파일명 자동 부착 서비스를 위한 비지도 학습 기반 파일명 추출방법

선주오○, 장영진, 김학수
건국대학교 인공지능학과
qssz1326@konkuk.ac.kr, danyon@konkuk.ac.kr, nlpdrkim@konkuk.ac.kr

For Automatic File Name Attachment Service Unsupervised Learning-based File Name Extraction Method

Ju-oh Sun○, Youngjin Jang, Harksoo Kim
Konkuk University Department of Artificial Intelligence

요 약

심층 학습은 지속적으로 발전하고 있으며, 최근에는 실제 사용자에게 제공되는 애플리케이션까지 확장되고 있다. 특히 자연어처리 분야에서는 대용량 언어 말뭉치를 기반으로 한 언어모델이 등장하면서 사람보다 높은 성능을 보이는 시스템이 개발되었다. 그러나 언어모델은 높은 컴퓨팅 파워를 요구하기 때문에 독립적인 소형 디바이스에서 제공할 수 있는 서비스에 적용하기 힘들다. 예를 들어 스캐너에서 제공할 수 있는 파일명 자동 부착 서비스는 하드웨어의 컴퓨팅 파워가 제한적이기 때문에 언어모델을 적용하기 힘들다. 또한, 활용할 수 있는 공개 데이터가 많지 않기 때문에, 데이터 구축에도 높은 비용이 요구된다. 따라서 본 논문에서는 컴퓨팅 파워에 비교적 독립적이고 학습 데이터가 필요하지 않은 비지도 학습을 활용하여 파일명 자동 부착 서비스를 위한 파일명 추출 방법을 제안한다. 실험은 681건의 문서 OCR 결과에 정답을 부착하여 수행했으며, ROUGE-L 기준 0.3352의 성능을 보였다.

주제어: 비지도 학습, WordRank, 파일명 추출

1. 서론

최근 BERT[1], GPT[2], BART[3]와 같은 사전 학습된 언어모델이 연구됨에 따라 이를 활용한 여러 애플리케이션의 성능이 크게 향상되었다. 그러나 언어모델은 큰 파라미터로 인하여 높은 컴퓨팅 파워를 요구하기 때문에, 소형 디바이스에서 제공할 수 있는 애플리케이션에 적용하기 힘들다는 문제점이 있다. 본 논문에서 다루고자 하는 파일명 자동 부착은 스캐너에서 제공할 수 있는 애플리케이션으로, 스캔된 문서에 자동으로 부착되는 파일명 (‘SCAN001’, ‘SCAN002’)이 아닌 문서 내용을 유추할 수 있는 파일명으로 자동 부착하는 작업을 의미한다. 파일명 자동 부착은 스캐너의 낮은 컴퓨팅 환경에서도 동작해야 하며, 입력받는 스캔 문서의 도메인이 정해져 있지 않기 때문에 지도 학습을 위한 데이터 구축이 어렵다는 문제점이 있다. 따라서 본 논문에서는 컴퓨팅 파워에 비교적 독립적이고 학습 데이터가 필요하지 않은 비지도 학습 기반의 파일명 추출방법을 제안하고자 한다.

2. 관련 연구

비지도 학습 기반 키워드 추출방법은 대표적으로 TextRank[4]와 WordRank[5]가 있다. TextRank는 Google의 그래프 기반 랭킹 알고리즘인 PageRank[6]에 기반해 문서 내의 키워드 추출을 수행하는 알고리즘이다. PageRank는 그래프의 노드를 웹 페이지, 엣지를 웹 페이지 내의 하이퍼 링크로 정의한다. 노드의 초기 중요도를 1로 설정하고 하단의 식 (1)을 통해 각 노드의 중요도를 계산한다.

$$R(P_i) = \frac{(1-d)}{N} + d \times \sum_{P_j \in In(P_i)} \frac{1}{|Out(P_j)|} R(P_j) \quad (1)$$

$R(P_i)$는 웹페이지 P_i의 중요도이다. $In(P_i)$는 P_i를 가리키는 페이지의 집합이고, $Out(P_j)$는 P_i를 가리키는 페이지 P_j의 out-link의 집합을 의미한다. d는 damping factor로 무작위 타 페이지로 이동할 확률을 의미하며 0.85를 사용한다. N은 모든 웹페이지의 개수다. TextRank는 PageRank의 노드를 단어로 엣지를 co-occurrence로 변경하여 문서 내 단어들의 중요도를 계산하고 이를 정렬하여 Top-N개의 단어를 주요 키워드로 결정하는 알고리즘이다.

WordRank는 HITS(Hyperlink-Induced Topic Search)[7] 알고리즘에 기반한 키워드 추출방식으로, 순위모델을 통해 단어 표현을 학습한다.

3. 파일명 추출

본 논문에서 제안하는 파일명 추출방법의 목적은 파일명으로 사용하기 적합한 구를 스캔 된 문서에서 추출하는 것이다. 하단의 그림 1은 파일명 자동 부착의 실제

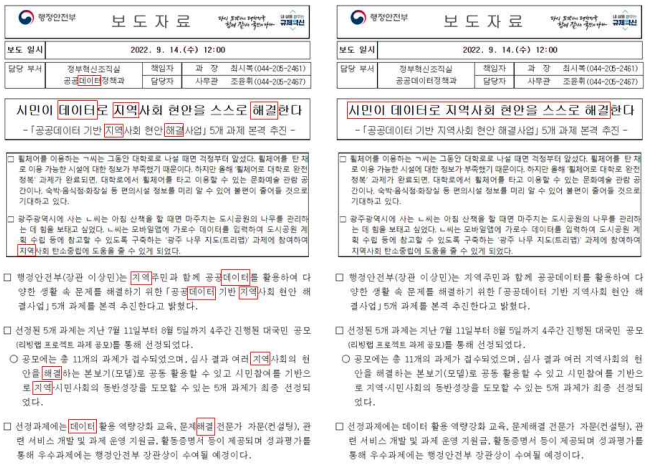

그림 1. (좌) 키워드 추출 결과, (우) 파일명 태깅 정답 예시를 보여준다.

그림 1의 좌측은 WordRank를 기반으로 스캔 된 문서에서 키워드를 추출한 결과이고, 우측은 사람이 직접 문서에 부착한 파일명 정답을 나타낸 그림이다. 그림 1에 따르면 단순히 추출된 키워드는 실제 파일명과 많은 차이가 존재하는 것을 알 수 있다. 따라서 본 논문에서는 추출된 키워드를 실제 파일명과 비슷하도록 구 나 문장 형태로 확장하여 사용자에게 파일명을 제공하고자 한다. 구체적으로 구문 분석 결과를 활용하여 키워드를 구로 확장하는 방법을 제안한다.

3.1 키워드 추출

본 논문에서는 키워드 추출에 KR-WordRank[8]를 사용했다. KR-WordRank는 WordRank를 변형한 키워드 추출방법으로 띄어쓰기 정보를 반영하여 한국어에 더 적합한 키워드 추출 결과를 제공한다. KR-WordRank는 아래 그림 2와 같이 동작을 수행한다.

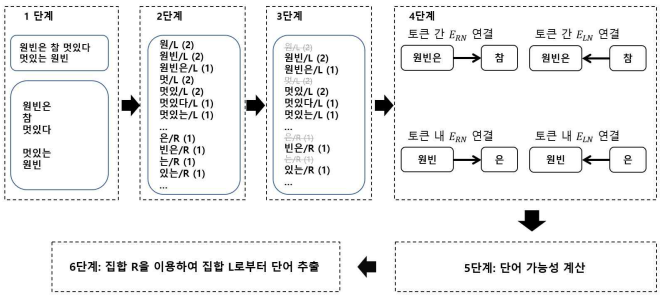

그림 2. KR-WordRank framework

그림 2와 같이 KR-WordRank는 각 문장을 어절 단위로 분할한 뒤, 각 어절을 substring으로 나눈다. 이때, 왼쪽 substring은 L 태그, 오른쪽 substring은 R 태그를 부착한다. 분할된 substring의 빈도수를 계산한 뒤 substring 간 링크를 구성한다. 이후 HITS 알고리즘을 수행해 각 단어의 score를 계산한다. 상위에 rank 된 substring R은 조사나 어미라고 판단하여, 이를 활용해 L에서 조사나 어미를 제거한다. KR-WordRank를 통해 추출된 단어는 파일명에 적합하도록 형태소 분석기[9]를 통해 명사 태그를 갖는 단어만 키워드 후보로 간주했다.

3.2 위치정보 반영

본 논문에서는 OCR 결과에서 제공되는 bounding box의 위치정보를 활용하고자 한다. 정답이 부착된 파일의 정답 등장 위치에 대해 아래 그림 3과 같이 빈도수 분석을 진행했다.

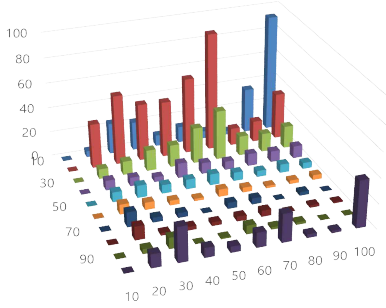

그림 3. 위치에 따른 정답 빈도 분포

그림 3에 따르면 파일명이 문서의 상단 또는 하단에서 빈번하게 등장하는 것을 알 수 있다. 따라서 본 논문에서는 단순히 키워드를 확장하는 방식이 아닌 정답 출현 빈도수를 반영하는 방법을 제안하고자 한다. 하지만 빈도수는 이산적인 분포를 갖고 있기 때문에 그대로 반영할 경우 WordRank 점수와 비교하여 큰 값을 갖게 된다. 이는 파일명 추출 결과가 위치정보에 크게 종속되는 문제점을 야기할 수 있으므로 Convolve Smoothing과 Sigmoid 함수를 이용하여 0~1 사이의 값을 갖도록 변환했다. 하단의 그림 4는 정답 출현 빈도수에 Convolve Smoothing을 적용한 그래프를 보여준다.

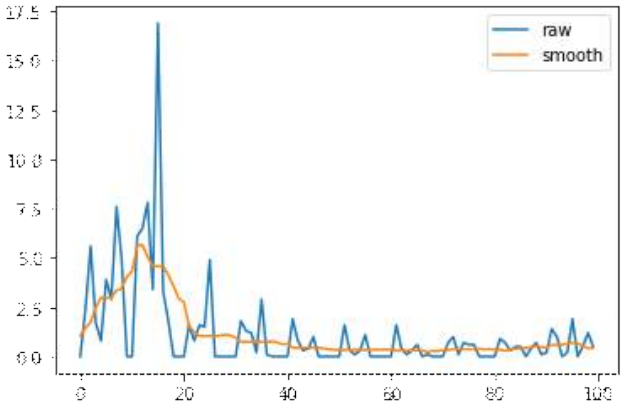

그림 4. Convolve Smoothing 적용 결과

그림 4에서 raw는 실제 정답 출현 분포 그래프를 의미하고, smooth는 실제 정답 출현 분포에 Convolve Smoothing을 적용한 결과를 의미한다. 그림 4에서 알 수

있듯, Smoothing 방법을 적용해 정답 출현 빈도의 편차가 줄어든 것을 알 수 있다. 이후, Sigmoid 함수를 통해 Smoothing을 적용한 정답 출현 빈도수를 0~1사이 값($Position\ Weight$)으로 변환한다. 계산된 $Position\ Weight$는 아래의 과정을 통해 파일명 후보 점수 계산에 사용된다.

3.3 피지배소 확장

3절에서 언급했듯, 본 논문에서는 파일명으로 사용하기 적합한 구를 추출하기 위해 구문분석 결과를 활용하고자 한다. 본 논문에서는 KR-WordRank 결과 상위 3개의 단어를 문서에 대한 키워드로 간주하며, 각 키워드가 존재하는 문장에 대해 구문분석기[10]를 활용해 구문분석을 진행했다. 추출된 키워드를 기준으로 피지배소 확장을 진행하고, 확장된 단어 기준으로 피지배소 확장을 반복적으로 진행해 정답 후보 구를 추출했다. 피지배소가 존재하지 않는 경우 해당 키워드를 포함하는 문장 전체를 후보 구로 선택했다. 이에 대한 예시는 아래의 그림 5와 같다.

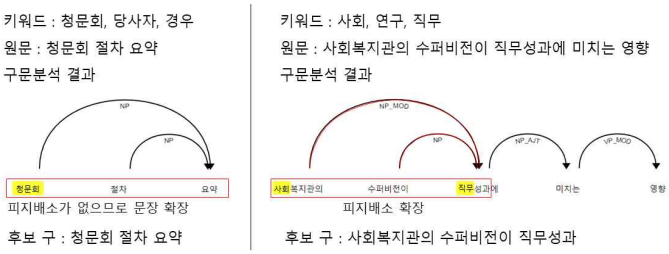

그림 5. 추출 예시 (좌) 문장 확장 (우) 피지배소 확장

추출된 후보 구 중 제목에 적합한 후보 구를 선택하기 위해 각 후보 구에 대한 Scoring을 진행했다. 각 후보 구의 점수 S를 구하는 수식은 아래의 식 (2)와 같다.

$$S(phrase) = (\frac{1}{N}\sum_{1}^{N} w_i) * Position\ \ Weight \qquad (2)$$

위 수식에서 w_i는 $phrase$에 포함된 키워드이다. $Position\ Weight$는 3.2절에서 언급한 bounding box에 따른 $phrase$의 위치 점수를 의미한다. N은 $phrase$에 포함된 키워드의 개수를 의미한다. 부여된 점수에 따라 후보 구는 정렬되며, 실험적으로 결정된 threshold에 의해 최종 파일명을 결정한다.

4. 실험 및 평가

4.1 실험 준비

본 논문은 수동 정답을 태깅한 681건의 문서를 활용하여 실험을 진행하였다. 각 문서는 제안서, 논문, 특허문서, 공문 등으로 구성되어 있다.

4.2 실험 결과

실험에 사용된 성능 지표는 ROUGE-L을 사용했다. 하단의 표 1은 구 점수 계산 방식에 따른 성능 변화를 보여준다. 확장된 모든 구를 최종 파일명으로 간주하지 않고, 구 점수가 일정 threshold 이상인 경우에만 최종 파일명으로 결정했다.

표 1. Global threshold 기준 평가

Scoring	ROUGE-L
Sum	0.2637
Average	**0.3352**

위의 표 1에서 Sum은 구에 포함된 키워드 중요도의 합에 Position Weight를 곱해 구의 점수를 계산하는 방식이다. Average는 식 (2)를 통해 구의 점수를 계산한 방식이다. 표 1의 Sum과 Average의 성능 비교를 통해 파일명 추출 작업에는 추출된 구에 다양한 키워드가 등장하는 것보다 높은 중요도를 갖는 키워드가 등장하는 것이 더 중요한 것을 알 수 있다.

아래의 그림 6은 제안 모델을 통해 실제 추출된 파일명 예시를 보여준다.

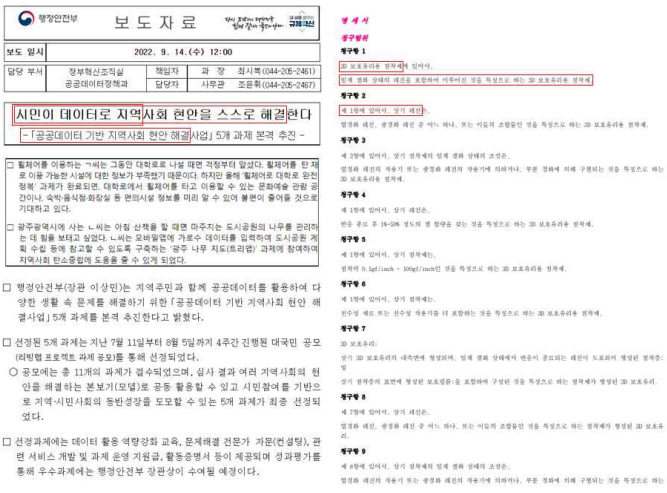

그림 6. Average + 문장 확장 실제 결과 예시

그림 6의 왼쪽 예시는 파일명을 위한 구 추출이 잘 수행되는 것을 보여준다. 그러나, 그림 6의 오른쪽 예시의 경우 파일명이 '명세서' 라는 단어로 부착되었으나 추출되지 않은 것을 볼 수 있다. 키워드 추출은 문서의 내용에 기반해 수행되기 때문에, '명세서' 와 같이 문서의 내용과 관련이 적은 단어에 대해서는 추출에 어려움이 있다는 문제점이 있다.

5. 결론 및 향후 연구

본 논문에서는 파일명 자동 부착 서비스를 위한 파일명 추출방법을 제안했다. 제안 방법은 낮은 컴퓨팅 파워 환경에서도 동작할 수 있는 비지도 학습 기반의 방법을 적용했다. 구문 분석 결과를 이용하여 단순 키워드의 나열 형태가 아닌, 구나 문장 형태의 결과를 제공하는 것을 확인할 수 있었으며, OCR에서 제공하는 구조적인 위치정보를 반영하는 방법을 제안했다. 실험을 통해 ROUGE-L 기준 0.3352의 성능을 보였으나 4.2 절에서 언

급한, 파일명이 본문의 내용과 거리가 먼 경우를 올바르게 추출하지 못한다는 문제점이 있었다. 또한, 최종 파일명 추출에 사용되는 threshold 값이 부착된 정답과의 비교를 통해 실험적으로 결정되는 점에서 한계점이 존재한다. 따라서 향후 연구로, threshold 값을 자동으로 결정하는 방법과 사전 등을 이용하여 본문과 관련이 적은 파일명을 추출하는 방법에 관해 연구하고자 한다.

감사의 글

본 연구는 HP Printing Korea 산학연구용역 과제의 지원을 받아 수행되었음

참고문헌

[1] Devlin, Jacob. Chang, Ming-Wei. Lee, Kenton. Toutanova, Kristina. “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding”. arXiv preprint arXiv:1810.04805, 2018.

[2] Radford, Alee. Wu, Jeffrey. Child, Rewon. Luan, David. Amodei, Dario. and Sutskever, Ilya. “Language models are unsupervised multitask learners”. OpenAI blog, 1(8), 9.2019.

[3] Lewis, Mike, Yinhan, Liu. Goyal, Naman. Ghazvininejad, Marjan. Mohamed, Abdelrahman. Levy, Omer. Stoyanov, Ves and Zettlemoyer Luke. "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension". arXiv preprint arXiv:1910.13461 2019.

[4] Mihalcea, Rada, and Paul Tarau. "Textrank: Bringing order into text". In Proceedings of the 2004 conference on empirical methods in natural language processing, pp. 404-411. 2004.

[5] Ji, Shihao. Hyokun Yun. Yanardag, Pinar. Matsushima, Shin and Vishwanathan, S. V. N. "WordRank: Learning Word Embeddings via Robust Ranking". In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 658-668. 2016.

[6] Page, L., Brin, S., Motwani, R., and Winograd, T. “The PageRank citation ranking: Bringing order to the web”. Stanford InfoLab. 1999.

[7] J, Kleinberg. "Authoritative sources in a hyperlinked environment." Journal of the ACM (JACM) 46.5, pp. 604-632.1999.

[8] Kim, Hyun-joong, Sungzoon Cho, and Pilsung Kang. "KR-WordRank: An unsupervised Korean word extraction method based on WordRank." Journal of Korean Institute of Industrial Engineers 40.1. pp 18-33. 2014.

[9] 최맹식, 김학수. "기계학습에 기반한 한국어 미등록 형태소 인식 및 품사 태깅." 정보처리학회논문지, 제 18-B 권 1, pp. 45-50. 2011.

[10] 최맹식, 정석원, 김학수. CRFs를 이용한 의존 구조 분석 및 의존 관계명 부착. 정보과학회 논문지: 소프트웨어 및 응용. 41(4), pp.302-8. 2014.