는지 확인하는 시간도 절약할 수 있는 이점이 있다. 기존의 방법보다 메모리와 시간을 절약하여 긍/부정 어휘가 풍부한 텍스트 문서를 수집할 수 있게 된다.

본 논문의 구성은 다음과 같다. 2장에서는 본 연구와 관련 있는 연구를 정리하여 소개하였다. 3장 제안방안에서는 '블룸 필터를 이용한 감성 웹 문서 크롤링 알고리즘'에 대해 구체적으로 설명한다. 4장에서는 실험 환경 및 결과에 대해 자세히 논의한다. 5장에서 결론 및 향후 연구 방향을 다룬다.

2. 관련 연구

온라인 세계에서 엄청난 양의 고객 의견(VOC; Voice Of Customer)이 생성되고 있다. VOC는 제품에 대한 고객의 의견을 보기 위해 마케팅 또는 제품 디자인 부서에 매우 유용하다. 이러한 온라인 리뷰 사이트의 VOC를 수집, 저장 및 분석하여 전자 제품에 대한 고객 선호도에 어떤 요소가 영향을 미칠 수 있는지 확인한다. 이 연구는 VOC 데이터 분석 기법에 대하여 제안한다(Chi-Hwan Choi, 2013). 제안 방안은 다음과 같다. 1) 리뷰 사이트

- 이 논문은 2016년도 정부(미래창조과학부)의 한국연구재단의 중견연구자지원사업(No. NRF-2016R1A2B1014843)의 연구비 지원으로 수행하였습니다. 이 논문은 2017년 정부(과학기술정보통신부)의 재원으로 한국 연구재단의 지원을 받아 수행된 연구임(NRF-2017M3C4A7068188)

* 교신저자 : 온병원

의 URL을 선택한다. 2) 웹 크롤링 후 결과를 데이터베이스에 저장한다. 3) 관심 있는 주제와 관련이 없는 페이지 또는 데이터를 필터링한다. 4) 필터링 후 자연어 처리 모듈은 수집 된 문장에서 의미 있는 단어를 식별한다(단어가 제품이름, 브랜드 이름, 정서 관련 단어와 일치하는 경우 의미 있는 단어)[2]. 이 연구에서 제안방안 1)과 2)를 보면 일단 URL을 크롤링하여 웹 페이지의 데이터를 수집하고 데이터베이스에 저장한다. 그 후 저장되어 있는 전체 데이터에서 필요 없는 데이터들을 필터링을 한다. 그러나 본 논문에서 제안하는 방안은 웹 페이지를 크롤링하여 데이터를 저장하기 전에 관심이 있는 데이터를 식별하여 저장한다는 점에서 메모리와 시간을 절약할 수 있다는 차이가 있다. 본 논문에서 관심이 있는 데이터는 긍/부정 어휘가 풍부한 텍스트 문서이다.

경제주체들의 경기상황에 대한 판단 및 전망은 경기변동에 영향을 미치므로 경기심리지수와 거시경제지표들 간에는 밀접한 관련성을 나타내는 것으로 알려져 있다. 이 연구는 비정형데이터에서 정보를 추출해 경기심리지수를 생성하고, 경제분석에서의 활용 가능성을 검토하였다.(송민채, 2017) 제안 방안은 다음과 같다. 1) 데이터 수집, 2) 전처리, 3) 감성사전 구축, 4) 경기심리지수 생성 , 4)을 이행할 때 전처리하여 추출된 어휘들이 감성사전에 등록되어있는지 확인하는 작업을 한다[3].

최근 우리나라는 사회적 요인에 의한 재난이 빈번하게 발생하고 있다. 어떤 위기가 도시민들을 위협할지 예측하기 어려워 우려가 높아지고 있다. 본 연구에서는 텍스트 클러스터링 분석과 오피니언 마이닝 분석을 통하여 사회적 재난에 대해 정신적 충격과 불안감을 평가하였다(서민송, 2017). 제안 방안은 다음과 같다. 1) 트윗 데이터를 수집, 2) 텍스트 클러스터링 및 오피니언 마이닝(Opinion Mining) 수행, 3) 감성 분석[4].

3. 제안방안

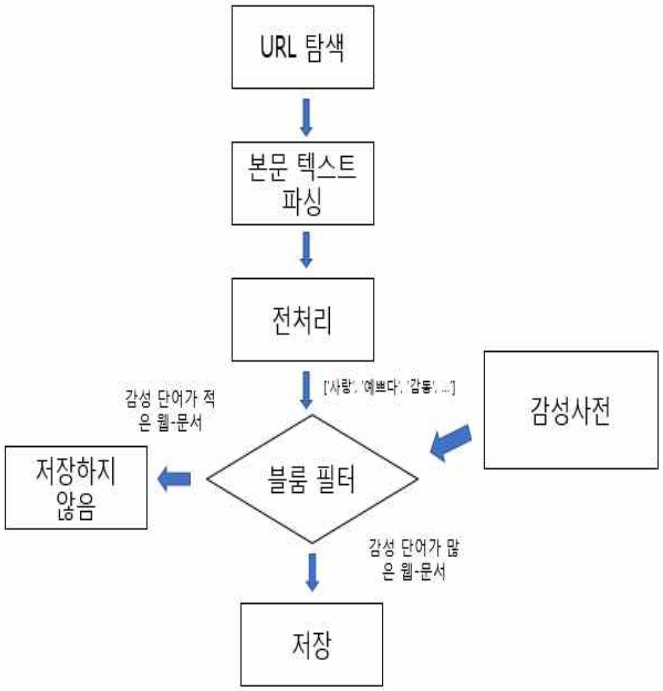
3.1 문제 정의

본 논문에서는 웹 문서를 크롤링 하는 동안 긍/부정 어휘가 풍부한 텍스트 문서들을 보다 빠르고 정확하게 수집하기 위해 블룸 필터를 이용하여 감성 웹 문서 크롤링 알고리즘을 제안한다. 일반적으로 웹 문서를 파싱(Parsing)해오는 크롤링에서 텍스트 문서를 스캔하는 시간과 스캔한 문서의 단어에 대하여 감성 사전에 있는지 확인하는 시간이 발생한다. 감성 웹 문서 크롤링을 하였을 때 추가적으로 발생하는 시간을 최소화하기 위하여 처리 속도가 빠르고 정확도가 높은 블룸 필터를 적용시킨다. 제안방안은 다음과 같다.

제안방안은 (1) 텍스트 문서 크롤링 (2) 긍/부정 어휘가 풍부한 텍스트 문서 식별 (3) 감성스코어 계산 및 텍스트 문서 저장 등으로 구성된다. 실험에 있어서 크롤링은 deque를 이용한 너비 우선 탐색인 Breath First Search(BFS) 알고리즘을 사용하고, 크롤링 코드에 블룸 필터를 적용한다. 크롤링의 첫 시작 URL이라고 불리는 SEED_URL은 네이버 뉴스, 중앙일보, 한겨레 신문 총 3개의 URL을 사용하였고 최대 탐색 URL의 수를 100개로 제한하여 진행한다.

SEED_URL
- 네이버뉴스 : https://news.naver.com/
- 중앙 일보 : https://joongang.joins.com/
- 한겨레 신문 : http://www.hani.co.kr/

(그림 1)은 전체적인 제안방안의 흐름도를 설명한다.



(그림 1) 제안방안의 흐름도

(그림 1)을 보면 URL을 수집한 후 본문 텍스트를 원하는 형태로 가져오는 파싱을 한다. 파싱한 텍스트 문서를 전처리하여 블룸 필터에 적용한다. 감성 단어(긍/부정 어휘)가 많다고 판단되면 저장하고 그렇지 않으면 저장하지 않는다.

3.2 텍스트 문서 크롤링

웹 페이지에서 텍스트 문서를 수집하기 위해 크롤링 코드를 구현한다. (그림 2)은 크롤링 알고리즘 중 BFS 알고리즘을 설명한다. BFS 알고리즘을 통하여 SEED_URL을 시작으로 하이퍼링크가 걸려있는 URL들을 탐색한다. 탐색한 URL들에 접근하여 텍스트 문서를 가져온다.

```
Crawling algorithm (BFS)
Input: SEED_URL
Procedure:
[1] enqueue(url_queue, SEED_URL)
[2] if (not lenth(url_queue) > 99)
[3]     while (not empty(url_queue))
[4]         url = dequeue(url_queue)
[5]         page = crawl_page(url)
[6]         euqueue(crawled_pages, (url, page))
[7]         url_list = extract_urls(page)
[8]         for u in url_list
[9]             enqueue(links, (url, u))
[10]            if (u not visit url_queue and (u,-) not in crawled_pages)
[11]            enqueue(url_queue, u)
[12]        reorder_queue(url_queue)

Function description:
  enqueue(queue, element)  : qppend element at the of queue
  dequeue(queue)           : remove the element at the beginning
                             of queue and  return it
  reorder_queue(queue)     : reorder queue using information in links
```

(그림 2) Breath First Search Algorithm

(그림 2)의 전체적인 흐름은 먼저 SEED_URL을 큐에 넣는다. 큐가 비어있지 않을 경우 큐의 가장 앞에 있는 URL을 내보내고 하이퍼링크가 걸려 있는 인접한 URL들 중 아직 방문하지 않은 URL을 큐에 넣는다. 큐가 비어있게 될 때까지 위의 과정을 반복한다.

3.3 긍/부정 어휘가 풍부한 텍스트 문서 식별

크롤링을 하여 파싱한 텍스트 문서들을 전처리하여 블룸 필터에 적용한다. 각 텍스트 문서에서 감성어(긍/부정 어휘)를 식별해 준다.

3.3.1 블룸 필터(Bloom filter)

블룸 필터는 m개의 비트 배열(bit array)과 k개의 해시 함수(hash function)를 이용해서 검사하고자 하는 원소가 이미 검사 되어 있는지를 판단한다. 이 때 사용되는 해시 함수란 데이터의 효율적 관리를 목적으로 임의의 길이의 데이터를 고정된 길이의 데이터로 매핑(mapping)하는 함수이다. 제한적인 비트 수를 가지고 모든 원소들에 대해서 표기를 해야 한다. 따라서 실제로 원소가 집합 내에 존재하지 않는데 존재한다고 판단할 수 있는 'False Positive'가 발생할 수 있다. 반면에 실제로 원소가 집합 내에 존재하는데 존재하지 않는다고 판단하는 'False Negative'는 결코 발생하지 않는다는 특징이 있다[1]. 간단한 예시를 통하여 블룸 필터에 대해서 더 자세히 알아본다. (그림 3)는 블룸 필터의 예시를 설명한다. 예시에서는 보다 쉬운 설명을 위하여 해시 값을 결과 값이라 명명한다.

(그림 3) 블룸 필터의 예

(그림 3)는 10개의 비트 배열과 3개의 해시 함수를 이용하는 예시이다. 각 비트에는 0 또는 1의 값이 들어간다. 그리고 특정 원소를 삽입할 때, 3개의 해시 함수를 사용한다. 예를 들어 최초의 아무 데이터도 없는 상태의 배열은 모든 인덱스의 값이 0이다. 그리고 "a"이라는 데이터를 삽입한다고 하면, 3개의 해시 함수를 거쳐서 결과 값을 얻어낸다. 얻어낸 값이 3, 5, 7이라고 한다. 그리고 다음으로 "b"라는 데이터를 삽입한다. 마찬가지로 3개의 해시 함수를 통해 얻어낸 결과 값에 맞게 넣어준다. b의 결과 값이 1, 5, 8 이라고 한다면 5는 겹친다. 그런 경우에는 그냥 그대로 둔다. 이제 검색할 데이터는 "c"라고 한다. 우선 "c"라는 데이터에 대해 3개의 해시 함수 결과 값이 (2, 6, 8) 라고 하면 해당 배열의 값을 확인해보니 2번 인덱스: 0, 6번 인덱스: 0, 8번 인덱스: 1 이다. 그렇다면 이 집합에 "c"라는 데이터가 들어있지 않다는 사실은 확실하다. 이처럼 'False Negative'는 절대 발생하지 않는다. 다음 데이터도 검색해본다. 이번에 검색할 데이터는 "d"다. "d"의 해시 함수 결과 값이 (5, 7, 8)이라고 하면 해당 배열의 값을 확인해보니 5번 인덱스: 1, 7번 인덱스: 1, 8번 인덱스: 1 이다. 모두 '1'의 값이 들어가 있다. "d"라는 데이터를 삽입한 적이 없었다. 하지만 있다고 나온다. 따라서, 특정 데이터가 들어있다 라는 결과 값에 대해서는 100% 신뢰할 수 없다. 이처럼 'False Positive'는 발생할 수 있다.

3.3.2 KNU 한국어 감성사전

감성사전 구축을 위하여 다양한 연구가 진행되어왔다. 그 중에서 한국어를 대상으로 실제 적용 가능하고 도메인에 의존적이지 않는 'KNU 한국어 감성사전'을 사용한다(온병원 외 2인, 2018). 'KNU 한국어 감성사전'의 구축 알고리즘은 다음과 같다. 국립국어원에서 발생하는 '표준국어대사전'(국립국어원 2018)을 구성하는 모든 단어와 그에 해당하는 뜻풀이(Gloss)를 수집하고 정제한다. 수집된 단어들에 대하여 형태소 분석을 실시하고 형용사, 부사, 동사, 명사를 품사로 갖는 단어들에 대한 뜻풀이를 추출한다. 본 논문에서는 뜻풀이가 지니는 감성에 따라 긍정의 의미를 지니는 뜻풀이에서는 긍정에 관한 감성어를, 부정의 의미를 지니는 뜻풀이에서는 부정에 관한 감성어를 추출하기 위해서 뜻풀이를 분류한다. 분류를 위한 학습 데이터는 추출된 뜻풀이의 형용사, 부사 그리고 일부 동사에 대해 3명의 투표자가 수작업으로 구축한다. 학습 데이터는 뜻풀이와 감성 값으로 구성되며, 뜻풀이에 대한 감성 값은 긍정, 부정, 중립으로 나뉜다. 구축된 학습 데이터를 딥러닝 기법 중 하나인 Bi-directional LSTM(Bi-LSTM)에 학습 시킨 후 학습된 모델을 통해 나머지 동사와 명사에 대한 뜻풀이의 감성을 분류한다. 분류된 뜻풀이를 통해 본 연구에서는 1-gram, 2-gram, 어구(n-gram), 문형에 해당하는 감성어를 수작업으로 추출한다.

감성어를 추출한 후 3명의 평가자는 각 감성어의 감성

정도(degree)와 도메인에 독립적인지에 대한 여부를 판단한다. 각 단어에 대한 감성 정도와 도메인에 독립적인지에 대한 판단이 3명의 평가자가 모두 일치하는 경우, 해당 감성 정도를 감성어의 감성 정도로 부여한다. 3명의 의견이 모두 일치하지 않는 경우에는 해당 단어에 대해 평가자가 토론 후 해당 단어에 대한 감성 정도와 도메인에 독립적인지에 관한 여부를 결정한다.

'KUN 한국어 감성사전'에 대하여 더 자세한 사항이 궁금하다면 데모시연과 깃허브에서 다운로드가 가능한 참고문헌 [6]을 참고하면 좋을 것 같다.

3.4 감성스코어 계산 및 텍스트 문서 저장

3.3.1에서 설명한 블룸 필터를 적용하기 위해 기준이 될 감성어 들은 'KNU 한국어 감성사전' 에 등재되어 있는 감성어 들을 사용한다. 블룸 필터의 비트 배열은 50만개를 사용하고 해시 함수는 7개를 사용한다. 블룸 필터에 의해 나온 값을 'False Negative'가 결코 발생하지 않고 'False Positive'는 발생할 수 있기 때문에 블룸 필터를 적용해 나온 0은 'Nope', 1은 'Probably'로 명명한다.

감성스코어(Sentiment_감성스코어)는 각 텍스트 문서에서 나온 'Nope'의 수와 'Probably'의 수를 더한 후 'Probably'의 수에 나눈 것이다.

$$\text{감성스코어(문서)} = \frac{\#\ of\ 'Probably'}{\#\ of\ 'Nope' + \#\ of\ 'Probably'}$$

긍/부정 어휘가 풍부한 텍스트 문서라고 판단되는 기준은 스코어가 0.015 이상일 경우라고 가정한다. 감성스코어가 0.015 이상이 되면 텍스트 문서를 저장하고 그렇지 않으면 저장하지 않는다. 세 개의 SEED_URL인 네이버뉴스, 중앙일보, 한겨레 신문의 감성스코어를 확인한다. 세 개의 SEED_URL 모두 감성스코어가 0.015 이하일 경우 그래프의 맨 위 꼭짓점을 이었을 때 기울기가 완만하다. 기울기가 완만하다는 의미는 곧 대부분의 텍스트 문서는 감성스코어가 0.015이하라는 뜻이다. 따라서 긍/부정 어휘가 풍부한 텍스트 문서의 기준을 감성스코어가 0.015 이상일 경우로 정한다.

4. 실험 환경 및 결과

(표 1) 실험 환경

Ubuntu	Python	BeautifulSoup	KoNLPy
15.1	3.4.3+	4.5.3	0.5.1

4.1 실험 환경

리눅스(Linux) 환경인 우분투(Ubuntu)에서 파이썬(Python)으로 구현하였다. URL에 접근하여 파싱을 할 때에는 파이썬의 라이브러리인 BeautifulSoup를 사용하였다. 블룸 필터에 적용하기 위해 파싱한 텍스트 문서들을 전처리한다. 전처리하는 과정에서 'KNU 한국어 감성사전'에 등재되어 있는 단어들은 어근으로도 따로 등재되어있어 KoNLPy의 mecab을 사용하여 어근만 추출한다. SEED_URL을 넣은 후 각 100개의 URL을 수집하지만 URL 안에 본문 텍스트가 있을 수도 있고, 없을 수도 있어 파싱된 텍스트 문서의 수는 다르다.

4.2 결과

웹 사이트는 업데이트되기 때문에 결과가 달라 질 수도 있다. 본 결과는 2018년 9월 3일 오후 4:15 기준 이다. 감성스코어가 0.015 이상이면 저장을 한다.

4.2.1 실행 시간

(표 2)와 (그림 3)은 '감성분석 웹 크롤링'의 각 부분별로 실행 시간을 설명한다.

(표 2) '감성분석 웹 크롤링'의 각 부분 별 실행 시간

	네이버뉴스	중앙일보	한겨레 신문
파싱	15,661	25,096	21,218
전처리	788	1,987	404
URL크롤링	379	159	162
기존 크롤링 실행 시간	16,828	27,242	21,784
블룸 필터 추가 실행 시간	16,828 + **90**	27,242 + **104**	21,784 + **73**

(url: 100개, 시간 단위: Millisecond; ms)

(표 2)에서 보는 것처럼 '감성분석 웹 크롤링'에 SEED_URL을 넣고 실행하였을 때의 부분 별 처리 시간을 알 수 있다. '감성분석 웹 크롤링'의 각 부분은 파싱, 전처리, URL 크롤링, 블룸 필터 이렇게 4 부분으로 나누었다. 각 처리 시간은 네이버뉴스는 15,661ms, 788ms, 379ms, **90ms**이고 중앙일보는 25,096ms, 1,987ms, 159ms, **104ms**이고 한겨레 신문은 21,218ms, 404ms, 162ms, **73ms**이다. 세 개의 SEED_URL 모두 4 부분의 처리 시간 중 블룸 필터의 처리 시간이 가장 적은 비중을 차지하는 것을 알 수 있다.

4.2.2 블룸 필터의 성능

블룸 필터의 특성상 'False Positive'가 발생할 수 있다. 그러나 'False Negative'는 결코 발생할 수 없기 때문에 'Nope'은 확인을 안 해보아도 되지만 'Probably'는 확인해 보아야 한다.

(표 3) 'Nope', 'Probably', '중복제거 Probably'의 수

뉴스	Nope	Probably	중복제거 Probably
네이버뉴스	25,526	763	171
중앙일보	56,201	2,346	227
한겨레 신문	11,556	451	124

(표 3)는 각 SEED_URL인 네이버뉴스, 중앙일보, 한겨레 신문의 'Nope'과 'Probably'와 중복제거를 한 'Probably'의 수를 나타낸다. 'Probably'라고 나온 감성어를 중복 제거하여 나온 535개를 수작업으로 'False Positive'가 발생하였는지 안 하였는지 확인해 보았다. 감성어 535개중 1개의 감성어가 'False Positive'가 발생하였다. 정답률은 전체 'Probably' 중에 'False Positive'가 발생하지 않은 비중이라 명명한다. (535-1)/535 = 0.9981 이므로 정답률은 99.81%가 된다. 정답률로 블룸 필터의 정확도를 판단한다. (그림 4)는 블룸 필터의 정확도를 설명한다.

(그림 4) 블룸 필터의 정확도

(그림 4)를 보면 감성어가 137개일 때까지 정답률은 100%를 유지하고 308개를 넘어갈 때부터 99.81%로 떨어지고 535개를 확인하였을 때에도 99.81%를 유지하였다. 99.81%로 높은 정확도를 확인할 수 있다.

4.2.3 '감성분석 웹 크롤링'의 결과 문서

(그림 5), (그림 7), (그림 9)은 각 네이버뉴스, 중앙일보, 한겨레를 SEED_URL로 사용한 '감성분석 웹 크롤링'의 각 결과 텍스트 문서 감성스코어를 설명한다.

(그림 5) 네이버뉴스

네이버뉴스가 SEED_URL일 경우 100개의 URL 중에서 총 56개의 텍스트 문서들을 파싱하여 전처리 후 블룸 필터까지 적용하였다. 감성스코어는 0.033부터 0.002까지 계산되었다. (그림 5)을 보면 감성스코어가 0.033에서부터 0.002까지 내림차순으로 정렬되어 있는 것을 볼 수 있다. 감성스코어 0.015 이상인 텍스트 문서의 개수는 총 15개이다.

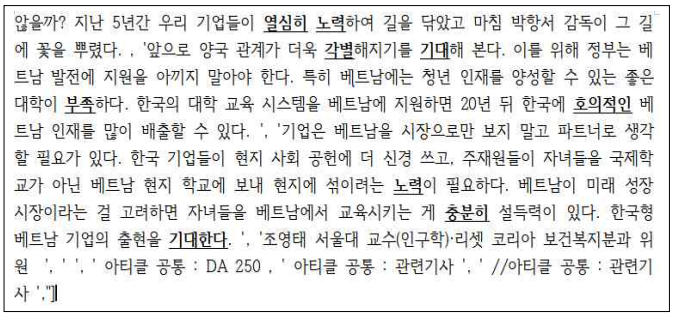
울시장을 박원순에게 흔쾌히 양보해서 신선한 충격을 줬다"며 "손학규 역시 지지난 총선에서 실패한 뒤 깔끔하게 정치은퇴를 선언했다. 한때 같은 당에 있었던 사람으로서 저는 참 가슴이 아팠다"고 말했다. ', '이어 "그대(손 대표) 말했지요. '앞으로 다른 방면으로 국가에 도움이 되는 일을 하겠다. 정치는 그만하겠다'고 했다"며 "그런데 누구 말대로 '페이크'였나 보다. 그냥 서울 아파트가 아니라 '강진토굴'로 들어가 '호시탐탐 기회'를 보고 있던 거다. 그러다가 타이밍이라는 것을 놓쳤다가 민주당을 탈당해 드디어 바른미래당 간판이 됐다"고 꼬집었다. ', '이어 안 전 의원에 대해서도 "'새 정치'는 분리수거 확실히 하고 구태정치를 그대로 답습했다"며 "독일 간다고 했다가 서울 있던 것을 들키자 36계 줄행랑을 치는 모습을 보자니 참 감동란 먹다가 토할 뻔 했다"고 비판했다. ', '전 전 의원은 "감동의 정치는 누가 할 수 있는가? 버릴 수 있는 사람, 내려놓을 수 있는 사람, 그리고 가진 것이 별로 없는 사람이 할 수 있다"며 바른미래당에 '올드보이의 귀환'이 아닌 '세대교체'가 절실했다고 강조했다. ', '최정아 동아닷컴 기자 cja0917@donga.com', '▶ 동아일보 단독', ' / ', '동아일보 공식 페이스북', '▶ 핫한 경제 이슈와 재테크 방법 총집결(클릭!)', '© 동아일보 & donga.com, 무단 전재 및 재배포 금지', ' 본문 내용 ', ']']

(그림 6) 감성스코어가 0.015 이상인 문서 중 일부분

(그림 6)은 '감성분석 웹 크롤링'으로 SEED_URL이 네이버뉴스인 경우 파싱한 문서중 감성어가 많다고 판단되는 감성스코어가 0.015 이상인 텍스트 문서를 저장하고 그 중 일부분을 캡처한 화면이다. 캡처한 부분에서 '충격', '아팠다', '도움', '바른', '감동' 등의 많은 감성어가 포함되어 있다는 것을 알 수 있다. 감성스코어 0.015 이상인 텍스트 문서의 약 0.9%정도는 5가지 이상의 감성어를 포함하고 있는 것을 확인할 수 있다.

(그림 7) 중앙일보

중앙일보가 SEED_URL일 경우 100개의 URL 중에서 총 66개의 텍스트 문서들을 파싱하여 전처리 후 블룸 필터까지 적용하였다. 감성스코어는 0.031부터 0.001까지 계산되었다. (그림 7)을 보면 감성스코어가 0.031에서부터 0.001까지 내림차순으로 정렬되어 있는 것을 볼 수 있다. 감성스코어 0.015 이상인 텍스트 문서의 개수는 총 15개이다.

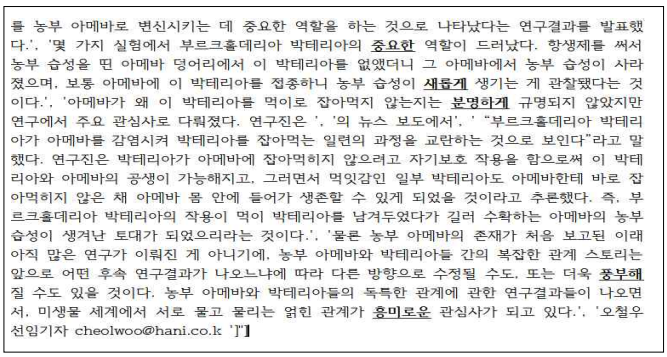

않을까? 지난 5년간 우리 기업들이 열심히 노력하여 길을 닦았고 마침 박항서 감독이 그 길에 꽃을 뿌렸다. , '앞으로 양국 관계가 더욱 각별해지기를 기대해 본다. 이를 위해 정부는 베트남 발전에 지원을 아끼지 말아야 한다. 특히 베트남에는 청년 인재를 양성할 수 있는 좋은 대학이 부족하다. 한국의 대학 교육 시스템을 베트남에 지원하면 20년 뒤 한국에 호의적인 베트남 인재를 많이 배출할 수 있다. ', '기업은 베트남을 시장으로만 보지 말고 파트너로 생각할 필요가 있다. 한국 기업들이 현지 사회 공헌에 더 신경 쓰고, 주재원들이 자녀들을 국제학교가 아닌 베트남 현지 학교에 보내 현지에 섞이려는 노력이 필요하다. 베트남이 미래 성장 시장이라는 걸 고려하면 자녀들을 베트남에서 교육시키는 게 충분히 설득력이 있다. 한국형 베트남 기업의 출현을 기대한다. ', '조영태 서울대 교수(인구학)·리셋 코리아 보건복지분과 위원 ', ' ', ' 아티클 공통 : DA 250 , ' 아티클 공통 : 관련기사 ', ' //아티클 공통 : 관련기사 ',"]

(그림 8) 감성스코어가 0.015이상인 문서 중 일부분

(그림 8)는 '감성분석 웹 크롤링'으로 SEED_URL이 중앙일보인 경우 파싱한 문서중 감성어가 많다고 판단되는 감성스코어가 0.015 이상인 텍스트 문서를 저장하고 그 중 일부분을 캡처한 화면이다. 캡처한 부분에서 '열심히', '노력' 두 번, '각별', '기대' 두 번, '부족하다', '호의적인', '충분히' 등의 많은 감성어가 포함되어 있다는 것을 알 수 있다. 감성스코어 0.015 이상인 텍스트 문서의 약 0.7%정도는 9가지 이상의 감성어를 포함하고 있는 것을 확인할 수 있다.

(그림 9) 한겨레 신문

한겨레가 SEED_URL일 경우 100개의 URL 중에서 총 47개의 텍스트 문서들을 파싱하여 전처리 후 블룸 필터까지 적용하였다. 감성스코어는 0.028부터 0까지 계산되었다. (그림 9)를 보면 감성스코어가 0.028에서부터 0까지 내림차순으로 정렬되어 있는 것을 볼 수 있다. 감성스코어 0.015 이상인 텍스트 문서의 개수는 총 15개이다.


를 농부 아메바로 변신시키는 데 중요한 역할을 하는 것으로 나타났다는 연구결과를 발표했다.', '몇 가지 실험에서 부르크홀데리아 박테리아의 중요한 역할이 드러났다. 항생제를 써서 농부 습성을 띤 아메바 덩어리에서 이 박테리아를 없앴더니 그 아메바에서 농부 습성이 사라졌으며, 보통 아메바에 이 박테리아를 접종하니 농부 습성이 새롭게 생기는 게 관찰됐다는 것이다.', '아메바가 왜 이 박테리아를 먹이로 잡아먹지 않는지는 분명하게 규명되지 않았지만 연구에서 주요 관심사로 다뤄졌다. 연구진은 ', '의 뉴스 보도에서', ' "부르크홀데리아 박테리아가 아메바를 감염시켜 박테리아를 잡아먹는 일련의 과정을 교란하는 것으로 보인다"라고 말했다. 연구진은 박테리아가 아메바에 잡아먹히지 않으려고 자기보호 작용을 함으로써 이 박테리아와 아메바의 공생이 가능해지고, 그러면서 먹잇감인 일부 박테리아도 아메바한테 바로 잡아먹히지 않은 채 아메바 몸 안에 들어가 생존할 수 있게 되었을 것이라고 추론했다. 즉, 부르크홀데리아 박테리아의 작용이 먹이 박테리아를 남겨두었다가 길러 수확하는 아메바의 농부 습성이 생겨난 토대가 되었으리라는 것이다.', '물론 농부 아메바의 존재가 처음 보고된 이래 아직 많은 연구가 이뤄진 게 아니기에, 농부 아메바와 박테리아들 간의 복잡한 관계 스토리는 앞으로 어떤 후속 연구결과가 나오느냐에 따라 다른 방향으로 수정될 수도, 또는 더욱 풍부해질 수도 있을 것이다. 농부 아메바와 박테리아들의 독특한 관계에 관한 연구결과들이 나오면서, 미생물 세계에서 서로 물고 물리는 얽힌 관계가 흥미로운 관심사가 되고 있다.', '오철우 선임기자 cheolwoo@hani.co.k ']"]

(그림 10) 감성스코어가 0.015이상인 문서 중 일부분

(그림 10)은 '감성분석 웹 크롤링'으로 SEED_URL이 한겨레인 경우 파싱한 문서중 감성어가 많다고 판단되는 감성스코어가 0.015 이상인 텍스트 문서를 저장하고 그 중 일부분을 캡처한 화면이다. '중요한', '새롭게', '분명하게', '풍부한', '흥미로운' 등의 많은 감성어가 포함되어 있다는 것을 알 수 있다. 감성스코어 0.015 이상인 텍스트 문서의 약 1.7%정도는 5가지 이상의 감성어를 포함하고 있는 것을 확인할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 메모리와 시간을 절약할 수 있는 '블룸 필터를 이용한 감성 웹 문서 크롤링 알고리즘' 방안을 제안하였다. 기존의 긍/부정 어휘가 풍부한 텍스트 문서를 수집하는 방법에 블룸 필터라는 자료구조를 적용하였다. SEED_URL을 입력 받아 최대 100개까지의 URL을 수집한다. 수집한 URL들은 파싱하여 전처리 후 블룸 필터에 적용한다. 블룸 필터에 의해 감성어가 많아 긍/부정 어휘가 풍부한 문서로 판단되면 저장한다. 블룸 필터를 사용하면 별도의 식별하는 과정을 할 필요가 없기 때문에 전체 텍스트 문서를 스캔하는 시간과 스캔한 문서의 단어에 대하여 감성 사전에 있는지 확인하는 시간도 절약할 수 있게 된다.

본 연구의 실험 결과 블룸 필터의 처리 속도와 정답률 면에서 우수한 성능을 보였다. '감성분석 웹 크롤링'의 처리 속도 중에서 블룸 필터가 차지하고 있는 비중은 네이버 뉴스는 약 0.5%, 중앙일보는 약 0.4%, 한겨레 신문은 약 0.3% 정도로 매우 적었고, 감성어 535개중 534개를 맞추어 99.81%라는 높은 정답률을 보였다.

향후 연구로는 블룸 필터의 기준이 되었던 'KNU 한국어 감성사전'에 좀 더 많은 감성어 들을 추가할 예정이다. 더불어 해시 조인(Hash Join)을 적용해보고 블룸 필터와 비교하여 좀 더 실행 속도가 빠르고 정확도가 높은 방안을 제시할 것이다.

참고문헌

[1] Bloom filter, https://ko.wikipedia.org/wiki/블룸_필터, (Accessed, 2018).

[2] Chi-Hwan Choi, Jeong-Eun Lee, Gyeong-Su park, Jonghwa Na, Wan-Sup Cho, "Sentiment Analysis for Customer Review Sites", The 3rd International Conference on Circuits on Circuits, CES-CUBE 2013, ASTL Vol. 25, pp.157 - 162, 2013.

[3] 송민채, 신경식, "뉴스기사를 이용한 소비자의 경기심리지수 생성", 지능정보연구, 제23권, 제3호, pp.1-27, 2017

[4] 서민송, 유환희, "오피니언 마이닝 기법을 이용한 사회적 재난의 시민 감성도 분석", 한국지형공간정보학회지, 제25권, 제1호, pp.37-46, 2017

[5] 블룸 필터, "https://blog.naver.com/cutup9999/22126281547", (Accessed, 2018).

[6] 온병원, 박상민, 나철원, "KNU 한국어 감성사전", "http://dilab.kunsan.ac.kr/knusl.html",(Accessed, 2018).