Journal of the Korean Institute of Industrial Engineers
Vol. 40, No. 1, pp. 18-33, February 2014.
ISSN 1225-0988 | EISSN 2234-6457
http://dx.doi.org/10.7232/JKIIE.2014.40.1.018

<Original Research Paper>

KR-WordRank : WordRank를 개선한 비지도학습 기반 한국어 단어 추출 방법

김현중[1] · 조성준[1] · 강필성[2†]

[1]서울대학교 산업공학과 / [2]서울과학기술대학교 글로벌융합산업공학과

KR-WordRank : An Unsupervised Korean Word Extraction Method Based on WordRank

Hyun-joong Kim[1] · Sungzoon Cho[1] · Pilsung Kang[2]

[1]Dept. of Industrial Engineering, Seoul National University
[2]Dept. of Industrial and Information Systems Engineering, Seoul National University of Science and Technology

A Word is the smallest unit for text analysis, and the premise behind most text-mining algorithms is that the words in given documents can be perfectly recognized. However, the newly coined words, spelling and spacing errors, and domain adaptation problems make it difficult to recognize words correctly. To make matters worse, obtaining a sufficient amount of training data that can be used in any situation is not only unrealistic but also inefficient. Therefore, an automatical word extraction method which does not require a training process is desperately needed. WordRank, the most widely used unsupervised word extraction algorithm for Chinese and Japanese, shows a poor word extraction performance in Korean due to different language structures. In this paper, we first discuss why WordRank has a poor performance in Korean, and propose a customized WordRank algorithm for Korean, named KR-WordRank, by considering its linguistic characteristics and by improving the robustness to noise in text documents. Experiment results show that the performance of KR-WordRank is significantly better than that of the original WordRank in Korean. In addition, it is found that not only can our proposed algorithm extract proper words but also identify candidate keywords for an effective document summarization.

Keywords: Word Extraction, Keyword Extraction, Text Mining, Unsupervised Learning, WordRank

1. 서 론

최근 화두가 되고 있는 빅데이터 환경에서는 숫자 및 기호로 이루어진 테이블 형태의 정형 데이터(structured data)보다 인터넷, 소셜 네트워크 서비스(social network service, SNS) 등의 사용을 통해 생성되는 텍스트 데이터와 같은 비정형 데이터(unstructured data)의 가치가 훨씬 더 클 것으로 전망되고 있다(McKinsey Global Institute, 2011). 따라서 이러한 공공 및 비즈니스 목적에 부합하는 새로운 가치를 창출하기 위한 텍스트 데이터 가공 및 분석 방법론의 개발을 주로 다루는 텍스트마이닝(text mining)에 대한 중요성이 크게 부각되고 있는 실정이다. 텍스트마이닝의 주요 활용 분야로는 키워드 기반의 연관관계 분석(keyword-based association analysis), 자동 문서 분류(automatic document classification), 문서간 유사도 탐색(similarity detection between documents), 특이 문서 탐지(anomaly detection), 문서 간 링크 분석(link analysis) 등이 있다(Hotho *et al.*,

본 연구는 서울과학기술대학교 교내연구비의 지원을 받아 수행되었음.
† 연락저자 : 강필성 교수, 139-743 서울시 노원구 공릉길 138 서울과학기술대학교 글로벌융합산업공학과, Tel : 02-970-7286, Fax : 02-974-5388, E-mail : pskang@seoultech.ac.kr
2013년 11월 18일 접수; 2013년 12월 30일 수정본 접수; 2014년 1월 9일 게재 확정.

2005; Berry and Castellanos, 2007; Cho and Kim, 2012).

앞서 언급한 모든 텍스트마이닝 기법들의 기저에는 특정 언어에서 사용되는 문법적인 단어의 인식뿐만 아니라 단어의 형태 및 의미 분석이 높은 정확도로 구현할 수 있다는 가정이 내포되어 있다. 즉, 언어의 가장 기본적인 구성단위인 단어에 대해 사람들이 문맥(context)상에서 자연스럽게 인식하는 것과 같이 컴퓨터 또한 해당 단어에 대한 자동 인식이 가능한 어휘사전이 구축되어 있음을 전제하는 것이다. 이를 기초로 연속된 단어들 간의 관계를 파악하기 위한 연어(collocation) 이해, 단어들의 문법적 종류를 분류하기 위한 형태소 분석, 각 단어 간의 문법적 관계를 파악하기 위한 의존성 분석(dependency parsing) 등을 이용한 다양한 텍스트마이닝 알고리즘들이 실행되는 것이다. 대부분의 데이터마이닝 관련 연구 동향에서 알 수 있듯이, 텍스트마이닝을 위한 기초 연구 역시 영어권 국가, 특히 미국의 연구자들에 의해 주도적으로 진행되고 있는 실정이다. 영어의 경우에는 단어에 대한 형태학적 분석 및 의미적 관계를 파악하여 데이터베이스를 구축하는 WordNet 프로젝트를 통해 상당히 높은 완성도를 갖는 어휘사전(lexical database)의 구축이 진행 중에 있으며(Fellbaum, 2005), 형태소 분석 또한 Snowball 프로젝트의 Porter2 Stemmer등을 통해 높은 수준의 처리 능력을 확보한 상황이다(Willett, 2006). 반면 한글과 관련된 연구에서는 세종 말뭉치(http://www.sejong.or.kr/)로 대표되는 어휘사전과 꼬꼬마 형태소 분석기로 대표되는 형태소 분석기(이동주 외, 2010)가 개발되어 있으나 영어에 비해서는 그 완성도가 높지 않은 편이다.

한글에 대한 어휘사전 및 형태소 분석기의 완성도를 높이기 위해 선결되어야 하는 가장 중요한 이슈는 현실 세계에서는 단어를 인식하는 것 자체가 어려운 경우가 상당히 빈번하게 존재한다는 것이다. 첫째로 사용되는 모든 단어를 예상할 수 없는 경우가 있다. 언어는 시간이 지남에 따라 끊임없이 진화하는 생물체와 같기 때문에 신조어가 만들어지기도 하며, 전문적인 분야의 문서에서는 각 도메인에서만 사용되는 단어들이 존재하기도 한다. 이와 같이 일반적으로 사용되지 않는 단어들을 인식하기 위해서는 단어에 관련된 사전 지식이 필요하며, 사전 지식으로 정의되지 않는 단어를 미등록 단어(out-of-vocabulary)라 한다(Jurafsky and Martin, 2009). 둘째로 데이터에 많은 오류가 있는 경우가 있다. 한국어를 비롯하여 유럽권 언어나 중동 지역의 언어들은 단어를 인식하기 위하여 띄어쓰기를 단어의 경계로 이용한다. 하지만 소셜 네트워크와 같은 비전문가들에 의하여 작성된 많은 온라인 문서에서 빈번하게 발생하는 철자법 오류, 언어 파괴 현상, 그리고 띄어쓰기 오류 등은 단어의 인식을 매우 어렵게 만들고 있다. 단어 인식이 올바르게 이루어지지 않을 경우 텍스트 데이터의 전처리 과정을 올바르게 수행할 수 없기 때문에 뒤이어 이루어지는 문서 요약, 주제 탐지 및 추적과 같은 응용 연구에 사용되는 알고리즘의 정확도가 현저히 저하되는 문제점이 발생한다.

단어 인식 방법은 학습데이터를 이용하여 단어를 추정할 수 있는 정보를 학습하는 지도학습 기반 방법과 사전 지식 없이 통계적인 정보를 기반으로 단어를 추정하는 비지도학습 기반 방법으로 나눌 수 있다(Jin and Tanaka-Ishii, 2006; Zhao and Kit, 2007). 지도학습 기반 방법은 학습데이터에 빈번하게 등장하는 단어에 대한 인식 정확도는 상당히 높은 반면, 그렇지 않은 않은 단어나 규칙 등의 인식 능력은 상대적으로 취약하다. 또한 띄어쓰기를 기반으로 단어를 인식하는 경우에는 띄어쓰기 오류 발생 시 해당 단어가 학습데이터에 등장한 단어라 할지라도 다른 단어로 오판을 하거나 미등록 단어로 분류할 위험성이 크다. 그렇기 때문에 지도학습 기반 방법은 언어 오류가 많고 신조어와 같이 예상할 수 없는 단어가 끊임없이 생성되는 분야의 문서를 분석하기에 적합하지 않다. 지도학습 기반 단어 인식의 또 다른 문제는 모든 단어가 학습데이터에 등장하였더라도 문맥적 모호성과 도메인 적합성 문제를 해결하여야 한다는 것이다. 한 예로 띄어쓰기가 되지 않은 '서울대강당'이라는 단어는 '서울 대강당' 혹은 '서울대 강당'을 의미할 수 있다. 만약 학습데이터에 '서울 대강당'이 '서울대 강당'보다 많이 등장하였다면 학습된 모델은 전자로 의미를 해석할 것이다. 하지만 분석하고자 하는 문서가 '서울시'와 관련된 문서 집합이 아니라 '서울대학교'와 관련된 것이라면 '서울대 강당'의 의미로 해석하는 것이 적절하다. 이러한 예시는 띄어쓰기 오류에 의하여 모호성이 발생하였고, 학습데이터와 주어진 문서 집합의 도메인이 다름에 따라서 도메인 적합성 문제가 발생할 수 있음을 보여주는 예이다.

이러한 문제를 해결하기 위하여 학습데이터와 같은 사전지식에 의존하지 않으며 주어진 문서집합에서 통계적 정보를 이용하여 단어를 인식하기 위한 비지도학습 기반 단어 인식 방법이 제안되었다(Sun *et al.*, 1998; Feng *et al.*, 2004; Jin, 2006). 이 방법은 지도학습 기반 방법과 비교하여 다음과 같은 장점이 있다. 첫째, 문서 집합으로부터 실제로 사용되는 단어들을 추출하기 때문에 전문 용어나 신조어와 같이 예상할 수 없는 단어를 인식하는데 효과적이며, 고정된 학습데이터를 구축하지 않기 때문에 시간과 비용이 절약될 수 있다. 둘째, 문서집합이 포함하는 전역적 정보를 이용할 수 있다(Zhao and Kit, 2007). 전역적 정보란 각 문서를 독립적으로 살펴볼 때는 알지 못하지만, 문서집합 전체를 살펴봄으로써 얻을 수 있는 정보를 뜻한다. 전역적 정보의 한 예로 서울대학교와 관련된 문서라면 '서울대'라는 단어가 '서울'보다 자주 등장할 것이다. 이 경우 '서울대강당'은 '서울대'와 '강당'의 합성어일 가능성이 높다. 비지도학습 단어 인식 방법은 학습데이터를 사용하지는 않지만 단어의 가설(word hypothesis)을 반영하는 통계적 기준을 사전에 정의한 뒤, 이를 바탕으로 단어를 인식한다. 단어를 정의하는 가설에 따라 다양한 비지도학습 단어 인식 방법이 제안되었는데, Sun *et al.*(1998)은 응집성(cohesion)이 높은 연속된 글자를 단어로 인식하기 위하여 상호 정보량(mutual information)을 이용하는 방법을 제안하였다. 글자간의 응집성이 단어를 구

성하는 글자간의 정보로부터 통계적 정보를 추출하는 내부 경계 값(interior boundary value)이라면, 단어 주변의 다른 글자들로부터 통계적 정보를 추출하는 외부 경계 값(exterior boundary value)을 계산하는 방법도 제안되었다(Feng *et al*., 2004; Jin and Tanaka-Ishii, 2006). 특히 Harris(1955)의 "The uncertainty of tokens coming after a sequence helps determine whether a given position is at a boundary" 주장을 통계적으로 표현하기 위하여 좌우에 등장하는 글자의 종류가 많은 경우 단어로 인식하는 Accessor Variety(Feng *et al*., 2004)와 좌우에 등장하는 글자의 엔트로피를 계산한 뒤 그 값이 큰 경우 단어로 인식하는 Branch Entropy(Jin and Tanaka-Ishii, 2006)가 제안되었다. Accessor Variety는 이산적 방법이고 Branch Entropy 방법은 Accessor Variety의 연속형 방법이다. 또한 베이지안 모델을 이용하여 비지도 학습 방법으로 단어를 인식하는 방법론이 제안되었다(Mochihashi *et al*., 2009). 그 외에도 자연어분석에서 많이 사용되는 그래프 구조를 이용한 단어 추출 방법도 제안되었는데, Chen *et al*.(2011)의 연구에서는 단어 후보를 그래프의 마디로 구성한 뒤 Kleinberg(1999)에서 제안된 mutual reinforcing relationship 방법을 이용하여 네트워크상에서 중심성이 높은 후보 마디를 단어로 추출하는 WordRank 방법이 제안되었다. 이 방법은 베이지안 모델과 같이 복잡한 계산을 요구하지 않으며 비슷한 성능으로 사전지식 없이 단어를 인식할 수 있다.

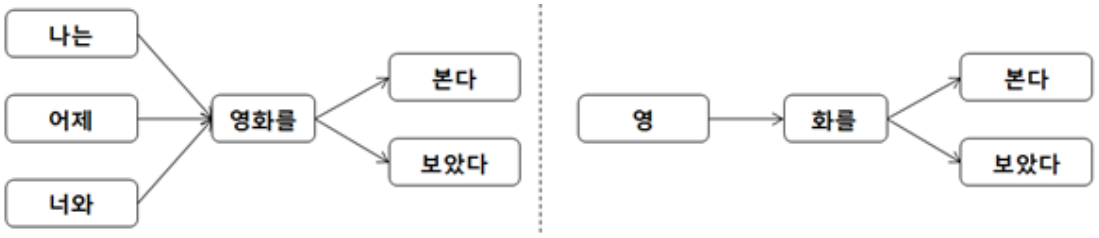
WordRank 방법은 중국어와 일본어의 단어 인식에 대해 복잡한 계산을 하지 않으면서도 상당히 높은 정확도를 나타내는 것으로 알려져 있다(Chen *et al*., 2011). 그러나 한글에 WorkRank 방법을 적용할 경우, 다의적인 1음절 글자의 존재로 인해 단어 인식의 성능이 저하되는 문제점이 발생한다. 본 연구에서는 한글 단어 인식에서 WorkRank가 갖는 문제점을 해결하기 위해 부분 글자의 위치정보를 고려하는 KR-WordRank 방법을 제안하고 그 효과를 검증하고자 한다. 본 연구에서 제안된 KR-WordRank는 비지도학습 기반의 단어 인식 기법으로서 학습데이터를 사용하지 않기 때문에 사전 지식에 의존적이지 않으며 다양한 데이터 원천으로부터 수집되어 띄어쓰기와 같은 오류가 많은 문서 집합에서도 효과적으로 단어를 인식할 수 있을 것으로 기대한다. 또한 제시된 방법론이 철자법과 띄어쓰기 오류가 빈번하게 발생하는 실제 온라인 문서 집합으로부터 단어를 올바르게 추출할 수 있는지를 검증하기 위하여 실제 국내 개봉 영화의 감상평 데이터를 사용하여 기존 방법론인 WorkRank와의 단어 인식 성능을 비교하였다.

본 논문은 다음과 같이 구성되어 있다. 제 2장에서는 본 연구에서 제안된 방법론의 기본이 된 비지도학습 단어 추출 방법인 WordRank에 대하여 알아본 뒤, 한국어에 적용할 경우 발생하는 문제점과 원인에 대하여 살펴본다. 제 3장에서는 WordRank의 문제점을 극복하여 한국어 단어 추출 성능을 높인 KR-WordRank 방법을 제안하고 두 방법의 차이점을 비교한다. 제 4장에 노이즈가 많은 실제 온라인 문서 데이터와 세종 말뭉치를 이용하여 두 알고리즘의 한글 단어 추출 성능을 비교한다. 또한, KR-WordRank가 단어와 키워드를 동시에 추출할 수 있음을 보이고 KR-WordRank의 한계점 및 추후 발전 방향을 논의한다. 제 5장에서는 본 연구의 결론과 함께 향후 연구 방향을 논의한다.

2. WordRank

WordRank는 Chen *et al*.(2011)의 연구에서 제안된 외부 경계 값(exterior boundary value; EBV)과 내부 경계 값(interior boundary value; IBV)을 모두 이용하여 단어를 인식하는 비지도학습 방법이다. 외부 경계 값이란 주어진 단어의 좌우 주변에 다른 단어가 나타날 가능성을 의미하며, 내부 경계 값이란 주어진 단어를 이루는 연속적인 글자의 응집성을 의미한다. WordRank의 외부 경계 값은 링크로 연결되어 있는 웹 공간에서 웹 문서의 중요도를 계산하기 위하여 제안된 mutual reinforcing relationship(Kleinberg, 1999)을 이용하여 계산한다. 이 방법은 각 웹 페이지의 중요도를 authority라 명한 뒤, authority가 높은 웹 페이지로부터 유입되는 링크가 많은 웹 페이지의 authority가 높다고 가정한다. 이 방법은 PageRank(Lawrence *et al*., 1999)처럼 각 마디의 authority를 각 마디로 유입되는 링크를 가지는 다른 마디의 authority의 합으로 정의한다. WordRank에서는 올바른 단어의 좌우 경계에는 다른 올바른 단어가 존재할 것이라고 가정한 뒤, 각 단어의 외부 경계 값인 authority를 주위의 올바른 단어들의 authority에 의하여 강화하는 방식이다. 즉 WordRank에서는 외부 경계 값이 높은 단어 후보들이 인접한 다른 후보들의 외부 경계 값을 높임으로서 서로 외부 경계 값을 상호 강화한다. 단어 후보 마디는 각각 두 종류의 외부 경계 값을 갖는데, 왼쪽 경계의 authority는 left boundary value(LBV)로 정의되고 오른쪽의 경계의 authority는 right boundary value (RBV)로 정의되며 이 두 경계 값을 바탕으로 authority 값이 높은 단어 후보를 단어로 추출한다.

문서 집합이 클 경우 <Figure 1>(a)와 같이 '영화를' 왼쪽에는 '나는' 외에도 '어제', '너와' 등 높은 외부 경계 값을 가지는 다양한 단어들이 나타나지만, 단어가 아닌 '화를'의 경우 <Figure 1>(b)와 같이 올바르지 않은 몇 글자만이 왼쪽에 인접하여 존재한다. 특히 '나는 영화를 본다'라는 문장에서 '영화를'의 왼쪽에는 '나는'이라는 올바른 단어 경계가 존재하기 때문에 LBV와 RBV이 모두 크다. '화를'의 경우 오른쪽에 다른 단어가 등장할 가능성은 '영화를'과 같기 때문에 동일한 RBV을 가지지만, 왼쪽에는 '영'이라는 잘못된 단어 경계가 존재하기 때문에 작은 LBV를 가진다. <Figure 1>(a)에서 '영화를'은 왼쪽 경계를 '나는', '어제', '너와'와 공유하고, 오른쪽 경계를 '본다', '보았다'와 공유한다. 즉 WordRank에서 단어 후보 그래프는 식 (1)에서 나타난 바와 같이 각 마디(단어 후보)는 좌/우 방향 외부 경계 값을 가지고, 각 마디 간의 관계는 왼쪽 이웃 관계를 나타



(a) (b)

Figure 1. An example of correctly classified words (a) and incorrectly classified words (b)

내는 호(E_{LN})와 오른쪽 이웃 관계를 나타내는 호(E_{RN})로 구성되어 있다.

$$G(V, E_{LN}, E_{RN}) \quad (1)$$

WordRank는 단어 후보 그래프를 구성하기 위하여 <Figure 2>(a)와 같이 가능한 모든 단어 후보를 생성하고, 각 단어 후보의 위치에 따라서 왼쪽 이웃 호(E_{LN})와 오른쪽 이웃 호(E_{RN})로 각 단어 후보를 연결한다. <Figure 2>(b)에서 '나는'은 '영화'의 왼쪽에 등장하였기 때문에 '영화' → '나는' 방향으로 E_{LN}을 결하고, '나는' → '영화' 방향으로 E_{RN}을 연결한다. 각 단어 후보의 RBV는 오른쪽에 등장한 단어 후보의 LBV의 합으로, LBV는 왼쪽에 등장한 단어 후보의 RBV의 합으로 정의된다. 이는 두 단어 후보가 같은 단어 경계에 대하여 각각 다른 종류의 단어 경계 값을 공유하기 때문이다. 예를 들어 <Figure 1>(a)에서 '영화를'의 오른쪽에 등장하는 '본다'와 '보았다'는 '영화를'의 오른쪽에 등장하기 때문에 '영화를'의 RBV를 설명하는데 이용된다. 반대로 '영화를'은 '본다'와 '보았다'의 왼쪽에 등장하여 두 단어의 LBV를 설명하는데 이용된다. 이 때 '영화를'의 오른쪽에 높은 LBV의 값을 갖는 단어 후보들이 많이 등장할수록 '영화를'의 RBV의 값은 강화되고, 그 반대로 '본다'의 왼쪽에 높은 RBV를 지닌 단어 후보들이 많이 등장할수록 '본다'의 LBV가 강화된다. 각 단어 후보는 다른 단어 후보의 외부 경계 값을 상호 강화시키기 위하여 식 (2) 및 식 (3)과 같은 식을 사용한다.

$$LBV(w)^{i+1} = \sum_{(l,w)\in E_{LN}} RBV(l)^{i} \quad (2)$$

$$RBV(w)^{i+1} = \sum_{(l,w)\in E_{RN}} LBV(l)^{i+1} \quad (3)$$

where $LBV(w)^0 = RBV(w)^0 = \frac{1}{|V|}$

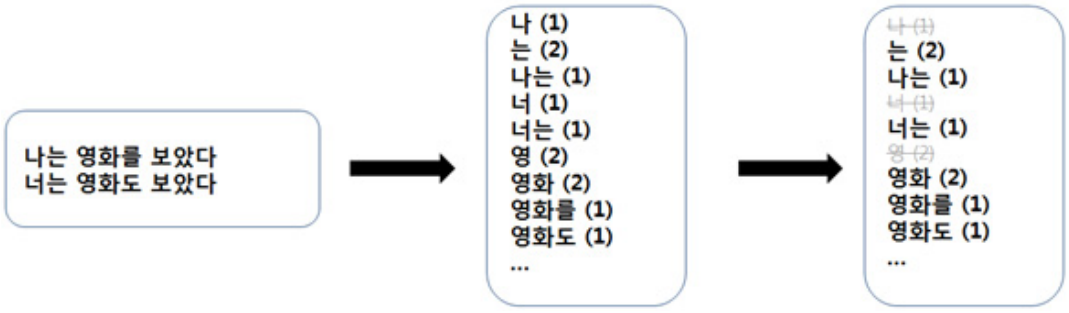
and $|V|$: number of nodes

i는 mutual reinforcing의 반복 계산 횟수를 의미하는 인덱스로 사전에 정의된 반복 횟수 k까지 식 (2)와 식 (3)을 차례로 반복한다. 각 단어 후보의 LBV는 좌측에 인접한 다른 단어 후보들의 RBV의 합이기 때문에 E_{LN}으로 연결되어 있는 단어 l의 RBV 합을 w의 LBV 값으로 업데이트한다. 그리고 업데이트된 LBV의 합을 E_{RN}에 속한 w의 RBV로 업데이트한다. 상호강화 관계 방법은 PageRank와 다르게 각 반복 단계 마다 외부 경계 값의 합이 증가하기 때문에 업데이트 후 외부 경계 값의 합이 일정하도록 식 (2)와 식 (3)의 계산을 한 번 수행한 뒤 정규화 과정을 거친다.

WordRank에서는 단어 후보를 선별하는 과정에서 다음과 같은 경험적 방법이 도움이 될 수 있다. 첫째, 문서 집합에서 단어 후보의 출현 빈도수가 사전에 정의한 기준보다 작을 경우 후보에서 제외할 수 있다. 철자나 문법 오류에 의한 글자들의 출현 빈도는 매우 낮기 때문에 이를 통하여 단어 후보 그래프를 구성하기 전 다수의 불필요한 후보를 제외할 수 있다. 둘째, Substring Reduction(Lü *et al.*, 2004) 방법은 동일한 정보를 가지고 있는 단어 후보를 그래프에서 제외시킬 수 있다. 한 예로

(a) (b)

Figure 2. A process of candidate words generation (a) and the relation between words (b)



(a) 문서 집합 (b) 단어 후보와 빈도수 (c) Substring reduction 후 단어 후보

Figure 3. A process of eliminating unnecessary words using Substring Reduction

<Figure 3>에서 '영'과 '영화'의 빈도수는 모두 2이고, '영' 오른쪽에는 반드시 '화'가 등장하기 때문에 '영'은 '영화'를 의미한다. 즉 '영'과 '영화'는 동일한 정보를 지닌 중복된 단어 후보이다. 이 경우 길이가 가장 긴 글자인 '영화'만을 단어의 후보로 선택한다. 셋째, 언어적 사전 지식을 이용하여 단어 후보를 선별할 수 있다. 영어의 경우 단어가 되기 위해서는 하나 이상의 모음이 필요하기 때문에, 자음으로만 이루어진 글자는 단어 후보에서 제외할 수 있고, 다른 언어의 경우 적절한 언어 규칙을 적용할 수 있다.

WordRank는 단어의 내부 경계 값을 계산하기 위하여 mutual information(MI)을 사용한다. MI는 두 글자가 연속적으로 등장할 가능성을 수치화한 방법으로 식 (4)와 같이 기술된다.

$$MI(a,b) = \frac{p(a)p(b)}{p(ab)} \tag{4}$$

두 개 이상의 MI를 계산하기 위하여 다양한 확장 방법이 제안되었다(Petrović, 2010). WordRank에서는 단어 후보의 길이가 L일 경우 L-1개의 연속된 글자 조합의MI를 계산한 뒤, 최소값을 단어를 구성하는 글자의 응집성으로 판단하였다. 예를 들어 'abc'의 경우 $\min(MI(`a', `b'), MI(`b', `c'))$의 값을 내부 경계 값으로 사용하였다. 이와 같이 MI를 확장할 경우 '서울대'가 자주 등장한 문서 집합에서는 MI('서', '울')과 MI('울', '대')가 모두 크기 때문에 '서울대'의 내부 경계 값이 크다. 하지만 '서울대는'과 같이 자주 나타나는 단어가 아닌 경우에는 MI('대', '는')의 값이 작기 때문에 '서울대는'은 '서울대' 보다 상대적으로 작은 내부 경계 값을 갖는다.

단어는 다른 단어와 경계를 잘 나눌 수 있고 글자간의 응집성이 높아야 하므로, WordRank로부터 계산된 단어 가능성 점수 $WR(w)$는 식 (5)와 같이 외부 경계 값과 내부 경계 값의 곱으로 표현된다.

$$\begin{aligned} WR(w) &= EBV(w) \times f(IBV(w)) \\ &= LBV(w) \times RBV(w) \times f(IBV(w)) \end{aligned} \tag{5}$$

이 때 함수 $f(x)$는 내부 경계 값과 외부 경계 값의 가중치를 조절하기 위한 함수로 사용되었다. 일반적으로 $f_{poly}(x) = x^a$, $f_{\exp}(x) = b^x$를 사용할 수 있다. 내부 경계 값과 외부 경계 값의 중요도를 조절하는 변수인 a와 b는 값이 커질수록 $f(x)$의 값이 커지기 때문에 단어 가능성을 계산할 때 내부 경계 값의 비중을 키우는 효과가 있다.

3. KR-WordRank

3.1 WordRank의 한국어 적용 및 한계 : 다의적인 1음절 글자로 인한 단어 추출 성능 저하

알고리즘의 설명에 앞서 다음과 같이 용어를 정의한다. '단어 후보'와 '글자'는 아직 단어인지 확인할 수 없는 연속된 글자를 의미하고, 단어는 실제로 단어로 확인된 글자를 의미한다. 단어 추출 방법은 단어 후보 중에서 단어를 고르는 문제이고, 단어 가능성은 각 단어 후보 중에서 단어일 가능성(값, 순위)이다.

WordRank의 한글 적용 가능성을 알아보기 위하여 본 연구에서는 우선 WordRank가 제안된 형태 그대로 영화 '아저씨'의 평점과 함께 기록된 40자평 문장으로 이루어진 한글 데이터에 적용하여 단어 추출 성능을 평가하였다. 단어 가능성이 높은 글자 30개를 추출한 결과는 <Table 1>(a)와 같다. 여기서 * 표시는 올바르게 추출된 단어를 뜻하며, 문제 종류 A는 해석이 불가능 하거나 의미가 다의적인 경우이고, 문제 종류 B는 조사나 어미가 결합된 단어이다. 또한, 문제 종류 C는 어미나 조사가 단어로 추출된 경우를 의미하며, 문제 종류 D는 복합명사와 같이 유의미한 두 개 이상의 단어가 결합된 경우이다. WordRank의 적용 결과, 정상적으로 추출된 (* 표시) 단어 외에도 의미가 없거나 해석이 어려운 1음절 글자들의 단어 가능성이 높게 계산되는 것을 확인할 수 있다. 이러한 결과가 나타난 이유는 사용된 데이터에서 길이가 1음절인 글자의 좌우에 다양한 글자들이 나타났기 때문이다. WordRank는 단어의 주변에는 다양한 단어가 등장할 것이라고 가정한 뒤, 단어 후보 그래프에서 마디의 중요도인 authority가 높은 허브(hubs) 마디를 단어로 추출한다. 하지만 한국어의 경우 1음절 글자가 여러 상황에

Table 1. Top 30 word candidates identified by WordRank from 24,302 sentences of 40 characters-long comments on a certain movie collected from movie.naver.com

순위	(a) WordRank를 적용한 결과 (길이가 1음절 이상인 경우)		(b) WordRank를 적용한 결과 (길이가 2음절 이상인 경우)		(c) KR-WordRank를 적용한 결과 (길이가 1음절 이상인 경우)	
	단어(빈도수)	문제 종류	단어(빈도수)	문제 종류	단어(빈도수)	문제 종류
1	이(14448)	A	영화(10559)*		영화(10559)*	
2	영화(10559)*		원빈(9248)*		원빈(9243)*	
3	다(14351)	A	액션(3767)*		정말(2783)*	
4	원빈(9248)*		정말(2783)*		액션(3766)*	
5	액션(3767)*		최고(4255)*		최고(4251)*	
6	정말(2783)*		진짜(2105)*		진짜(2105)*	
7	도(5684)	A	대박(2009)*		대박(1988)*	
8	한(7061)	A	연기(2460)*		너무(1859)*	
9	만(6004)	A	너무(1860)*		연기(2460)*	
10	가(5036)	A	아저씨(1281)*		아저씨(1280)*	
11	아(5769)	A	감동(1109)*		완전(1073)*	
12	최고(4255)*		스토리(1022)*		감동(1109)*	
13	고(11221)	A	완전(1074)*		스토리(1022)*	
14	는(8919)	A	원빈의(1390)	B	보고(1875)*	
15	에(5386)	A	한국영화(1041)	D	한국영화(944)	D
16	지(6908)	A	원빈이(1078)	B	그냥(789)*	
17	진짜(2105)*		보고(1875)*		평점(1245)*	
18	나(4069)*		하지만(718)*		테이큰(753)*	
19	말(4501)	A	지만(1995)	C	본(1441)*	
20	기(4988)	A	그냥(789)*		굿(578)*	
21	화(11114)	A	네요(1670)	C	좀(557)*	
22	의(4621)	A	하고(960)	C	한국(1867)*	
23	대박(2009)*		평점(1245)*		이런(573)*	
24	연기(2460)*		테이큰(753)*		또(966)*	
25	어(4585)	A	한국(1868)*		배우(685)*	
26	요(6073)	A	이영화(504)	D	내용(566)*	
27	로(3066)	A	최고의영화(519)	D	처음(676)*	
28	너무(1860)*		에서(753)	C	연기력(437)*	
29	대(4137)	A	배우(685)*		이렇게(345)*	
30	아저씨(1281)*		내용(566)*		잔인(1608)*	

서 사용되기 때문에 단어 후보 그래프에서 허브로 등장한다. 또한 1음절 글자의 주변에는 단어뿐 아니라, 단어가 아닌 글자 역시 많이 등장한다. 즉 단어가 될 수 없는 불필요한 마디들 간의 연결 때문에 WordRank의 단어 추출 성능이 저하되는 결과가 나타난다. 이로부터 WordRank를 한국어의 단어 추출 문제에 적용하기 위해서는 불필요한 단어 후보 마디를 효과적으로 줄일 필요가 있음을 알 수 있다.

WordRank의 한국어 적용 가능성에 관련된 직관을 얻기 위하여 위와 동일한 데이터로부터 단어 길이가 2음절 이상인 상위 30개의 단어 후보를 선별한 결과는 <Table 1>(b)와 같다. 한국어에서는 길이가 2음절 이상이면서 의미가 없는 경우는 적기 때문에 <Table 1>(b)의 결과는 어떤 의미를 지니는 글자들이 단어로 추출된 것으로 보인다. 어떠한 문제도 없는 단어의 경우에는 도메인 지식으로 살펴볼 때, 영화 '아저씨'를 기술하기 위한 중요한 단어들로 해석된다. 단어 후보 그래프의 허브 마디들은 단어 가능성이 높을 뿐 아니라 문서 집합에서 자주 언급되거나 다른 여러 중요한 단어 주변에 등장하는 단어이기 때문에 문서 집합의 키워드로 생각할 수 있다. WordRank와 같이 학습데이터를 이용하지 않으며 키워드를 추출하는 방법인 TextRank(Mihalcea and Tarau, 2004) 역시 mutual reinforcing rela-

tionship 방법으로 그래프의 각 마디의 중요도인 authority를 계산한 뒤, 이를 단어/키워드 가능성으로 이용한다. 두 알고리즘의 차이점은 TextRank는 단어가 인식된 상황에서 각 단어 간의 관계를 그래프로 표현한 뒤 키워드를 추출하지만, WordRank는 단어를 추출함과 동시에 주어진 문서집합에서의 키워드를 추출한다. 하지만 B로 표시된 글자는 '명사+조사'로 결합된 형태이고 C로 표시된 글자는 어미 혹은 조사이다. '원빈은'과 같이 단어에 조사가 결합되었다고 하더라도 좌, 우로 의미 있는 단어가 나타날 가능성이 높기 때문에, '원빈'이 단어로 추출된 후에도 높은 authority를 지니는 허브 마디인 '원빈은' 혹은 '원빈이' 역시 단어로 추출된다. 또한 어미나 조사의 경우에는 왼쪽에 다양한 어근이나 명사가 등장하고, 오른쪽에는 새로운 단어나 어근, 구문이 등장하기 때문에 단어 후보 그래프의 허브에 위치한다. 그렇기 때문에 단어 가능성이 높은 글자들에 어미, 조사가 포함되어 있다.

일반적으로 텍스트 데이터를 백터로 만드는 경우에는 명사나 어근이 어미나 조사보다 중요한 역할을 한다. 이를 위하여 백터화 과정에서 품사 태깅을 이용하여 불필요한 품사를 제거하기도 한다. 하지만 품사 정보를 사용하기 어려운 상황에서는 단어 가능성이 높은 글자들을 {명사, 어근}와 같이 의미를 지니는 단어 집합과 {어미, 조사}과 같이 문법적 기능을 하는 단어 집합으로 분류할 수 있어야 한다. 한 가지 가능한 방법은 구축하기 쉽도록 규모가 작은 학습데이터의 일부를 이용하는 것이다. {명사, 동사, 어근}의 경우에는 새로운 단어들이 나타나기 쉬운 열린 집합(open class)에 속한다. 하지만 기능적 역할을 하는 {어미, 조사}의 경우에는 새로운 단어가 나타날 가능성이 적은 닫힌 집합(close class)에 속하고(Jurafsky and Martin, 2009), 이에 대한 사전을 구축하는 것은 상대적으로 적은 비용이 든다. 즉 새로운 단어들이 만들어질 가능성이 적으므로 규칙을 이용하여 제거가 가능하다. 그렇지만 규칙기반으로 어미나 조사를 제거하는 것은 Porter's stemmer(Porter, 1980)와 비슷한 위험이 따르는데 이는 규칙기반 방법은 불확실하고 애매한 상황을 인식할 수 없다는 문제점을 안고 있다는 것이다. 한 예로 영화 '마음이'에 관련된 텍스트를 분석한다고 하자.

- (예제 문장 1) 마음이 정말 따뜻해졌다.
- (예제 문장 2) 마음이 정말 명작이다.
- (예제 문장 3) 마음이는 정말 명작이다.

예제 문장 1은 영화 제목이 아니라 사람의 마음이 따뜻해졌다는 의미로 해석할 수 있다. 이때는 '명사+-이/조사'의 문법 규칙을 이용하여 '마음'이라는 명사를 추출할 수 있다. 하지만 예제 문장 2에 이와 같은 규칙을 적용시키면 영화 제목 '마음이'를 지칭함에도 불구하고 이를 제대로 인식할 수 없다. 실제로 예제 2는 예제 3의 문장의 비문이지만 한국어는 조사를 사용하지 않아도 문장의 이해가 쉽게 되기 때문에 위와 비슷한 상황은 자주 발생한다. 이를 해결하기 위하여 영화 '마음이'에서 '마음이'라는 단어가 등장하면 모두 영화 제목으로 인식한다는 규칙을 적용하면 예제 문장 1의 '마음'을 제대로 인식할 수 없다. 그렇기 때문에 규칙이 아닌 데이터로부터 {명사, 어근} 등의 의미를 지니는 단어 집합과 {어미, 조사}와 같이 문법적 기능을 하는 단어 집합을 구분할 수 있는 방법이 필요하다.

3.2 KR-WordRank : 부분 글자의 위치 정보를 이용한 단어 추출 능력 향상

WordRank는 중국어나 일본어처럼 사용되는 글자가 다양하며 띄어쓰기를 이용하지 않는 언어로부터 단어를 추출하기 위한 방법으로 제안되었다. 그렇기 때문에 WordRank와 같은 방법론을 한국어의 단어 추출 문제에 적용하기 위하여 중국어와 한국어의 차이점을 살펴보아야 한다. 첫째, 중국어는 표의문자이기 때문에 한 글자가 모두 의미를 가진다. 한국어 역시 각 글자가 대부분 의미를 지니기 때문에 사실상 표의문자에 가깝다. 왜냐하면 한자의 음을 글자로 옮겼기 때문이다. 이에 더하여 1음절 단어의 의미 또한 다양하다. '의'의 경우에는 의리, 옷, 혹은 명사와 결합하는 조사 등 의미가 다양하다. 그렇기 때문에 기존의 사전을 이용한 띄어쓰기 교정이나 품사 태깅의 경우에는 사전에 등록되어 있지 않는 단어를 1음절 단어들로 나누어 인식할 위험이 있다. 실제로 중국어의 경우 약 2만자의 글자가 사용되며, 그 중에서도 7000자 정도가 자주 사용되고, 약 2500 글자가 중국어의 98%를 차지한다. 하지만 한글에서는 약 642개의 1음절 글자가 전체의 98%를 차지하며, 그 중에서도 자주 사용되는 글자의 수는 매우 적다. 즉 1음절 글자의 경우 다의어로 생각할 수 있다. <Figure 4>는 세종 말뭉치에 등장한 총 2,237개의 1음절 글자 중에서 빈도수 기준 상위 k개의 글자에 따른 누적비율이다. 둘째, 중국어는 공식적으로 띄어쓰기를 사용하지 않는다. 이에 비해 한국어는 공식적으로 띄어쓰기를 사용하지만, 웹 공간의 언어에서 띄어쓰기를 지키지 않는 경우가 있다. 한국어의 경우 띄어쓰기 오류가 있더라 하더라도 일부 주어진 띄어쓰기 정보는 단어 추출의 중요한 힌트가 될 수 있다. 그렇기 때문에 일부 주어진 띄어쓰기 정보를 함께 사용하는 방법은 한국어의 단어 추출 방법에 큰 도움이 될 수 있다.

앞서 언급한 바와 같이 중국어와 한국어의 가장 큰 차이점은 완벽하지 않다 하더라도 띄어쓰기 정보가 있다는 것이다. 이 정보를 이용하여 본 연구에서는 한국어에 적용이 가능한 형태의 KR-WordRank를 제안한다. 이 방법의 특징은 {명사, 어근} 집합과 {어미, 조사} 집합을 분류하기 위하여 각 단어 후보의 위치 정보를 이용한다는 것이다. 한국어의 명사구는 명사 단독으로 사용되거나 '명사+조사' 혹은 '명사+명사'의 형태이고, 동사구나 형용사구는 '어근+어미'의 형태이다. 즉, 명사나 어근은 띄어쓰기를 기준으로 나눈 토큰의 왼쪽에 위치하고, 어미나 조사는 오른쪽에 존재한다. 띄어쓰기 오류가 있는 경우에도 위 규칙은 성립한다. 한 예로 '나는 영화를'이란 토큰의

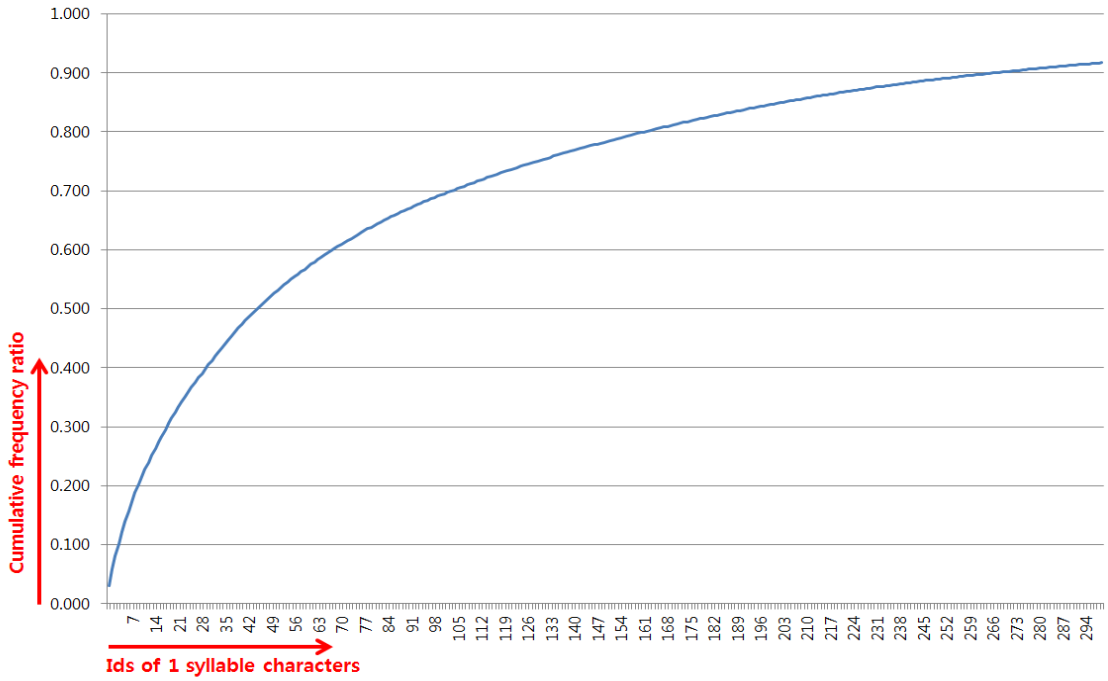


Figure 4. Cumulative frequency ratio of 1 syllable characters in Sejong Corpus

경우 명사 '나'는 왼쪽에 존재하며, 조사 '를'은 오른쪽에 존재한다. 명사 '영화'의 경우 문서집합 전체에서 한 번만 등장한 단어라면 빈도수를 기준으로 단어 후보에서 제외될 것이며, 여러 번 등장한 단어라면 띄어쓰기가 제대로 되어 토큰의 왼쪽에 등장할 가능성이 높기 때문에 다른 문장에 의하여 단어로 추출된 가능성이 높다. 그러므로 KR-WordRank는 단어 후보를 토큰의 왼쪽에 등장하는 후보 집합 L과 토큰의 오른쪽에 등장하는 후보 집합 R로 분리한다. 즉 L은 명사나 어근의 후보 집합이고, R은 어미나 조사의 후보 집합이다. 명사는 조사와 결합하지 않고 단독으로 사용되기도 하므로, 토큰 자체가 단어 후보인 경우에는 L집합의 원소로 취급한다. 즉 KR-WordRank의 단어 후보 그래프는 다음 식 (5)와 같이 V_L, V_R 두 종류의 마디와 E_{LN}, E_{RN} 두 종류의 호로 구성된다.

$$G(V_L, V_R, E_{LN}, E_{RN}) \quad (5)$$

이를 통하여 다의적인 의미를 지니는 글자를 구분할 수 있다. 예시 문장 '다 크다'의 경우 첫 글자 '다'는 토큰의 왼쪽에 나타났으므로 '다/L'이고, 마지막 '다'는 '크다'의 어미 '다/R'이다. 그러나 위 방법으로도 <Table 1>(b)의 A, B의 문제를 해결할 수 없다. <Table 1>(b)에서 '영화'라는 단어가 추출되었음에도 불구하고 '영화는', '영화가', '영화를' 역시 허브이기 때문에 단어로 추출되었다. 하지만 '영화'라는 단어가 추출되었기 때문에, 위 세 개의 단어는 중복되어 추출된 단어라고 판단할 수 있다. 이런 '단어+조사'의 경우를 제거하기 위하여 각 단어 후보의 위치(L, R) 정보를 이용할 수 있다. 예를 들어 집합 R에 해당하는 상위 k개의 단어는 대부분 조사와 어미에 해당되므로 이를 조사와 어미로 추출하는 방식을 사용하는 것이다. 상위 k개의 숫자는 데이터에 기반한 수치로 문서의 속성에 따라 유연하게 결정될 수 있다. 이렇게 추출된 조사와 어미를 이용하여 <Figure 5>에 나타난 방법으로 L집합에서 명사와 어근 추출 방법을 보완한다. 이를 바탕으로 본 연구에서 제안하는 KR-WordRank는 <Figure 6>에 나타난 바와 같이 여섯 단계의 절차를 통해 정확한 한글 단어를 추출하게 된다.

Input :
- W_{All} (후보 단어 집합) : (w_1, w_2, ⋯, w_m)
- W_L (단어 가능성으로 정렬된 집합 L의 단어 후보 리스트) : (w_{l1}, w_{l2}, ⋯, w_{ln})
- W_R (단어 가능성으로 정렬된 집합 R의 단어 후보 리스트) : (w_{r1}, w_{r2}, ⋯, w_{rp})
- W_{RK} (W_R 중 상위 k개의 단어 후보 리스트) : (w_{r1}, w_{r2}, ⋯, w_{rk})
- $W_{All} = W_L \cup W_R$, $W_{RK} \subset W_R$.

Output :
- $W_{Extract}$ (최종적으로 추출된 단어 후보 집합)

<Algorithm>

for each word candidate w_{li} in W_L
  if w_{il} is not form of $w_{li}+w_{rs}$ (w_{li} in W_L & authority(w_{li}) > authority(w_{li}), w_{rs} in W_{RK})
    Add wil to $W_{Extract}$
  end if
end for

Figure 5. A word filtering process for L group using the top candidate words identified in R group

- **1단계 : 띄어쓰기를 이용하여 문장을 여러 개의 글자 집합인 토큰(token)으로 구분** - '원빈은 참 멋있다'의 문장의 경우, 띄어쓰기를 이용하지 않는 WordRank의 경우 '원빈은 참'이라는 잘못된 부분 글자를 단어 후보에 포함시키지만 KR-WordRank에서는 '원', '원빈', '원빈은', '빈은', '참' 등과 같이 띄어쓰기로 구분된 각 토큰에서의 부분 글자만을 고려한다. 이 과정을 통하여 띄어쓰기가 완벽히 된 단어의 경우에는 불필요한 단어 후보를 고려하지 않을 수 있다.
- **2단계 : 각 부분 글자의 위치 확인** - '원빈은'이라는 토큰에서 '원빈'은 토큰의 왼쪽에 위치한다. 이러한 부분 글자들의 집합을 L(left)이라 명하고, '원빈은'에서 '은'과 같은 경우 R(right)이라 명한다. 각 부분 글자에 대하여 (L, R)의 위치 태그를 부여하여 단어 후보를 만든 뒤 각 단어 후보의 빈도수를 계산한다. 이 때, 부분글자가 토큰 전체일 경우 L에 속하는 것으로 정의한다.
- **3단계 : 2단계로부터 얻어진 부분 글자 집합에 대하여 Substring Reduction 수행**-제 2장에서 설명한 바와 같이 한 부분글자 A가 다른 부분글자 B의 진부분집합이며 A의 출현 빈도와 B의 출현 빈도가 같을 경우 A를 단어 후보에서 제외한다. <Figure 6>의 예시에서 '원/L'은 '원빈/L'의 부분집합이고 두 단어 후보의 출현 빈도가 2로 같기 때문에 '원/L'이 단어 후보에서 제외되었다. 반면, '원빈/L'은 '원빈은/L'의 부분집합이기는 하나 출현 빈도가 더 크기 때문에 단어 후보에서 제외되지 않는 것이다.
- **4단계 : 각 단어 후보 간의 호 생성**-<Figure 6>의 예시에서 3단계를 통해 집합 L에는 {'원빈', '원빈은', '참', '멋있', '멋있다', '멋있는'}이 속하게 되고 집합 R에는 {'빈은', '있는', '있다', '빈'}이 속하게 된다. 이를 이용하여 다음 세 가지 조건을 만족하는 단어 후보들 사이에 호를 연결한다. 첫째, 이론적으로 단어 후보 간 호는 L-L, L-R, R-L, R-R의 네 가지 조합을 통해 생성될 수 있으나 하나의 토큰을 L집합에 속하는 것으로 가정할 경우, R-R 조합은 생성될 수 없으므로 이 조합은 고려 대상에서 제외한다. 둘째, 실제 텍스트에 1회 이상 출현한 순서에 대해서만 호를 연결한다. 즉, '빈은-참'이라는 호는 R-L 조합이고 실제 텍스트인 '원빈은 참 멋있다'에서 출현하였기 때문에 호를 연결하나, '원빈-있다'의 호는 L-R 조합으로 생성 가능한 호이지만 실제 텍스트에서 출현한 적이 없기 때문에 호를 생성하지 않는다. 셋째, 호 연결은 3단계에서 제거되지 않은 단어 후보들만을 고려한다. <Figure 6>의 예시에서는 L-R 조합을 통한 호를 연결하는 것이 가능하나, 모든 호가 실제 텍스트에서 한 번도 출현하지 않았거나(예 : '원빈-있다'), 3단계에서 제거된 단어 후보이기 때문에(예 : '원빈-은') 호가 연결되지 않은 것이다. 이 단계를 통해 기존의 WordRank와는 달리 '원빈-은참멋'과 같은 불필요한 단어 후보 간의 호를 생성하지 않기 때문에 정확한 단어 인식 가능성을 높임과 동시에 효율성 또한 증가시킬 수 있을 것으로 기대한다.
- **5단계 : mutually reinforcement relationship을 이용하여 단어 가능성 계산**-이 단계에서 계산 방법은 WordRank에서 사용되는 식 (2), 식 (3)과 동일한 식을 사용한다.
- **6단계 : 집합 R로부터 상위 k개의 단어 후보를 추출하여 집합 L로부터 '명사+조사', '어근+어미'의 결합 단어를 제거하여 최종적인 단어 추출**-이 단계에서는 <Figure 5>에 나타난 알고리즘을 이용하여 중복 가능성이 높은 단어 후보를 제외한다. 여기서 핵심적인 사항은 '명사+조사'('어근+어미')의 단어 가능성이 '명사'('어근')의 단어 가능성보다 높을 경우에는 제거하지 않고 그 형태를 보존하여 단어로 인식한다는 것이다. 예를 들어 '원빈'의 단어 가능성이 '원빈은'의 단어 가능성보다 높게 나타날 경우 '원빈은'은 단어 후보에서 제외된다. 그러나 만약 '원빈은'의 단어 가능성이 '원빈'의 단어 가능성보다 높게 나타날 경우 '원빈은'은 제거되지 않는다. 이는 KR-WordRank는 한글의 문법적인 형태보다는 실제로 사람들이 빈번하게 사용하는 형태를 단어로 인식하기 위한 알고리즘이기 때문이다.

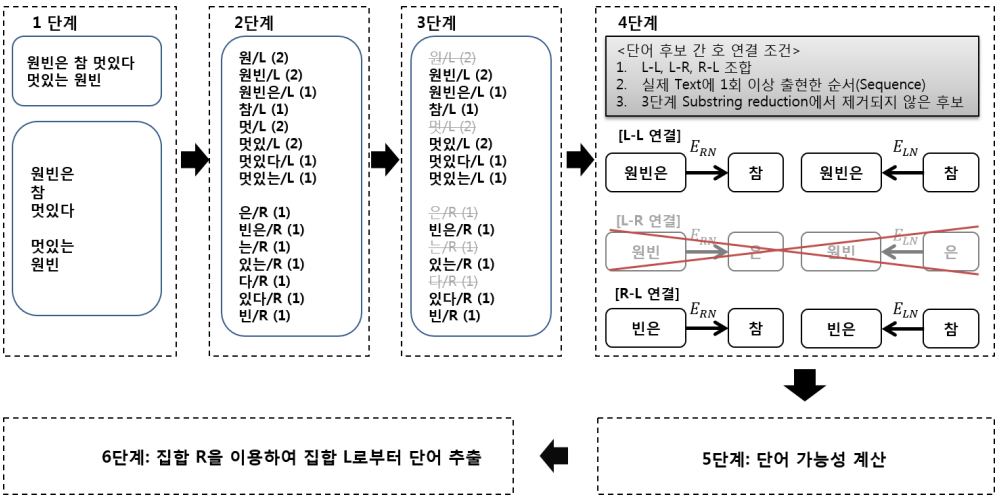


Figure 6. A framework of KR-WordRank

KR-WordRank에서는 기존 WordRank의 수행 절차에 2단계, 4단계, 그리고 6단계가 추가되었다. 2단계와 4단계를 통하여 '멋있다'에서 '있'과 같이 토큰의 중간에 위치한 무의미한 부분 글자를 단어 후보에 포함시키지 않음으로써 의미 있는 단어 후보를 걸러내는 역할을 한다. 또한 '다'라는 글자에 대하여 '멋지다'와 같이 어미로 사용되는 경우와 '다 같이'와 같이 부사로 사용되는 경우를 구분하여 인식하도록 하는 정보를 얻을 수 있다. 또한 6단계를 통하여 조사나 어미의 결합으로 인한 중복적인 단어를 부분적으로 제거할 수 있다.

위와 같은 과정으로 추출된 단어를 정제한 결과는 <Table 1>(c)와 같다. 본 연구에서 제안하는 KR-WordRank를 이용하여 추출된 단어의 경우 조사나 어미가 결합되거나 그 자체가 단어로 추출되는 경우가 없음을 볼 수 있다. 또한 추출된 1음절 단어 역시 데이터를 살펴보면 영화 '아저씨'의 40자평 데이터를 표현하기에 유의미한 단어임을 알 수 있다. 예를 들어 '본', '좀' 그리고 '또'는 '내가 본 영화 중 좀 하는 듯', '또 보고 싶은 영화'와 같이 의미로 사용되었음을 알 수 있다. 즉 이는 40자평의 도메인을 잘 설명하는 단어로 판단된다.

4. KR-WordRank 성능 검증

4.1 실험 설정

(1) 데이터 및 변수 설정

본문에서 WordRank와 KR-WordRank의 차이를 설명하기 위하여 영화 '아저씨'의 40자평 24,302개의 문장으로부터 단어를 추출하는 과정을 예시로 보였다. <Table 1>에서 볼 수 있듯이 각 알고리즘으로부터 추출된 상위 30개 단어에서는 KR-WordRank가 합리적으로 단어를 추출함을 정성적으로 알 수 있지만, 객관적인 성능을 평가하기 위해서 세종 말뭉치를 이용하여 두 알고리즘의 성능을 평가하였다. 세종 말뭉치의 문어 문서 279개에 포함되어있는 837,805개의 문장으로부터 자동으로 단어를 추출하였다. 모든 문장에서 한글을 제외한 숫자, 기호, 외국어는 제거하였다.

WordRank와 KR-WordRank 두 알고리즘이 공통으로 사용하는 변수는 동일하게 설정하였다. 두 알고리즘 모두 길이가 7음절 이하인 글자들을 단어의 후보로 고려하였으며, 문장의 개수를 고려하여 50번 이하로 등장한 모든 글자는 단어 후보에서 제외하였다. 또한 두 알고리즘 모두 Substring Reduction을 통하여 최종적으로 단어 후보를 설정하였다. 두 알고리즘 모두 50번의 반복 계산을 통하여 단어 가능성(authority)의 값이 수렴하도록 하였다.

(2) 평가 지표

WordRank와 KR-WordRank는 각 단어의 형태소를 분리하기 위한 방법이 아니라 의미 있는 단어를 추출하기 위한 방법이기 때문에 품사 태깅과 다른 평가 방법을 사용하였다. 세종 말뭉치에는 기호, 숫자, 그리고 외국어를 제외한 33개의 품사가 <Table 2>에 나타난 바와 같이 존재한다. 이 중에서 조사, 어미, 접두사나 접미사는 문법적 역할을 하지만, 의미를 지니지는 않는다. 이와 다르게 체언, 용언, 관형사, 부사, 감탄사, 그리고 어근은 해석이 가능한 의미를 지닌다. 그렇기 때문에 WordRank

Table 2. Part-of-Speech tag system of Sejong corpus

대분류	태그	설명	대분류	태그	설명
체언	NNG	일반명사	조사	JKS	주격조사
	NNP	고유명사		JKC	보격조사
	NNB	의존명사		JKG	관형격조사
	NR	수사		JKO	목적격조사
	NP	대명사		JKB	부사격조사
용언	VV	동사		JKV	호격조사
	VA	형용사		JKQ	인용격조사
	VX	보조용언		JX	보조사
	VCP	긍정 지정사		JC	접속조사
	VCN	부정 지정사	선어말 어미	EP	선어말어미
관형사	MM	관형사		EF	종결어미
부사	MAG	일반부사		EC	연결어미
	MAJ	접속부사	어말 어미	ETN	명사형 전성 어미
감탄사	IC	감탄사		ETM	관형형 전성 어미
어근	XR	어근	접미사	XSN	명사 파생 접미사
접두사	XPN	체언 접두사		XSV	동사 파생 접미사
				XSA	형용사 파생 접미사

와 KR-WordRank에서 추출해야하는 대상은 해석이 가능한 의미를 지니는 후자이다. 그 중 체언은 '영화'와 같이 독립적으로 사용되거나 '영화/NNG+를/JKO' 처럼 조사와 결합되어 사용된다. 하지만 용언은 '보/VV+다/EF'와 같이 어미와 결합되어야만 사용이 가능하다. 이러한 복합형태소 역시 의미를 해석할 수 있다. 이와 다르게 의미를 해석할 수 없는 복합형태소 역시 존재한다. '말이라고'의 '이/VCP+라고/EC'는 복합형태소이지만 독립적으로 사용되지 못하고, 따라서 의미를 지니지 못한다. 그러므로 WordRank와 KR-WordRank가 추출해야하는 대상은 의미 해석이 가능한 체언, 용언, 관형사, 부사, 감탄사, 그리고 독립적으로 사용가능한 복합형태소이다. 이를 고려하여 단어의 정확도를 계산하는 함수를 식 (6)과 같이 제안한다.

$$accuracy_{word}(w) = \frac{\sum_{c \in C} f_c(w)}{f(w)} \tag{6}$$

$f_c(w)$: 단어 종류가 c인 w의 빈도수
$f(w)$: 문서 집합에서 w의 등장 빈도수
C = {체언, 용언, 관형사, 부사, 감탄사, 독립적으로 사용가능한 복합형태소}

예를 들어서 '한'이라는 단어 후보가 총 124,694번 등장하였다. '한'은 세종 말뭉치에서 체언으로 2,844번, 관형사로 34,742번, 그리고 '하/VV+ㄴ/ETM'이나 '하/VA+ㄴ/ETM' 등의 복합형태소로 7,400번, '한/XPN'과 같은 접두사로 1170번 등장하였다. 이 때 '한'의 단어 정확도 $accuracy_{word}$('한')는 의미 해석이 가능하지 않은 집합인 접두사 '한/XPN'을 제외한 체언(2,844), 관형사(34,742), 의미 해석이 가능한 복합형태소(7,400)의 빈도수의 합(44,986)을 총 빈도수의 합(124,694)으로 나눈 식 (7)과 같이 표현할 수 있다.

$$accuracy_{word}(\text{'한'}) = \frac{2{,}844 + 34{,}742 + 7{,}400}{124{,}694} = \frac{44{,}986}{124{,}694} = 0.361 \tag{7}$$

단어 가능성은 KR-WordRank의 단어 후보 그래프에서의 단어 경계 값을 의미하고, 단어 정확도는 단어 후보로부터 추출된 단어가 실제로 의미를 지니는 단어일 정확도이다. 위의 방법을 통하여 WordRank와 KR-WordRank로부터 추출된 각 단어 후보의 단어 정확도를 계산하였다. 그 뒤 상위 k개의 단어 후보의 평균 단어 정확도를 계산하였다. 이를 통하여 두 방법론의 상위 k개에 대한 단어 추출 능력을 비교하였다.

4.2 실험 결과

WordRank와 KR-WordRank를 이용하여 단어 가능성이 높은 상위 100개의 단어 후보를 추출한 결과는 각각 <Table 3> 및 <Table 4>에 나타나 있다. 제 3.2절의 KR-WordRank의 6단계에

Table 3. Extracted top 100 words by WordRank

이(0.47)	다(0.011)+	문(0.029)	계(0.001)	실(0.008)
도(0.006)	라(0.001)	적(0.049)	관(0.008)	임(0.01)
의(0)	상(0.013)	동(0.014)	치(0.078)	당(0.037)
가(0.075)	수(0.446)	소(0.018)	단(0.034)	미(0.039)
지(0.101)	정(0.013)	신(0.036)	입(0.188)	위(0.128)
기(0.008)	구(0.004)	원(0.141)	유(0.008)	개(0.183)
인(0.01)	조(0.04)	을(0)	산(0.076)	스(0)
사(0.025)	해(0.037)	비(0.042)	식(0.047)	체(0.01)
시(0.053)	전(0.115)	공(0.024)	국(0.007)	행(0.006)
자(0.031)	주(0.302)	거(0.113)	요(0.004)	내(0.393)
고(0.001)	제(0.043)	선(0.041)	용(0.008)	생(0.005)
대(0.046)	장(0.034)	어(0.001)	면(0.026)	분(0.102)
보(0.417)	나(0.233)	마(0.005)	회(0.029)	재(0.012)
과(0.009)	성(0.032)	경(0.002)	진(0.008)	영(0.013)
부(0.016)	에(0)	방(0.082)	간(0.124)	업(0.012)
하(0.46)	리(0.007)	아(0.007)	처(0.004)	안(0.356)
일(0.344)	화(0.013)	세(0.102)	는(0)	우(0.002)
만(0.09)	들(0.087)	연(0.013)	학(0.005)	물(0.106)
한(0.116)	은(0.001)	여(0.004)	교(0.004)	형(0.124)
로(0)	서(0.026)	중(0.212)	무(0.004)	모(0.013)

Table 4. Extracted top 100 words by KR-WordRank when size of R set is 400

그(0.386)	나(0.762)	아(0.747)	잘(0.952)	어느(0.865)
이(0.551)	모두(0.953)	함께(0.999)	어떻게(0.379)	보고(1)
한(0.864)	년(0.964)	사람(0.999)	들어(0.339)	월(0.954)
또(0.64)	그런(0.553)	큰(0.833)	왜(0.289)	그러나(1)
일(0.878)	제(0.934)	해(0.788)	것은(0.847)	사실(0.083)
가(0.977)	후(0.989)	바로(0.941)	이러한(1)	차(0.814)
다(0.948)+	어떤(0.528)	많은(1)	즉(0.902)	게(0.238)
내(1)	개(0.225)	채(1)	간(0.855)	데(1)
전(1)	저(0.977)	할(1)	곧(0.288)	선(1)
의(0.614)	하는(0.987)	하나(0.905)	날(1)	주(1)
등(0.972)	이런(1)	지금(0.739)	난(0.234)	아직(1)
때(1)	하고(0.288)	온(0.826)	없는(0.992)	경우(1)
더(0.284)	건(0.932)	따라(1)	도(1)	먼저(1)
있는(0.441)	뒤(0.999)	새(1)	집(0.783)	여러(1)
중(0.788)	시(0.984)	이렇게(0.928)	특히(0.477)	있어(1)
나는(0.978)	안(0.742)	자신의(0.997)	역시(1)	같이(0.948)
우리(0.645)	자(1)	것이(1)	번(0.999)	원(0.993)
다른(1)	두(0.973)	세(1)	본(0.962)	만(0.999)
같은(1)	말(0.222)	그렇게(1)	인(0.999)	걸(0.831)
대(1)	모든(1)	새로운(0.871)	은(1)	참(1)

Table 5. Extracted top 100 candidate words in R set by KR-WordRank

도	로	와	요	들은	지만	이라는	에서도	이다	할
는	에서	까지	서는	게	서도	든	시	이라고	사
의	을	에는	처럼	자	들	들의	여	상	두
에	다	하고	부터	에도	보다	선	대로	하며	장
가	를	라	며	에서는	하여	지는	보다는	하면	로서
이	과	한	면	기	니	적인	면서	라면	로는
나	지	이나	해	이고	이며	다는	적으로	간	거나
서	고	라는	하는	란	에게	하게	된	다고	지를
은	으로	들이	라고	일	들을	해서	진	리	가는
만	인	야	데	라도	적	어	이란	인데	엔

사용할 R 집합의 단어는 상위 400개를 사용하였으며, 그 중 상위 100개를 <Table 5>에 기술하였다. <Table 3>과 <Table 4>에는 단어 후보와 각 단어 후보의 단어 정확도를 괄호 안에 기술하였다. <Table 3>에서 볼 수 있듯이 WordRank의 경우 해석이 모호하거나 조사, 어미로 자주 사용되는 단어들이 상위에 추출되었음을 볼 수 있다. 이러한 결과의 원인은 제 3.1절에서 언급하였듯이 한국어의 경우 대부분의 1음절 글자가 의미를 지님에도 불구하고 WordRank는 가능한 모든 단어 후보를 추출하고, 그 단어 후보끼리의 네트워크를 형성하였기 때문이다. 한 예로 WordRank는 '서울대학교'의 경우 '서-울-대-학-교'나 '서-울대-학교'의 단어 후보의 연결고리를 만들 수 있다. 그렇기 때문에 1음절 글자의 단어 가능성이 높아지는 것이며, '울대'와 같이 단어의 가능성이 전혀 없는 음절 조합이 단어 후보로 만들어지기도 한다. 하지만 한국어의 경우 의미를 해석할 수 있는 단어나 복합형태소는 토큰의 왼쪽에 등장하고 문법적 기능을 하는 단어나 복합형태소는 토큰의 오른쪽에 등장한다. 이 규칙은 띄어쓰기가 되어있지 않다고 하더라도 동일하게 적용된다. 예를 들어 '영화를보다'의 경우 '영화/NNG'는 토큰의 왼쪽에, '다/EC' 는 토큰의 오른쪽에 등장한다. 즉 KR-WordRank는 단어 후보의 종류를 토큰의 왼쪽과 오른쪽에 등장하는 두 집합 L, R로 나눔으로써 불필요한 단어 후보를 선택하지 않았다. 그 결과 WordRank의 경우 7음절 이하인 3,168,314개의 글

자 중에서 빈도수가 50번 이하인 3,094,231개의 글자를 단어 후보에서 제외하였고, Substring Reduction을 통하여 4,552개의 글자가 추가로 단어 후보에서 제외되어 최종적으로 69,531개의 단어 후보를 선택하였다. KR-WordRank의 경우 7음절 이하인 2,754,395개의 글자 중에서 동일한 과정을 거쳐 최종적으로 62,404개의 단어 후보를 선정하였다. 그 중 집합 L에 해당하는 단어 후보는 44,024개이다. 이는 불필요한 단어 후보가 제거된 효과이다. 또한 '영화를보다'에서 KR-WordRank는 '보/VV'를 단어 후보에 포함되지 않는다. 하지만 띄어쓰기의 오류를 포함하는 토큰의 숫자가 띄어쓰기를 제대로 지킨 토큰의 숫자보다 적다면 '영화를보다'가 아닌 다른 토큰에서 '보/V'가 단어 후보로 선택될 것이다. 즉 KR-WordRank는 어느 정도의 띄어쓰기 오류에 큰 영향을 받지 않는다.

<Table 3>에서 +표시된 WordRank로부터 추출된 '다'는 '다 좋다'의 '다/MAG'인지, '어제다'의 '다/EF'인지 알 수 없기 때문에 단어 정확도가 0.011로 계산되었다. '다'는 <Table 4>에서 KR-WordRank에서 추출된 상위 100개 단어와 <Table 5>에서 모두 나타난다. 이 때 <Table 4>의 '다'는 '다 좋은'과 같이 부사로 사용되는 '다'를 의미하며 '좋다'와 같이 어미로 사용되는 '다/EF'의 경우 집합 R에 포함되기 때문에 KR-WordRank로부터 단어로 추출된 '다'는 의미 있는 단어 후보만으로 제한되었음을 알 수 있다. 즉 다의적인 성격을 지니는 한국어의 1음절 글자의 경우 집합 L과 R로 분리함으로써 그 용도가 분리되어 인식됨을 확인할 수 있다. 그렇기 때문에 <Table 4>의 KR-WordRank로부터 추출된 '다'는 단어 정확도가 0.948로 높게 나타났다. 이와 같은 다른 예제로는 '같은'이 있다. '같은'의 경우 '같은 반'과 같이 '같/VA+은/ETM'으로 사용될 수 있고, '조각같은'과 같이 '같/XSA+은/ETM'으로 명사 뒤에서 앞의 명사 '조각'을 형용사화 시켜줄 수도 있다. 이 때 전자의 '같은/R'은 KR-Word Rank에서 단어로써 추출하지만, 후자의 '같은/L'은 제외된다. 하지만 WordRank는 이러한 구분을 하지 않으며 모두 동일한 형태의 '같은'이라는 단어 후보로 판단한다.

KR-WordRank에서 단어 추출 성능에 영향을 주는 변수 중 하나는 6단계에서 사용하는 집합 R의 상위 단어 후보의 개수이다. 그렇기 때문에 <Figure 7>에서 집합 R의 단어 후보 개수를 각각 300개부터 500개까지 100개 단위로 증가시키며 각각의 단어 추출 성능을 비교하였다. 그림의 x축은 각 집합의 상위 단어 후보의 개수이고, y축은 상위 k 단어 후보를 단어로 추출하였을 때의 평균 단어 정확도이다. 이를 통하여 6단계에서 집합 R의 단어를 얼마만큼 이용하느냐에 따라서 전체적인 성능이 변할 수 있음을 확인할 수 있다. 이는 일반적으로 사용되는 어미, 조사, 혹은 복합형태소를 충분히 추출할 경우 '원빈', '원빈이'와 같이 이미 단어로 추출된 명사에 문법적 기능을 하는 형태소가 결합된 복합형태소를 걸러줄 수 있기 때문이라 판단된다.

WordRank를 이용하여 한국어 단어를 추출하였을 때에는 <Table 3>에 기술된 것처럼 조사, 어미로 사용되는 경우가 많은 단어들이 상위로 추출되었다. 어떠한 사전 정보도 이용하지 못하는 상황을 가정하여 세종 말뭉치 대신 WordRank로부터 추출된 단어로부터 토큰의 왼쪽에 자주 등장한(KR-WordRank에서 L에 해당하는 빈도수가 R에 해당하는 빈도수보다 높은 경우) 단어들을 제거한 뒤 평균 단어 정확도를 계산한 결

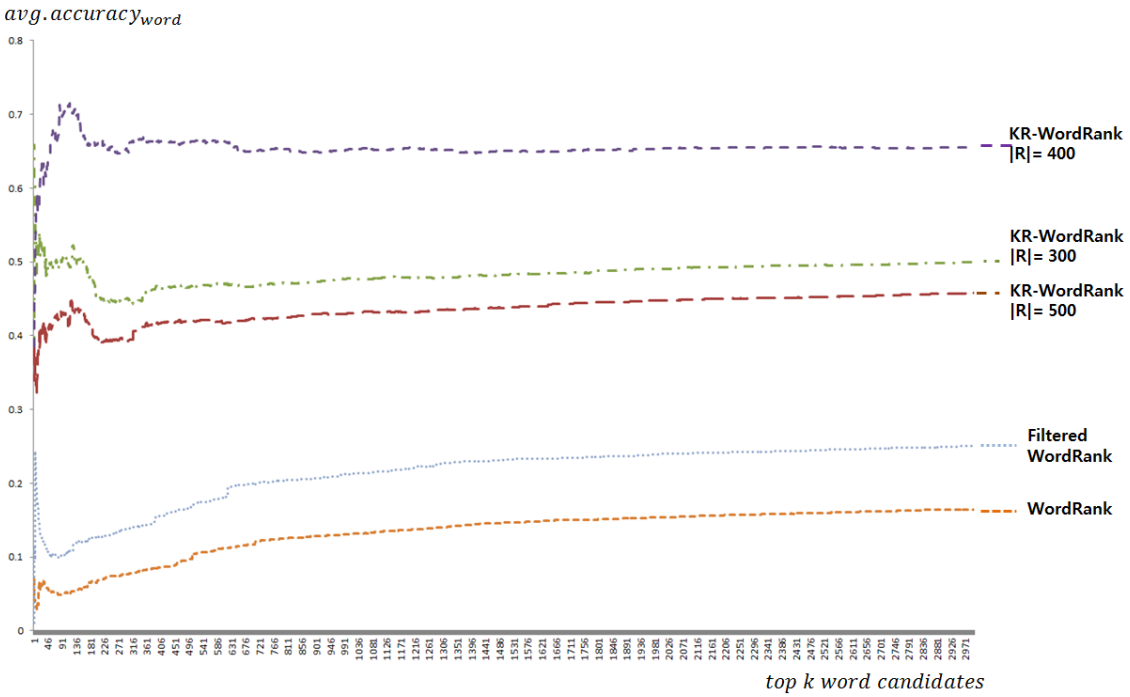


Figure 7. Average word extraction accuracy

과 거의 0에 가까운 평균 단어 정확도를 보였다. 제 3.1절에서 언급한 바와 같이 조사나 어미는 닫힌 집합에 속하는 단어이기 때문에(Jurafsky and Martin, 2009) 일반적으로 새로운 단어가 생성되지 않는 조사나 어미의 사전을 세종 말뭉치를 통하여 구축한 뒤, WordRank로부터 추출된 단어에서 세종 말뭉치에서 자주 등장한 상위 400개의 조사와 어미를 삭제한 결과, <Table 6>처럼 '간'이나 '들'과 같이 여전히 1음절이며 의미를 해석하기 어려운 글자들이 남아있음을 볼 수 있다. 이는 제 3.1절에서 언급한 바와 같이 토큰의 중간에 위치한 부분 글자들 역시 단어의 후보로 선택되었고, 다의적인 의미를 지니는 부분 글자의 좌, 우로 다른 단어들이 나타날 가능성이 많았기 때문이다. 즉 WordRank는 불필요한 단어 후보를 효과적으로 제거하지 못함으로서 <Figure 7>의 Filtered WordRank와 같이 조사나 어미를 제거한다고 하더라도 평균 단어 정확도가 많이 높아지지 않으며, 그 값이 KR-WordRank보다 항상 낮음을 볼 수 있다.

4.3 토의

KR-WordRank를 이용하여 세종 말뭉치로부터 단어를 추출하였을 때 '그것/NP+이/JKC(101위)', '그것/NP+을/JKO(102위)', 혹은 '그것/NP+도/JX(414위)'의 단어 가능성이 '그것/NP(489위)'보다 높게 나타남을 확인할 수 있다. 이는 KR-WordRank는 실제로 자주 사용되는 표현을 단어로 추출하기 때문이다. 실제로 각각의 사용 빈도수는 '그것이(4,013번)', '그것을(3,881번)', '그것도(1,240번)' 임과 비교하여 '그것'이 단독으로 사용된 경우는 426번이다. 즉, '그것/NP'은 독립적으로 사용되기보다는 일반적으로 다른 조사와 함께 결합되어서 사용되는 것이다. 다른 예제로 '같/VA(7,094위)'보다 '같/VA+은/EC(19위)', '같/VA+아/EC(1,781위)'가 단어 가능성이 더 높게 계산되었다. 즉 KR-WordRank에서 추출하는 대상을 정리하면 ***'의미를 지니며 실제로 자주 사용되는 독립적인 단어 혹은 복합형태소'***이다. KR-WordRank는 언어학적으로 형태소라는 것으로 정의되는 단어가 아닌 실제로 사람들이 사용하는 표현으로의 단어를 추출한다. 세종 말뭉치에서의 단어 추출 결과와 반대로 영화 '아저씨'에서는 '원빈'이 '원빈을', '원빈이' 보다 단어 가능성이 높게 계산되었고, '을', '이'가 집합 R의 단어로 추출되었기 때문에 6단계에서 '원빈을', '원빈이'는 단어로 선택이 되지 않았다. 이는 40자평 문서 집합에서는 '원빈'이 독립적으로 사용되는 경우가 많고, '원빈'과 주로 결합되는 일반적인 조사나 다른 명사가 없기 때문이다. 이는 40자평 문서 집합에서는 '원빈'이 '원빈이', '원빈을'보다 문서 집합을 요약할 수 있는 표현이라 생각할 수 있다.

KR-WordRank가 단어 추출을 잘 하기 위해서는 충분한 수의 문서가 제공되어야 하지만, 직관적으로 해석할 수 있는 키워드를 추출하기 위해서는 주어진 문서 집합에 포함된 주제의 종류가 적을수록 성능이 우수하다. 40자평 문서 집합의 경우에는 대부분의 문서에서 영화 '아저씨'와 관련된 이야기나 감

Table 6. Top 100 words which are extracted by WordRank and filtered with Sejong Corpus suffix dictionary

인(0.01)	동(0.014)	회(0.029)	영(0.013)	집(0.391)
사(0.025)	신(0.036)	진(0.008)	업(0.012)	할(0.256)
보(0.417)	원(0.141)	간(0.124)	안(0.356)	법(0.159)
부(0.016)	비(0.042)	처(0.004)	물(0.106)	배(0.158)
하(0.46)	공(0.024)	학(0.005)	형(0.124)	역(0.03)
일(0.344)	거(0.113)	교(0.004)	모(0.013)	합(0.009)
한(0.116)	선(0.041)	무(0.004)	명(0.21)	심(0.034)
상(0.013)	경(0.002)	실(0.008)	통(0.017)	민(0.012)
정(0.013)	방(0.082)	임(0.01)	발(0.051)	강(0.099)
조(0.04)	연(0.013)	당(0.037)	호(0.058)	권(0.069)
해(0.037)	중(0.212)	미(0.039)	반(0.07)	예(0.143)
전(0.115)	계(0.001)	위(0.128)	양(0.091)	천(0.22)
주(0.302)	관(0.008)	개(0.183)	이다(0.005)	론(0.002)
장(0.034)	치(0.078)	스(0)	감(0.047)	종(0.039)
성(0.032)	단(0.034)	체(0.01)	현(0.029)	파(0.06)
리(0.007)	입(0.188)	행(0.006)	설(0.025)	함(0.008)
화(0.013)	산(0.076)	내(0.393)	까(0.003)	질(0.042)
들(0.087)	식(0.047)	생(0.005)	작(0.096)	속(0.373)
문(0.029)	국(0.007)	분(0.102)	금(0.011)	등(0.691)
적(0.049)	용(0.008)	재(0.012)	차(0.202)	포(0.011)

상을 말하였다. 그렇기 때문에 <Table 1>(c)와 같이 영화 '아저씨'와 관련이 있거나 영화에 관련된 표현이 높은 순위의 단어인 키워드로 추출되었음을 볼 수 있다. 그러나 <Table 4>의 경우에 볼 수 있듯이 세종 말뭉치에서의 키워드는 어떤 주제에서도 공통적으로 사용될 수 있는 단어임을 볼 수 있다. 이는 세종 말뭉치를 구성하는 문서 집합은 다양한 주제의 문서들로 구성되어 있기 때문에 문서 집합에서 공통적으로 포함하고 있는 표현들이 키워드로 추출될 가능성이 높다. 즉 KR-WordRank를 이용하여 단어 추출 및 키워드 추출을 함께 하기 위해서는 비슷한 주제의 문서들을 사전에 분류함으로써 그 성능을 높일 수 있을 것으로 기대한다.

단어 추출을 위한 계산 속도는 단어 후보 그래프를 구성하는 마디와 호의 수에 의존한다. KR-WordRank는 WordRank와 비교하여 같은 형태의 글자에 대하여 위치 정보를 이용하여 집합 L과 R로 분리하기 때문에 그래프를 구성하는 마디의 수를 늘릴 수 있다. 하지만 토큰의 중간에 위치하는 부분 글자들을 단어 후보로 포함하지 않기 때문에 그래프의 마디의 수를 줄이는 효과 역시 있다. 또한 모든 부분 글자들이 집합 L과 R에 동시에 등장하지 않는다. 예를 들어 '~은데'와 같은 단어는 독립적으로 사용되지 않고 오른쪽에 다른 글자와 결합하기 때문에 집합 R에만 등장한다. 실험적으로 두 알고리즘을 비교하기 위하여 두 알고리즘 모두 7음절 이하이고 최소 30번 이상 등장한 단어 후보로부터 단어를 추출하였다. KR-WordRank를 이용할 경우 마디는 94,525개인 반면 WordRank의 마디는 99,412개였다. 하지만 KR-WordRank를 이용할 경우 호의 개수는 36,602,516개인 반면 WordRank는 7,490,244개로 KR-WordRank의 경우 단어 후보들을 연결하는 호의 개수가 증가했음을 알 수 있다. 이는 KR-WordRank가 단어 후보 개수를 늘리는 것이 아니라 오히려 불필요한 후보를 제거하였음을 의미한다. 또한 호의 개수가 늘어난 것은 <Figure 6>의 4단계에서 L-L, L-R, R-L의 모든 조합에 대하여 단어 후보들을 연결하여 단어 후보의 외부 경계 값을 설명하는 이웃 정보를 더 많이 구축하였기 때문이다. 그렇기 때문에 Java를 이용한 구현 예제에서 50번의 반복 계산에 대하여 71.02초 만에 계산되는 WordRank와 비교하여 KR-WordRank는 336.5초인 약 5배의 계산 시간을 필요로 하였고, 단어 후보 간의 호의 정보를 저장하기 위하여 메모리 사용 역시 2.3GB에서 3.8GB로 약 1.65배 증가하였다. 하지만 세종 말뭉치와 같이 거대한 텍스트 데이터에 적용하기에 두 알고리즘 모두 큰 무리가 없으며, KR-WordRank이 추가적인 메모리와 계산 시간을 요구하지만 현실적으로 사용하는데 큰 지장이 없는 수준으로 판단된다.

KR-WordRank나 WordRank뿐 아니라 Accessor Variety나 Branch Entropy 방법의 공통된 한계점 중 하나는 자주 등장하지 않는 단어의 경우 정상적으로 추출될 가능성이 낮다는 것이다. KR-WordRank나 WordRank는 단어 주위에 여러 단어들이 있어야만 각 단어의 authority를 전파받을 수 있다. 또한 이러한 단어는 Substring Reduction과정에서 단어 후보에서 지워지기도 한다. <Figure 3>의 경우, 오로지 두 문장만 존재할 경우, '나'는 단어임에도 불구하고 '나는'과 빈도수가 같기 때문에 단어 후보에서 탈락된다. 이러한 이유로 자주 등장하지 않는 단어는 제대로 추출되지 않을 위험이 있다. 이 문제를 다른 시각으로 해석할 수 있다. 주어진 문서집합의 텍스트 데이터를 벡터화하는 과정에서 자주 나오지 않는 단어는 불필요하기 때문에 자주 등장한 단어만을 이용하여 구조화 하는 것으로 해석할 수 있다. 하지만 만약 자주 등장하지 않는 단어에 대해서도 추출을 하려 한다면 다른 단어 추출 방법을 함께 사용해야 할 필요가 있다. KR-WordRank의 두 번째 한계점은 용언의 활용이 일어난 경우 이를 인식하지 못한다는 점이다. '예쁜 꽃'의 '예쁜'은 '예쁘/VA+ㄴ/ETM'이지만 활용에 의하여 '예쁜'으로 사용된다. 하지만 KR-WordRank는 음절 단위로 글자를 인식하고, 연속된 글자를 조합하여 단어 후보로 선택한다. '예쁜'이 충분히 많이 사용되었다면 문서 집합에서 단어로 추출이 될 수 있다. 한 예로 <Table 1>(c)의 영화 '아저씨' 40자평 문서에서 단어로 추출된 '본'은 '내가 본 영화'와 같이 사용되는 '보/VV+ㄴ/ETM'이지만 단어로 추출되었다. 하지만 '본'과 '봤던' 혹은 '예쁜'과 '예쁘고'를 동일한 의미를 지닌 단어로 인식할 수 있으려면 용언의 활용을 인식할 수 있도록 해야 한다. 마지막으로 KR-WordRank는 사용자가 경험적으로 정하는 변수에 의하여 단어 추출 성능이 좌우 될 수 있다. 명사나 어미 등을 추출하기 위하여 조사나 어미 단어 집합인 집합 R의 상위 k개의 단어를 사용한다. 변수 k 값이 클 경우 지나치게 많은 단어들이 집합 L의 단어에서 탈락될 수 있으며, 지나치게 작을 경우, 조사나 어미가 결합된 단어들이 걸러지지 않을 수 있다. 이에 대하여 경험적으로 변수 값을 정하는 것이 아니라, 문서 집합을 효과적으로 설명할 수 있는 최적 값의 변수 k를 정하는 방법의 연구가 필요하다.

5. 결 론

본 연구에서는 한국어에 적합한 단어 추출방식인 KR-WordRank 방법을 제안하고 이의 효과를 검증하였다. 중국어와 일본어에서 효과적이며 효율적인 단어 추출 성능을 보인 WorkRank의 경우 한국어 단어 추출에 대해서는 몇 가지 한계점이 있는 것으로 나타났다. 특히 표의문자에 가까운 한국어의 경우 1음절 글자들이 단어로 추출되는 경우가 많으며, 어미나 조사의 경우 독립적으로 단어로 사용되는 경우 역시 많음을 알 수 있다. 하지만 많은 응용연구의 경우 문서 집합으로 추출되어야 하는 대상은 조사나 어미 같은 문법적 기능을 하는 단어가 아니라 명사나 어근과 같이 의미를 지니는 단어일 경우가 많다. 그렇기 때문에 한국어 단어 추출 문제의 경우 문법적 기능을 하는 단어는 추출 대상에서 제외할 수 있어야 한다. 본 연구에서는 이러한 문제를 해결하기 위하여 단어 후보군을 L, R 두 집합으로 나누고, 어미나 조사의 역할을 할 것으로 기대하는

집합 R의 추출된 단어를 이용하여 집합 L로부터 의미를 지니는 단어를 추출하는 KR-WordRank 기법을 제안하였다.

제안된 KR-WordRank를 실제 데이터에 적용한 결과, 기존의 WordRank보다 정교한 한국어 단어 추출이 가능함을 입증하였다. 또한, KR-WordRank는 언어학적으로 정의되는 단어가 아닌, 주어진 문서집합에서 실제로 자주 사용되는 형태로 단어를 추출하는 특징이 있음을 실험을 통하여 알 수 있었다. 즉, KR-WordRank가 추출하는 대상은 '의미를 지니며 실제로 자주 사용되는 독립적인 단어 혹은 복합형태소'인 것이다. 이를 바탕으로 KR-WorkRank는 다음과 같은 장점을 가질 수 있을 것으로 기대할 수 있다. 첫째, 주어진 문서 집합 내에 포함되어 있는 주제의 종류가 적을수록, 즉 문서 집합의 내용이 이질적이지 않을수록 문서 내부의 단어 및 키워드를 잘 추출할 수 있다. 둘째, 실제로 사용되는 형태의 표현을 단어로 추출하기 때문에 신조어나 오탈자 등의 영향을 적게 받는다. 마지막으로, 학습데이터를 사전에 구축하지 않고도 단어를 인식할 수 있다. 이러한 장점으로 인하여 KR-WordRank는 빅데이터 환경에서 지속적으로 진화하는 언어로 작성된 온라인 문서의 요약 혹은 단어 추출과 같은 응용 연구에 활용될 수 있을 것으로 기대한다.

참고문헌

Berry, M. W. and Castellanos, M. (2007), Survey of Text Mining : *Clustering, Classification, and Retrieval*, Springer, New York, NY, USA.

Chen, S., Xu, Y., and Chang, H. (2011), A simple and effective unsupervised word segmentation approach, *In proceedings of the 25th AAAI Conference on Artificial Intelligence*, San Francisco, CA, USA.

Cho, S. G. and Kim, S. B. (2012), Finding meaningful pattern of key words in IIE Transactions using text mining, *Journal of the Korean Institute of Industrial Engineers*, **38**(1), 67-73.

Fellbaum, C. (2005), WordNet and wordnets, In: Brown, Keith *et al.* (eds.), *Encyclopedia of Language and Linguistics*, Second Edition, Oxford: Elsevier, 665-670.

Feng, H., Chen, K., Deng, X., and Zheng, W. (2004), Accessor variety criteria for Chinese word extraction. *Computational Linguistics*, **30**(1), 75-93.

Harris, Z. S. (1955), From phoneme to morpheme, *Language*, **31**(2), 190-222.

Hotho, A., Nürnberger, A., and Paaß, Gerhard (2005), A brief survey of text mining, *Ldv Forum*, **20**(1), 19-62.

Jin, Z. and Tanaka-Ishii, K. (2006), Unsupervised segmentation of Chinese text by use of branching entropy, *In Proceedings of the COLING/ACL on Main conference poster sessions*, Association for Computational Linguistics.

Jurafsky, D. and Martin, J. H. (2009), *Speech and Language Processing : An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*, Prentice Hall.

Kleinberg, J. M. (1999), Authoritative sources in a hyperlinked environment, *Journal of ACM*, **46**(5), 604-632.

Lawrence, P., Brin, S., Rajeev, M., and Terry, W. (1999), The PageRank citation ranking: Bringing order to the web. Technical Report, Stanford InfoLab.

Lee, D., Yeon, J., Hwang, I., and Lee, S.-G. (2010), KKMA : A tool for utilizing Sejong Corpus based on Relational Database, *Journal of KIISE : Computing Practices and Letters*, **16**(11), 1046-1050.

Lü, X., Zhang, L., and Hu, J. (2004), Statistical substring reduction in linear time, *In proceedings of the 1st International Joint Conference on Natural Language Processing (IJCNLP)*, Hainan Island, China.

Maosong, S. Dayang, S., and Tsou, B. K. (1998), Chinese word segmentation without using lexicon and hand-crafted training data, *In proceedings of the 17th International Conference on Computational Linguistics (COLING)*, Stroudsburg, PA, USA.

McKinsey Global Institute (2011), Big Data : The Next Frontier for Innovation, Competition, and Productivity.

Mihalcea, R. and Tarau, P. (2004), TextRank : Bringing order into texts, *In proceedings of 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Barcelona, Spain.

Mochihashi, D. Yamada T. and Ueda N. (2009), Bayesian unsupervised word segmentation with nested Pitman-Yor language modeling, *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*.

Petrović, S., Šnajder J., and Dalbelo B. (2010), Extending lexical association measures for collocation extraction, **24**(2), 383-394.

Porter, M. F. (1980), An algorithm for suffix stripping, *Program*, **14**(3), 130-137.

Willett, P. (2006), The Porter stemming algorithm : then and now, *Program : Electronic Library and Information Systems*, **40**(3), 219-223.

Zhao, H. and Kit, C. (2007), Incorporating global information into supervised learning for Chinese word segmentation, *In proceedings of the 10th Conference of the Pacific Association for Computational Linguistics (PCALING)*, Melbourne, Australia.