텍스트 랭크 알고리즘을 이용한 사용자 타임라인 요약 기법

(A User Timeline Summarization Technique using TextRank Algorithm)

안 인 석 † 김 현 우 † 김 형 주 ††
(In Seok An) (Hyunwoo Kim) (Hyoung-Joo Kim)

요 약 디지털 기기가 광범위하게 보급되면서 information stream이 정보 인지의 보편적인 수단으로 활용되고 있다. 트위터는 140자 미만의 단문을 서비스하는 마이크로 블로깅 서비스로 가장 인기 있는 소셜 미디어 서비스이다. 시간이 지남에 따라 트위터 사용자들은 구독하는 정보의 양이 많아진다. 결국 너무 많은 정보를 받게 되어 사용자가 정보를 모두 확인할 수 없는 상태가 된다. 본 연구에서는 텍스트 랭크 알고리즘이라는 자연어 처리 기법을 이용하여 트위터 타임라인에서 일어나는 정보 과다 현상을 해결하는 연구를 진행하였다. 타임라인을 이루고 있는 트윗들을 그래프로 모델링 한 후, 그래프 기반의 랭킹 알고리즘을 적용하여 정점의 스코어를 얻고, 그 스코어를 기반으로 산봉우리 개념을 그래프에 적용하여 타임라인을 요약하였다. 본 연구에서 제안한 텍스트 랭크를 이용한 타임라인 요약 방법이 기존의 빈도수 기반 요약 방법보다 효과적으로 타임라인을 요약할 수 있음을 실험을 통하여 확인하였다.

키워드 : 트위터, 타임라인, 요약, 자연어 처리, 키워드

Abstract As digital device has come into wide use, information streams have recently emerged as a popular means of information awareness. Twitter is one of the most popular micro-blogging service and social media with a limit of 140 characters. If a user follows Twitter accounts continuously, the user may subscribe information more than the user can process. In this paper, we apply TextRank algorithm to alleviate information overload in Twitter user's timeline. After modeling user timeline as a graph, we apply graph-based ranking algorithm to the graph of timeline. Based on the score of each vertex, we apply concept of summit to summarizing user timeline. The experimental results show that proposed method summarizes user timeline more effectively than existing method that rely mainly on frequency based method.

Key words : Twitter, Timeline, Summarization, NLP, Keyword

· 이 논문은 2011년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No. 020110030812). 본 연구는 BK-21 정보 기술 사업단의 연구결과로 수행되었음

† 비 회 원 : 서울대학교 컴퓨터공학부
isan@idb.snu.ac.kr
hwkim@idb.snu.ac.kr
(Corresponding author임)
†† 종신회원 : 서울대학교 컴퓨터공학부 교수
hjk@snu.ac.kr

논문접수 : 2012년 1월 4일
심사완료 : 2012년 3월 22일

정보과학회논문지: 데이타베이스 제39권 제4호(2012.8)

1. 서 론

최근 information stream이 정보 인지의 보편적인 수단으로 활용되고 있다. Information stream이란 트위터, 페이스북, 구글리더 혹은 다른 RSS 리더 어플리케이션처럼 웹 2.0 정보 공급원의 최신 업데이트를 받아 볼 수 있는 응용을 말한다. 웹 2.0 기술의 발달과 모바일 기기의 광범위한 보급으로 수많은 컨텐츠들이 온라인상에 쏟아져 나오고 있다. 지난 10년간 생성된 데이터보다 최근 2년간 생성된 데이터의 양이 훨씬 많으며, 2011년 한 해에만 1.8ZB의 디지털 정보가 쏟아져 나왔으며 이 수치는 점점 더 늘어날 것으로 예상되고 있다[1]. 그 결과 information stream을 이용하여 정보를 소비하는 사용자들은 자신이 구독하는 사용자가 늘어남에 따라 기

하급수적으로 늘어나는 정보량을 모두 처리 할 수 없게 되는 정보 과다 현상(information overload)을 겪게 된다. 따라서 대용량 정보들을 사람이 처리할 수 있는 정보량 이하로 요약해주는 기술이 필요하게 된다.

트위터는 트윗이라는 140자 이하의 단문을 발행 할 수 있는 마이크로 블로깅 서비스이다. 트위터에서 사용자들은 팔로우 관계를 형성하여 다른 사용자의 트윗을 구독 할 수 있다. 트위터는 2006년 3월 서비스를 시작한 이래로 꾸준히 사용자가 늘었으며 2011년 3월 1억 7천 5백만명의 사용자를 보유하고 있고, 하루 평균 46만명의 새로운 사용자가 꾸준히 유입되고 있다. 소셜 미디어에서 상대적으로 뒤쳐졌던 대한민국도 최근 트위터 사용자가 폭발적으로 늘고 있다. 2011년 11월 현재 대한민국을 지역기반으로 하고 있는 사용자의 숫자가 약 497만 명으로 집계되고 있으며 그 숫자는 빠르게 늘고 있다. 국내에서 트위터의 영향력도 갈수록 늘어나고 있다. 이렇게 트위터 사용자가 늘어남에 따라 트위터 상에 유통되는 트윗의 양도 폭발적으로 늘어나게 된다. 사용자는 점점 더 많은 사용자를 팔로잉하게 되고, 점점 더 많은 양의 트윗을 구독하게 된다. 그렇게 되면 사용자의 타임라인은 점점 더 많은 수의 트윗들로 채워지게 되고, 결국 사용자가 처리할 수 있는 한계량을 넘어서게 된다.

트위터에서도 실시간 트렌드라는 기능을 제공하여 선택된 지역에서 활발하게 전달되고 있는 트윗들의 주제를 보여주고 있다. 하지만 아직 대한민국은 대상 지역에 포함되지 않고 있으며 실시간 트렌드는 사용자의 타임라인이 아니라 사용자가 살고 있는 지역 전체에서 유통되고 있는 거시적인 주제를 의미하기 때문에 사용자 개개인의 관심사와 부합하지 않는다. 따라서 본 논문에서는 트위터 전체가 아니라 사용자 개개인의 타임라인을 이루고 있는 트윗들을 대상으로 키워드를 추출하여 정보 과다 문제를 해결하는 기법을 연구하였다.

본 논문의 구성은 다음과 같다. 2장에서 트위터 상의 정보를 다루는 연구들에 대해 살펴보고, 3장에서 텍스트 랭크 알고리즘에 대해서 상세히 다룬다. 4장에서는 트위터 환경에 맞게 텍스트 랭크 알고리즘을 적용하여 주제를 뽑아내는 시스템을 제안하며, 5장에서 해당 시스템의 성능을 평가하는 실험 방법과 평가에 대해 설명한다. 마지막으로 6장에서는 본 논문에서 제안한 키워드 추출 방법에 대한 결론과 향후 연구에 대해서 언급한다.

2. 관련연구

트위터가 기존의 매스 미디어를 대체하는 소셜 미디어로써의 기능을 갖게 되면서 트위터 상에 유통되는 정보를 다루는 연구들이 활발히 이루어지고 있다. 기존의 소셜 네트워크라는 측면의 트위터와 소셜미디어라는 측면의 트위터의 특성을 비교하여 미디어로써 트위터의 특징을 설명하는 연구가 진행되었다[2]. 미디어로써 트위터와 모바일 디바이스의 특징인 위치 정보를 결합하여 특정 지역에서 어떤 뉴스가 발생하였는지 알려주는 “TwitterStand”라는 시스템도 연구되었다[3].

트위터의 가장 큰 특징은 실시간 정보 전달에 있다. [4]에서는 트위터 상에 유통되고 있는 실시간 트윗들을 분석하여 검색 엔진의 성능을 향상시켰다. 사용자가 검색 쿼리를 입력했을 때, Twinner라는 시스템이 쿼리가 트위터에서 실시간으로 논의되고 있는 주제인지 판단하여, 검색의도가 뉴스 검색인지 판단한 뒤, 뉴스일 경우 검색 쿼리를 확장하여 보다 정확한 결과를 보여주도록 도와준다. 트위터의 이런 강력한 정보 전달 능력은 때로 잘못된 루머의 전파를 야기한다. [5]에서는 트위터 상에 유통되는 트윗들 중에서 정보성 트윗만을 판별하는 시스템을 제안하였다. 뉴스와 관련 된 트윗을 수작업으로 분류한 다음, 해당 뉴스 트윗의 신뢰도를 평가하였다. 트윗들을 기술할 수 있는 여러 가지 특성들을 정의하고, 이들을 이용해 기계학습 프로세스를 이용해 자동으로 트윗의 신뢰도를 평가해주는 연구를 하였다.

다수의 트윗에서 하나의 커다란 토픽을 추출해내는 시스템도 제안되었다. [6]는 수 많은 트윗들을 분석하여 사용자가 질의한 쿼리와 같이 사용된 키워드 그리고 관련된 트윗들을 분류해서 보여준다. [7]은 텍스트 랭크 알고리즘을 이용하여 트위터 사용자를 가장 잘 표현할 수 있는 태그를 추출하는 기법에 대하여 연구하였다.

3. 텍스트 랭크 알고리즘

Kleinberg[8]가 제안한 HITS 알고리즘이나 Sergey Brin과 Larry Page가 제안한 구글의 PageRank 알고리즘[9] 같은 그래프 기반 랭킹 알고리즘은 소셜 네트워크, 웹의 링크 구조, 논문의 인용 네트워크 등의 분석에 다방면으로 사용되어왔다. 그래프 기반의 랭킹 알고리즘은 네트워크 상에서 각 정점의 중요도를 결정하는 일련의 작업을 말한다. 그래프 기반의 랭킹 알고리즘으로 그래프 네트워크에서 가장 핵심적인 혹은 가장 중요한 정점들을 뽑아 낼 수 있다. 텍스트 랭크 알고리즘[10]은 그래프 기반 랭크 알고리즘을 자연어처리에 응용한 기법이다. 텍스트를 그래프로 모델링 한 뒤 그래프 기반 랭킹 알고리즘을 적용한 후, 핵심적인 정점을 추출해 내는 방법을 사용하였다. 본 논문에서는 키워드 추출이라는 자연어 처리 응용을 수행하기 위하여 텍스트 랭크 알고리즘을 키워드 추출에 맞도록 정의하였다.

3.1 텍스트의 그래프 모델링

텍스트 랭크 알고리즘을 이용하여 키워드를 추출하기 위해서는 분석할 텍스트를 그래프로 모델링해야 한다.

우선, 그래프를 구성할 정점을 결정한다. 키워드 추출 작업에 적합한 정점의 단위는 명사이다. 따라서 주어진 텍스트의 형태소를 분석하여 명사를 추출해낸 후 해당 명사들을 이용하여 그래프를 구성한다. 이 때 색인어로서 의미가 없는 불용어들을 제거하면 성능 향상에 도움이 된다. 이와 같은 과정에서 그래프에 포함될 정점들이 결정되면 정점들 사이의 관계를 이용해서 간선을 그리는 단계로 넘어간다. 가장 간단한 방법으로 동시 출현(co-occurrence) 관계를 이용해서 간선을 생성할 수 있다. 크기가 N인 윈도우 안에 두 토큰이 등장한다면 두 토큰 사이에 동시 출현 관계가 있다고 할 수 있다. 이 때, 동시 출현 빈도수가 간선의 가중치가 될 수 있다. 따라서 두 키워드 사이의 가중치는 등장하는 횟수가 되므로 자주 등장 두 키워드 사이의 관계는 가중치로 모델링 하여 의미를 나타내도록 하였다.

3.2 그래프 기반 랭킹 알고리즘 적용

주어진 텍스트로부터 생성된 그래프에 그래프 기반 랭킹 알고리즘을 적용하여 각 정점의 스코어를 얻어낸다. 이 때, 적용할 그래프 기반 랭크 알고리즘은 페이지 랭크 알고리즘을 이용한다.

$$\mathrm{PR}(\mathrm{V_i}) = (1-d) + d * \sum_{V_j \in In(\mathrm{V_i})} \frac{PR(V_j)}{|out(V_j)|} \quad (1)$$

V_i 는 그래프 상의 i 번째 정점을 의미하며, PR(V_i)는 페이지 랭크 알고리즘에 의해 얻어진 정점 V_i 의 스코어를 의미한다. $In(V_j)$ 는 정점 V_i 를 가리키는 이웃 정점의 집합을 의미하며, $Out(V_i)$ 는 정점 V_i 가 가리키는 이웃 정점의 집합을 의미한다. 마지막으로 d 는 damping factor 값으로 랜덤 서퍼 모델(random surfer model)을 반영하기 위한 값이다.

3.1절에 의해 생성된 그래프는 무방향성 가중치 그래프이다. 하지만 식 (1)에서 볼 수 있듯이 페이지 랭크 알고리즘은 가중치가 없는 방향성 그래프에 적용할 수 있도록 고안된 알고리즘이다. 따라서 페이지 랭크 알고리즘을 무방향성 가중치 그래프에 맞도록 변형시켜야 한다. 우선 무방향성 그래프의 경우 하나의 무방향성 간선을 두 개의 왕복하는 방향성 간선으로 생각하면 쉽게 변형이 가능하다. 가중치를 고려하기 위해 [11]에서는 페이지 랭크 알고리즘을 다음과 같이 가중치 그래프에 맞도록 변형시켰다.

$$PR^{\mathrm{w}}(\mathrm{V_i}) = (1-d) + d * \sum_{V_j \in In(\mathrm{V_i})} w_{ji} \frac{PR^{w}(V_j)}{\sum_{V_j \in In(\mathrm{V_i})} w_{ki}} \quad (2)$$

w_{jj} 는 정점 V_i 와 정점 V_j 사이에 존재하는 간선의 가중치를 의미한다. 실제 구현을 할 때, damping factor 인 d의 값은 0.85를 기본으로 사용한다. 이 식으로 그래프를 이루고 있는 모든 정점에 대하여 스코어를 반복해서 계산한다. 이 반복적인 계산은 모든 정점의 스코어의 변동폭이 특정 범위 이내로 수렴 할 때까지 계속된다. 그래프 기반의 랭크 알고리즘을 이용하여 얻어진 정점의 스코어를 기준으로 내림차순 정렬을 하여 상위 K개를 뽑게 되면 해당 정점이 텍스트를 가장 잘 설명할 수 있는 키워드가 된다.

3.3 텍스트 랭크 알고리즘의 장점

여러 자연어처리 응용 중에 텍스트 랭크 알고리즘만이 갖는 장점이 있다. 첫 번째로 텍스트 랭크 알고리즘은 별도의 학습데이터가 필요하지 않은 unsupervised learning 알고리즘이다. 많은 경우 학습 데이터를 얻을 수 없거나 얻어진 학습 데이터의 정확도가 떨어진다. 이런 경우 unsupervised learning 알고리즘인 텍스트 랭크 알고리즘이 유용하게 쓰인다. 특히 트위터와 같이 최신 정보 위주의 텍스트의 경우 더욱 unsupervised learning 알고리즘의 사용이 타당하다. 두 번째로 모듈화된 구성에 있다. 텍스트 랭크 알고리즘은 그래프로 모델링 하는 모듈, 그래프 기반 랭킹 알고리즘을 적용하는 모듈, 알고리즘에 의해 얻어진 스코어를 기반으로 적절한 정점을 추출해 내는 모듈로 이루어져 있다. 따라서 각 모듈을 수행하고자 하는 자연어 처리 작업의 특성에 맞게 수정하거나 대체 할 수 있다. 텍스트 랭크 알고리즘의 가장 큰 장점은 작업에 따라 적절하게 적용 할 수 있는 유연성에 있다.

4. 텍스트 랭크 알고리즘을 이용한 타임라인 요약

본 논문에서는 텍스트 랭크 알고리즘을 트위터의 타임라인을 구성하고 있는 트윗들의 키워드를 추출하는 응용에 사용한다. 트위터에서 유통되고 있는 트윗들의 주제는 매우 빠르게 변하고 있다. 어제의 주제와 오늘의 주제가 다르고 나날이 발행되는 트윗의 수는 늘어나고 있기 때문에 현재 논의되고 있는 주제를 파악하기가 점점 더 어려워지고 있다. 트위터에서도 이런 문제를 인지하여 실시간 트렌드라는 기능을 제공하고 있지만 실시간 트렌드는 개인의 관심사를 반영하지 못 하고 있다. 따라서 실시간 트렌드와 유사한 기능을 개인의 관심사가 반영된 타임라인에 적용을 한다면 트위터 전역에 대한 실시간 트렌드에 비해서 개인에게 유용한 정보를 줄 수 있다.

본 연구에서 제안하는 타임라인 요약 과정은 총 6단계를 거치게 된다. 사용자 개인의 타임라인을 수집하는 “Collect tweets” 단계를 거쳐, 정보성 트윗을 걸러내는 “Filtering” 단계, 걸러진 트윗을 이용하여 그래프로 모델링 하는 “Graph Modeling” 단계, 모델링 된 그래프에 페이지 랭크 알고리즘을 적용하여 스코어를 구하는 “Graph-based Rank Algorithm” 단계, 구해진 스코어를 기반으로 토픽 키워드와 하위 키워드를 추출하는

“Extract Topic Keyword” 단계, 마지막으로 구해낸 키워드들을 이용해서 키워드에 해당하는 트윗들을 분류하는 “Categorizing tweets” 단계를 거치게 된다.

4.1 트윗 수집 & 필터링 단계

본 연구의 분석 대상은 트위터 전체 데이터가 아니라 한 트위터 사용자 개인의 타임라인이다. 사용자의 타임라인을 이루고 있는 트윗들을 가져오기 위해 트위터에서 제공하는 open API를 이용한다. 트위터 상에 유통되고 있는 트윗들을 수집하여 분류해본 결과 29.5%의 트윗이 뉴스 정보를 담고 있었고, 34.9%의 트윗이 일상적인 대화를 담고 있는 것으로 나타났다[5]. 대화를 담고 있는 트윗의 경우, 개인의 일상을 담고 있는 경우가 많아 트렌드를 반영하지 못한다. 따라서 타임라인을 정리하는 본 논문의 분석대상에서 제외하였다. 비정보성 트윗을 걸러내기 위하여 트윗의 4가지 특성을 이용하였다. URL을 포함한 트윗의 경우 정보성 트윗을 가능성이 높다. 트윗된 트윗 역시 정보성 트윗일 가능성이 높다. 답글의 경우 비정보성 트윗일 가능성이 높다. “이벤트”, “경품” 같은 특정 키워드가 들어있는 트윗의 경우 트렌드보다 이벤트 적인 성격이 강해 비정보성 트윗일 가능성이 높다. 이 4가지 특성을 이용하여 분석에 포함될 트윗들을 필터링하여 정확도를 높였다.

4.2 그래프 모델링 단계

텍스트 랭크 알고리즘을 적용하기 위하여 필터링 된 정보성 트윗을 이용하여 타임라인 그래프를 구축한다.

4.2.1 정점의 선택

텍스트를 그래프로 모델링 하고자 할 때, 우선 정점으로 어떤 단위를 사용할 것인가를 정해야 한다. 본 논문에서는 타임라인을 이루고 있는 핵심 키워드를 뽑아 내는 작업을 하게 되므로 명사를 정점으로 취하게 된다. 텍스트에서 명사를 추출해내는 과정을 위해 형태소 분석기로 Lucene KoreanAnalyzer라는 Java 기반의 오픈 프로젝트 라이브러리를 사용하였다[12].

4.2.2 간선의 선택

텍스트에서 명사를 추출하여 단어 사이의 동시 출현 관계를 이용하여 간선을 만든다. 이 때, 함께 등장하는 빈도수를 가중치로 설정한다. 예를 들어, “스티브잡스”와 “애플”이라는 단어가 함께 등장한 횟수가 10이고, “스티브잡스”와 “겨울”이라는 단어가 한번 함께 등장했다고 하자. 가중치를 고려하지 않는다면 시스템의 전반적인 정확도가 떨어지게 된다. 따라서 보다 많은 횟수의 동시 출현 관계가 있다면 그 빈도수를 가중치로 두어, 두 단어 사이의 관련된 정도를 고려하도록 하였다.

4.2.3 전체 타임라인에 대한 그래프 모델링

타임라인을 정리하기 위해서는 각 트윗에 대한 그래프를 하나의 커다란 타임라인 그래프로 통합 할 필요가 있다. 타임라인 그래프를 만들 때 타임라인을 이루고 있는 각 트윗을 그래프로 모델링하고, 트윗들의 정점과 간선들의 합집합을 구하면 타임라인 전체에 대한 그래프를 얻을 수 있다. 간선의 가중치의 경우 합집합을 구하는 과정에서 모두 합한다. 예를 들어 트윗 1에서 정점 A와 정점 B사이 간선의 가중치가 3이고, 트윗 2에서 정점 A와 정점 B사이 간선의 가중치가 2라고 할 때, 타임라인 그래프로 통합하는 과정에서 정점 A와 정점 B사이 간선의 가중치는 5가 된다. 이렇게 하여 타임라인 전반에 걸쳐 빈번히 동시 출현 관계가 있는 단어 사이의 관련된 정도를 더 중요하게 모델링 할 수 있다.

4.3 그래프 기반 랭크 알고리즘의 적용

4.2 절에서 모델링 한 그래프는 가중치를 가지고 있는 무방향성 그래프이다. 따라서 이 그래프에 적용할 페이지 랭크 알고리즘은 식 (1)이 아닌 가중치 그래프로 적용할 수 있도록 변형된 식 (2)를 사용한다.

4.4 타임라인 키워드 추출

페이지 랭크 알고리즘을 적용하여 얻은 스코어를 기반으로 타임라인을 형성하고 있는 트윗의 키워드를 추출하게 된다.

4.4.1 스코어 기반 추출

우선 가장 단순한 방식으로 그래프 전체 정점의 집합을 스코어를 기준으로 내림차순 정렬하여 스코어가 높은 순으로 상위 K개의 키워드를 뽑아 내는 방법이 있다. 이 방법을 이용하게 되면, 텍스트를 이루고 있는 주요 키워드를 K개 뽑을 수 있다. 이 방법은 주로 텍스트가 단일 주제로 이루어져 있을 경우 높은 정확도를 갖게 된다. 트위터 사용자는 독자로서의 역할과 필자로서의 역할을 갖게 되는데[13], 필자로서의 트위터 사용자는 일관된 주제의 트윗을 발행하는 경향이 있다. 하지만 독자로서의 트위터 사용자는 다양한 주제를 구독하는 경향이 있다. 따라서 트위터 타임라인의 경우 일관된 주제를 다루고 있지 않아 스코어를 기준으로 상위 K개를 추출할 경우 전체 내용 중 주요 주제에 대한 키워드만 중복되어 추출된다. 그렇기 때문에 스코어를 기준으로 상위 K개를 추출하는 방법 이외의 다른 방법을 도입할 필요가 생긴다.

4.4.2 산봉우리 개념을 이용한 키워드 추출 방법

본 논문에서는 산봉우리 개념을 그래프에 적용하여 추출되는 키워드의 편향성을 제거하는 방법을 제안하였다. 지도를 들여다보면, 산의 지형을 논할 때, 산의 여러 지점 중에 주변보다 높은 지역을 ‘xx봉’이라고 하여 중요하게 생각한다. 이 봉우리를 기준으로 주변 지역을 말하게 된다. 이 개념을 스코어가 매겨진 그래프에 적용을 해보면 그래프에서도 주변보다 스코어가 높은 정점에 의미를 두어 토픽 레이블로 하고, 주변을 이루고 있는

정점이 토픽 레이블을 부가 설명하는 하위 키워드가 된다고 할 수 있다. 이 개념을 이용하면 타임라인에서 비슷한 주제를 나타내고 있는 키워드들은 모두 핵심 키워드의 하위 키워드로 모여 묶이게 된다. 따라서 상위 K개 안의 중복을 줄일 수 있고, 중요하지만 숨겨져 있던 비주류 주제들에 대한 키워드가 수면위로 떠오르게 된다. 표 1에 산봉우리 개념을 적용한 알고리즘을 간략하게 기술하였다.

표 1 산봉우리 개념을 이용한 핵심 키워드 추출

Input : Graph G and score of their vertices **Output** : List of keywords and their sub keywords
1: K = number of keywords to extract 2: L = List of vertices 3: P = List of keywords 4: for 1 < n < K 5: store the vertex that has highest core to T 6: add T to L 7: delete T from G 8: while L is not empty 9: Fetch one vertex from L 10: Add the vertex to T's sub keyword list 11: Delete the vertex from L 12: Fetch that vertex's neighbors 13: If(neighbors has lower score than the vertex) then 14: Add that neighbors to L 15: Delete that neighbors from G 16: End if 17: End while 18: End for

4.5 트윗 분류

타임라인을 이루고 있는 핵심 키워드 K개를 뽑은 뒤 마지막으로 각 키워드에 해당하는 트윗을 분류하는 작업을 거치게 된다. 트윗의 분류는 간단히 토픽 레이블 혹은 하위 키워드의 등장여부로 결정하게 된다. 예를 들어 토픽 레이블이 '애플'이고 하위 키워드가 '아이폰', '아이패드'일 때, 애플이라는 토픽에 해당하는 트윗으로 '애플', '아이폰' 혹은 '아이패드'가 들어 있는 트윗이 분류 된다. 이 과정을 위해 트윗들을 그래프로 모델링하는 과정에서 각 트윗에 트윗 번호를 부여하고 그 번호와 트윗이 담고 있는 명사를 기반으로 역색인(inverted index)을 구축하여 이용하였다.

5. 성능 평가

이 장에서는 산봉우리 개념을 응용한 타임라인 정리 기법의 성능을 측정하기 위한 실험의 환경 및 실험 방법, 실험 결과를 분석한다. 가장 단순하고 널리 이용되고 있는 키워드 추출 방법인 빈도수 기반의 방식과 본 연구에서 제안한 텍스트 랭크 기반에 산봉우리 개념을 더한 방식을 비교하여 성능을 평가한다.

5.1 실험 환경 및 데이터

실험에 사용할 데이터 수집을 위해 실험용 트위터 계정을 생성하였다. 표2에서 볼 수 있듯이 실험용 트위터 계정은 총 47개의 트위터를 팔로잉 하고 있으며 팔로잉 대상은 트위터 상에서 주요 정보 공급원 역할을 하고 있는 각 분야의 주요 언론, 정치인, 평론가, 유명 블로거 등으로 구성되어 있다. 이 47개의 계정들이 발행하는 트윗의 개수는 하루 평균 800개 이상이고, 다양한 주제를 다루고 있다.

실험에 사용된 데이터는 2011년 11월 25일 하루 동안 테스트 계정의 타임라인에서 수집된 764개의 트윗으로 진행을 하였다. 분석에 사용된 시스템은 Intel Core i7 2.93Hz의 CPU와 8GB의 메모리 사이즈를 갖는 Windows 7 Professional K 64bit 운영체제를 갖는다.

표 2 실험용 계정이 팔로잉하고 있는 트위터 목록

주요 언론	매일경제, 아이비타임즈, 컨슈머 타임스, Daily Asiaeconomy, Fudzilla, 전자신문, 스포츠 한국, 스포츠 조선 Baseball, 엑스포츠뉴스, SBS, 피디수첩, 미디어다음, MBC, 매일경제뉴스속보국, 프레시안, 오마이뉴스, The Korea Times, 조선일보, 연합뉴스, EBS, KBS, 파이낸셜 뉴스, 미디어 오늘, 블로터 닷넷, 시사인, 한겨레, 경향신문, 지디넷 코리아, 동아일보, 중앙일보, 중앙일보 편집국,
정치인	임종인, 김종철, 조승수, 정동영, 유시민, 김문수, 안희정, 심상정,
기타	광파리, 이해완 기자, IT수다떨기, 주진우기자, 이상호기자, 탁현민, 고재열 기자,

5.1.1 모델링 방법

사용자의 타임라인을 그래프로 모델링 하는 과정에서 unigram과 bigram으로 나누어 평가하였다. Unigram의 경우 하나의 단어가 정점이 되고, 동시 출현의 윈도우 크기는 2가 된다. Bigram의 경우 연속되는 두 개의 단어가 정점이 되고, 동시 출현 윈도우 사이즈는 3으로 하였다.

5.2 성능 평가 기준

타임라인의 요약이 얼마나 잘 되었는지 성능을 평가하기 위하여 정보 포함 범위(coverage), 토픽 간 유사도, 토픽 내 일관도의 총 3가지 평가 기준을 정하였다.

5.2.1 정보 포함 범위

최상위 K개의 토픽을 추출하였을 때, 그에 해당하는 트윗을 분류하게 된다. 이 때, 상위 K개의 토픽이 얼마나 많은 트윗을 분류하는지 커버리지를 평가하게 된다. 정확도가 같은 알고리즘이라 할 때, 커버리지가 더 높은 쪽이 더 많은 트윗을 분류할 수 있다는 뜻이다.

5.2.2 토픽 간 유사도

타임라인을 이루고 있는 K개의 토픽들을 뽑은 후 K

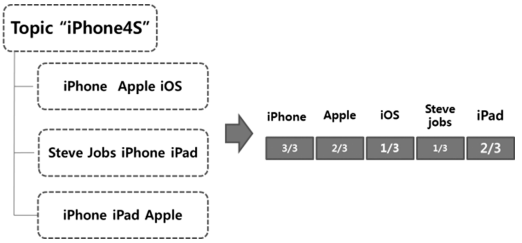


그림 1 각 주제별 토픽 벡터 생성

개의 토픽들이 얼마나 잘 분포되었는지 평가해야 한다. 이를 위해서 각 토픽에 분류된 트윗의 명사를 추출하여 그림 1에서 볼 수 있는 것처럼 토픽 벡터를 만든다. 이 때, 트윗을 이루고 있는 명사들이 토픽 벡터의 성분이 된다. 벡터를 이루고 있는 성분의 값은 출현 비율에 따라 결정된다. 이 방법을 이용하면 공통적으로 많이 등장하는 단어의 경우 성분 값이 높아 해당 토픽을 더 잘 설명한다고 할 수 있고, 우연히 쓰인 단어의 경우 성분 값이 작아지기 때문에 영향력이 핵심 단어에 비해 떨어지게 된다. 이렇게 상위 K개의 토픽에 대해 토픽 벡터를 생성한 다음, 각 토픽이 다루는 주제가 얼마나 유사한지 코사인 유사도를 이용하여 측정하게 된다. K번째 토픽의 유사도는 기존의 K-1개의 토픽 벡터와 K 번째 토픽 벡터 사이의 코사인 유사도를 모두 구해 가장 큰 값으로 한다.

5.2.3 토픽 내 일관도

잘 된 요약은 토픽간 분배 정도뿐만 아니라 분류된 트윗들이 일관된 주제를 다루고 있어야 한다. 트윗의 일관도를 측정하기 위하여 마찬가지로 토픽 벡터를 사용하였다. 각 토픽의 토픽 벡터를 구성한 다음 각 트윗의 명사를 벡터로 구성하여 토픽 벡터와의 코사인 유사도를 구한다. 각 토픽의 주제 내 일관도는 모든 트윗에 대해 이런 과정을 거친 후 얻어진 값을 평균 내어 측정하였다. 이 때, 트윗의 명사 벡터는 각 트윗을 이루고 있는 명사들을 성분으로 하고 값은 일괄적으로 1로 한다. 토픽 벡터의 경우 평균의 성격이 강하므로 이를 이용하여 각 트윗들이 얼마나 평균에 가까운 주제를 다루고 있는지 주제의 분산 정도를 알 수 있다.

5.3 실험 결과

앞의 절에서 설명한 기준을 이용하여 본 연구에서 제안한 텍스트 랭크를 이용한 방법과 빈도수 기반 방법의 성능을 평가, 분석하였다. 실험 결과의 graph는 그래프 기반 알고리즘, freq는 빈수기반 알고리즘을 의미한다. Uni는 unigram의 경우, bi는 bigram의 경우를 의미한다. 예를 들어 Graph-Uni의 결과는 graph 기반 알고리즘을 unigram으로 적용한 결과이다.

5.3.1 정보 포함 범위

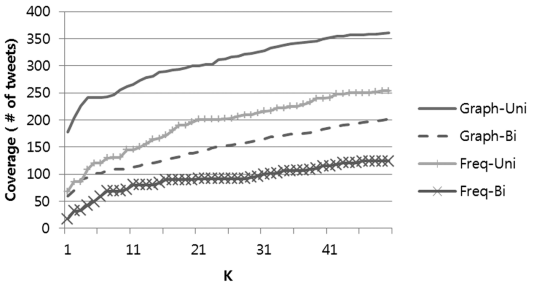


그림 2 분류 방법에 따른 커버리지

그림 2에서 볼 수 있듯이 커버리지의 경우 unigram의 경우와 bigram의 경우 모두 빈도수 기반의 방법보다 그래프 기반의 방법이 높았다. 따라서 같은 숫자의 트윗을 요약하는 데에 필요한 키워드의 개수가 빈도수 기반의 방법이 그래프 기반의 방법보다 더 많으며 정확도가 비슷하다고 할 때, 요약 능력이 그래프 기반의 방법이 더 뛰어나다고 할 수 있다. Bigram과 unigram의 경우 unigram을 이용한 방법이 커버리지가 더 높았다. 이를 해석하면, bigram으로 그래프를 모델링 한 경우 주제를 더 작은 여러 개의 주제로 나누게 되어 상위 K개의 키워드가 포함할 수 있는 트윗의 개수가 적어지게 되는 것이다.

5.3.2 토픽 간 유사도

토픽의 분배 정도를 나타내는 토픽 간 유사도의 경우 그래프 기반의 방식이 더 고르게 주제를 분류하는 것으로 나타났다. 상위 K개의 토픽들의 유사도 값을 5.2.2 절에서 정의한 방법에 따라 구한 뒤, 평균을 내어 K의 값이 늘어남에 따라 변화추이를 구해보았다. 그 결과 그림 3에서 볼 수 있듯이 빈도수 기반의 방식으로 키워드를 뽑았을 때, 중복되는 주제에 대한 토픽이 더 많이 뽑혀 유사도가 그래프 기반의 유사도보다 2배 이상 높게 나타났다. 따라서 그래프 기반의 방식이 좀 더 주제를 고르게 뽑아내는 것으로 볼 수 있다.

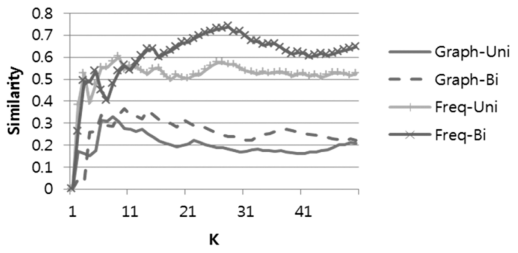


그림 3 분류 방법에 따른 토픽 간 유사도

5.3.3 토픽 내 일관도

그림 4를 보면 주제 내 일관도의 경우 대체로 빈도수

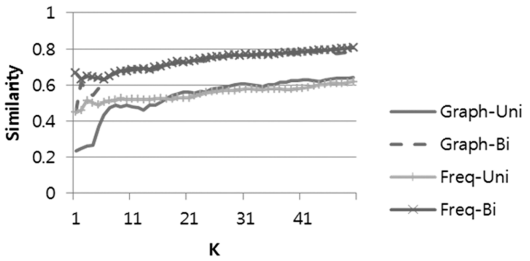


그림 4 분류 방법에 따른 토픽 내 일관도

기반의 방법이 그래프 기반의 방법보다 높은 일관도를 가진다는 것을 알 수 있다. 이는 주제 내 일관도의 정의에 따른 현상으로 빈도수 기반의 경우 무조건 하나 이상의 키워드가 모든 트윗에 포함되어 있어 기본적으로 일관도가 높게 나온다. 하지만 K가 늘어날수록 그래프 기반의 방법의 유사도 역시 증가하여 상위 20개의 토픽을 뽑았을 때, 일관도는 빈도수 기반의 방법과 그래프 기반의 방법에서 크게 차이가 나지 않았다. 특히 bigram의 경우 빈도수 기반의 방법과 그래프 기반의 방법 사이에 일관도 차이가 거의 없었다.

6. 논의사항

세 가지 기준으로 빈도수 기반의 방식과 본 연구에서 제안하는 방법을 평가한 결과 본 연구에서 제안한 방법이 더 높은 성능을 갖는 것으로 나타났다.

정보의 포함범위 측면에서 분석을 해보면, 빈도수 기반의 방식은 키워드 하나가 포함된 트윗들을 포함하지만, 본 연구에서 제안한 방식은 하나의 키워드와 그 키워드에 연결된 하위 키워드들을 포함한 트윗들을 골라온다. 따라서 하나의 토픽이 포함할 수 있는 양은 하나의 키워드로 트윗을 가져오는 빈도수 기반의 방식보다 여러 개의 키워드로 트윗을 가져오는 본 연구의 방식이 더 뛰어난 것으로 나타났다.

토픽간 유사도의 경우 얼마나 주제를 잘 분류했는지를 측정 할 수 있다. 트위터의 특성상 한가지 주제가 이슈화되면 비슷한 주제를 다루는 키워드들이 많이 등장하게 된다. 반면 본 연구에서 제안한 방식은 비슷한 주제를 다루고 있는 키워드를 하나의 덩어리로 합쳐서 다루기 때문에 분류된 주제들이 더 잘 분배되어 있을 수 있는 것이다.

토픽내 유사도는 빈도수 기반의 방식이 다소 높은 것으로 나타났다. 빈도수 기반의 방식은 하나의 키워드로 트윗을 가져온 반면 본 연구에서 제안한 방식은 하나의 토픽 키워드와 여러 하위 키워드를 이용해서 가져왔기 때문에 빈도수 기반의 방식이 토픽 내의 유사도가 높을 수 밖에 없다. 만약 토픽 내 유사도 측정 방식을 단순 키워드 벡터가 아닌 키워드의 시맨틱을 이용 할 수 있는 방식으로 하였다면 본 연구의 방식과 빈도수 기반 방식의 차이는 더 줄어 들 것이다.

7. 결론 및 향후 연구

본 연구에서는 트위터의 타임라인에서 문제가 되고 있는 정보 과다현상을 해결하기 위한 타임라인 요약 기법을 제안하였다. 텍스트 랭크 알고리즘을 사용하여 트위터의 타임라인을 정리하고 비슷한 키워드들을 하나로 묶어주는 산봉우리 개념이 적용된 알고리즘을 제안하였으며, 가장 널리 이용되는 트렌딩 토픽 추출 방법인 빈도수 기반의 방법과 성능을 비교 하였다. 또한 그래프로 모델링 하는 데 있어서 unigram과 bigram의 성능차이를 비교해보았다.

실험 결과에서 볼 수 있듯이 본 연구에서 제안한 텍스트 랭크 알고리즘과 산봉우리 개념을 적용한 타임라인 요약 기법은 가장 널리 사용되고 있는 빈도수 기반의 방법에 비해서 향상 된 성능을 보였다. Unigram과 bigram을 비교한 결과 unigram의 경우 커버리지는 높았지만 토픽 간 유사도와 토픽 내 일관도의 경우 bigram이 훨씬 더 좋은 성능을 보였다. 이는 bigram의 경우 주제를 좀 더 세분화하여 모델링하기 때문으로 보인다.

향후 연구로, 본 연구에서 제안한 모델에 의미론적인 요소가 가미된다면 더 정확한 요약을 할 수 있을 것이다. 추가적으로 온톨로지를 이용한다면 단어가 가지고 있는 의미를 보다 정확하게 사용 할 수 있을 것이다. 온톨로지 정보를 시스템에 통합하여 동의어를 효과적으로 처리한다면 좀 더 뛰어난 성능을 보일 수 있을 것이다. 또한, unigram과 bigram의 두 가지 그래프 모델링 방법을 효과적으로 함께 사용한다면 bigram의 장점인 높은 정확도와 unigram의 장점인 커버리지를 가질 수 있을 것이다.

참 고 문 헌

[1] http://www.bloter.net/archives/83805

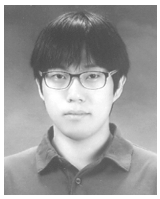
[2] H. Kwak, C. Lee, H. park, S. Moon, "What is Twitter, a Social Network or a News Media?" *Proc. of the 19th International World Wide Web Conference*, 2010.

[3] J. Sankaranarayanan, H. Samet, B. Teitlery, M. Lieberman, J. Sperling, "TwitterStand: News in Tweets," *Proc. of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, 2009.

[4] S. Abrol, L. Khan, "Twinner: Understanding News Queries with Geo-contents using Twitter," *Proc. of the 6th Workshop on Geographic Information Retrieval*, 2010.

[5] C. Castillo, M. Mendoza, B. Poblete, "nformation Credibility on Twitter," *Proc. of the 20th International World Wide Web Conference*, 2011.
[6] B. O'Connor, M. Krieger, D. Ahn, "TweetMotif: Exploratory Search and Topic Summarization for Twitter," *Proc. of the International AAAI Conference on Weblogs and Social Media*, 2010.
[7] W. Wu, B. Zhang, M. Ostendorf, "Automatic Generation of Personalized Annotation Tags for Twitter Users," *Human Language Technologies: The Annual Conference of the North American Chapter of the Association for Computational Linguistics*, 2010.
[8] J. Kleinberg, "Authoritative Sources in a Hyperlinked Environment," *Journal of the ACM*, 1999.
[9] S. Brin, L. Page, "The Anatomy of a Large-scale Hypertextual Web Search Engine," *Computer Networks and ISDN Systems*, 1998.
[10] R. Mihalcea and P. Tarau, "TextRank: Brigning Order into Texts," *Proc. of EMNLP and the Conference on Empirical Methods in Natural Language Processing*, 2004.
[11] R. Mihalcea, "Graph-based Ranking Algorithms for Setence Extraction Applied to Text Summarization," *Proc. of the ACL on Interactive poster and demonstration sessions*, 2004.
[12] cafe.naver.com/korlucene
[13] M. Welch, U. Schonfeld, D. He, J. Cho, "Topical Semantics of Twitter Links," *Proc. of the fourth ACM International Conference on Web Search and Data Mining*, 2011.

안 인 석
2010년 단국대학교 전자컴퓨터공학부(학사). 2012년 서울대학교 컴퓨터공학부(석사) 2012년~현재 티베로 재직 중. 관심분야는 데이터베이스, 소셜 네트워크, 시맨틱 웹, 스토리지 시스템



김 현 우
2007년 KAIST 전산학과(학사). 2007년~현재 서울대학교 컴퓨터공학부 석박통합과정 재학중. 관심분야는 추천, 태깅, 데이터베이스

김 형 주
1982년 서울대학교 전산학과(학사). 1985년 Univ. of Texas at Austin(석사). 1988년 Univ. of Texas at Austin(박사). 1988년~1990년 Georgia Institute of Technology(부교수). 1991년~현재 서울대학교 컴퓨터공학부 교수. 관심분야는 데이타베이스, XML, 시맨틱 웹, 온톨로지