JKIICE Journal of the Korea Institute of Information and Communication Engineering

한국정보통신학회논문지 Vol. 26, No. 4: 495~502, Apr. 2022

WCTT: HTML 문서 정형화 기반 웹 크롤링 시스템

김진환[1] · 김은경[2*]

WCTT: Web Crawling System based on HTML Document Formalization

Jin-Hwan Kim[1] · Eun-Gyung Kim[2*]

[1]Graduate Student, Department of Computer Science & Engineering, Korea University of Technology and Education, Cheonan, 31253 Korea
[2*]Professor, School of Computer Science & Engineering, Korea University of Technology and Education, Cheonan, 31253 Korea

요 약

오늘날 웹상의 본문 수집에 주로 이용되는 웹 크롤러는 연구자가 직접 HTML 문서의 태그와 스타일을 분석한 후 수집 채널마다 다른 수집 로직을 구현해야 하므로 유지 관리 및 확장이 어렵다. 이러한 문제점을 해결하려면 웹 크롤러는 구조가 서로 다른 HTML 문서를 동일한 구조로 정형화하여 본문을 수집할 수 있어야 한다. 따라서 본 논문에서는 태그 경로 및 텍스트 출현 빈도를 기반으로 HTML 문서를 정형화하여 하나의 수집 로직으로 본문을 수집하는 웹 크롤링 시스템인 WCTT(Web Crawling system based on Tag path and Text appearance frequency)를 설계 및 구현하였다. WCTT는 모든 수집 채널에서 동일한 로직으로 본문을 수집하므로 유지 관리 및 수집 채널의 확장이 용이하다. 또한, 키워드 네트워크 분석 등을 위해 불용어를 제거하고 명사만 추출하는 전처리 기능도 제공한다.

ABSTRACT

Web crawler, which is mainly used to collect text on the web today, is difficult to maintain and expand because researchers must implement different collection logic by collection channel after analyzing tags and styles of HTML documents. To solve this problem, the web crawler should be able to collect text by formalizing HTML documents to the same structure. In this paper, we designed and implemented WCTT(Web Crawling system based on Tag path and Text appearance frequency), a web crawling system that collects text with a single collection logic by formalizing HTML documents based on tag path and text appearance frequency. Because WCTT collects texts with the same logic for all collection channels, it is easy to maintain and expand the collection channel. In addition, it provides the preprocessing function that removes stopwords and extracts only nouns for keyword network analysis and so on.

키워드 : 웹 크롤링, HTML 문서 정형화, 텍스트 빈도 분석, 태그 경로 분석

Keywords : Web Crawling, HTML Document Formalization, Text Frequency Analysis, Tag Path Analysis

Received 6 December 2021, **Revised** 29 December 2021, **Accepted** 31 December 2021

*** Corresponding Author** Eun-Gyung Kim(E-mail: egkim@koreatech.ac.kr, Tel:+82-41-560-1350)
Professor, School of Computer Science & Engineering, Korea University of Technology and Education, Choenan, 31253 Korea

Open Access http://doi.org/10.6109/jkiice.2022.26.4.495 print ISSN: 2234-4772 online ISSN: 2288-4165

Ⅰ. 서 론

최근 빅데이터 분석이 활발해지면서 SNS나 뉴스, 블로그 등의 다양한 웹 사이트에서 본문 수집을 위해 웹 크롤러가 이용되고 있다[1, 2]. 웹 크롤러는 수집 대상 웹 서버에서 웹 페이지의 본문을 수집하는 과정을 자동화한다[3]. 웹 사이트의 종류가 다양해지고 텍스트의 양이 끊임없이 증가하고 있는 오늘날, 웹상에서 신속하게 본문을 수집하기 위해서는 웹 크롤러의 활용은 필수적이라고 할 수 있다.

지금까지 주로 이용되고 있는 웹 크롤러는 웹 페이지 구조를 기반으로 본문을 수집한다[4, 5]. 이때 웹 크롤러가 본문을 수집하기 위해서는 웹 크롤러 내부에 본문 수집 로직이 구현되어야 하며, 수집 로직은 태그와 스타일 속성을 이용하여 HTML 문서에서 본문을 수집한다. 그러나 이러한 수집 방법을 이용하는 웹 크롤러는 연구자가 직접 수집 채널마다 일일이 HTML의 태그와 스타일을 분석해야 하므로 웹 페이지의 템플릿이 다양한 수집 채널에서의 수집 성능이 떨어질 수 있으며, 수집 채널이 변경되거나 추가될 때 수집 로직을 수정하거나 추가로 개발해야 하므로 웹 크롤러의 유지 관리와 확장이 어렵다는 문제점이 존재한다. 이러한 문제점은 근본적으로 수집 채널마다 상이한 웹 페이지의 구조에 맞춰 웹 크롤러의 수집 로직이 구현되기 때문에 발생하며, 웹 크롤러가 수집한 HTML 문서를 모두 동일한 구조의 정형 데이터로 변환할 수 있다면 하나의 수집 로직만으로도 HTML 문서에서 본문을 수집할 수 있을 것이다.

웹 크롤러가 서로 다른 HTML 문서를 정형화하기 위해 이용할 수 있는 방법으로는 단어/링크 밀도를 이용한 방법[6, 7], 웹 페이지의 시각적 특징을 이용한 방법[8, 9], 텍스트의 출현 순서를 이용한 시퀀스 레이블링(Sequence Labeling) 기반의 방법[10] 등이 있으나, 이 방법들은 수집 채널마다 본문 수집의 정확도가 크게 차이가 난다는 단점이 있다. 이러한 단점을 해결하기 위해 본 연구의 이전 연구[11]에서는 DOM 트리(Document Object Model Tree) 상에서 추출한 텍스트 노드의 태그 경로와 텍스트 출현 빈도를 분석하여 HTML 문서를 정형화하는 방법을 제안하였으며, 다양한 수집 채널에서 높은 정확도로 본문을 수집하는 것을 증명하였다.

따라서 본 논문에서는 수집 채널과 무관하게 동일한 로직으로 본문 수집이 가능하도록 태그 경로 및 텍스트 출현 빈도를 분석하여 HTML 문서를 정형화한 후 본문을 수집하는 웹 크롤링 시스템인 WCTT(Web Crawling based on Tag path and Text appearance frequency)를 구현하였다. WCTT는 수집 채널이 변경되거나 추가되어도 본문 수집 로직을 수정하지 않아도 되므로 유지 관리 및 확장이 용이하다. 또한 최근 많이 이용되고 있는 빅데이터 분석 방법 중 하나인 키워드 네트워크 분석[12] 등에 바로 활용할 수 있도록 수집된 본문에서 불용어를 제거하고 명사를 추출하는 전처리 기능도 구현하였다.

본 논문의 2장에서는 웹 크롤러와 관련된 최근 연구에 대해 소개하고, 3장에서는 본 논문에서 구현한 WCTT의 설계에 대해 자세히 설명하였으며, 4장에서는 WCTT의 구현 결과를 설명하였다.

Ⅱ. 관련 연구

빅데이터 분석을 목적으로 웹상에서 본문을 수집하기 위해 지금까지 주로 이용되고 있는 웹 크롤러는 수집 채널의 웹 페이지 구조에 특화된 수집 로직을 구현해야 한다. 이러한 웹 크롤러[4, 5]는 BeautifulSoup, Selenium, Jsoup과 같은 웹 스크래핑 라이브러리를 이용하여 트위터나 인스타그램, 유튜브, 네이버 블로그 등의 특정 웹 사이트에서 본문을 수집하였지만, 이때의 수집 로직은 웹 페이지 구조가 상이한 모든 수집 채널 각각에 맞게 개별적으로 구현되어야 하므로 웹 페이지 구조가 변경되거나 새로운 수집 채널에 대한 요구가 생겼을 때 로직의 수정이나 추가 개발이 불가피하다.

이러한 문제점은 결국 본문 수집의 효율성을 떨어뜨리는 요인으로 작용하고 있지만, 아직까지 기존의 웹 크롤러를 대체할 수 있는 마땅한 시스템을 찾아보기는 어렵다. 텍스톰(TEXTOM)이나 빅카인즈(Big Kinds)와 같은 빅데이터 분석 솔루션들이 본문 수집을 지원하고는 있지만, 이들 솔루션은 수집한 원본 파일이 제공되지 않고 저장 공간이 제한적이기 때문에 여전히 많은 연구자들이 웹상에서 본문을 수집하기 위해 웹 크롤러를 직접 구현하고 있다. 웹 크롤러 구현에 소모되는 비용을 줄이기 위해 수집 채널에서 제공하는 수집 API를 이용할 수 있지만[13, 14], 이들은 데이터 제공량이 제한되기 때문에 많은 양의 데이터를 수집하기에는 근본적인 한계가 존재한다.

따라서 본 연구에서는 웹 크롤러를 직접 구현하지 않고 다양한 수집 채널에서 누구나 쉽게 본문을 수집할 수 있도록 웹 페이지의 태그 경로 및 텍스트 출현 빈도를 분석하여 본문을 수집하는 웹 크롤링 시스템인 WCTT를 구현하였다.

Ⅲ. WCTT 설계

3.1. HTML 문서 정형화

태그 경로 및 텍스트 출현 빈도 분석 방법[11]을 적용하여 웹 페이지에서 본문을 수집하기 위해서는 먼저 HTML 문서를 정형화하여 텍스트 블록의 집합으로 변환해야 한다. 여기서 텍스트 블록은 어떤 속성을 기준으로 텍스트 노드의 특징을 표현한 것이다. 본 논문에서는 텍스트 블록의 속성을 텍스트(text)와 태그 경로(tp), 그리고 텍스트 출현 빈도 순위(taf_rank)로 구성하고, DOM 트리(Document Object Model Tree) 상에서 나타나는 모든 텍스트 노드의 특징을 추출하여 그림 1과 같이 HTML 문서를 정형화하였다. 이때 태그 경로는 HTML 문서의 DOM 트리를 깊이 우선 탐색으로 순회하여 루트 노드에서 텍스트 노드까지의 경로 상에서 출현하는 태그를 순서대로 나열한 것이고, 텍스트 출현 빈도 순위는 사전에 구성된 TAF 순위 사전(TAF Rank Dictionary)을 이용하여 추출하였다.

TAF 순위 사전은 텍스트별 출현 빈도 순위를 나타내는 Key-Value 쌍의 사전으로서 수집 채널마다 별도로 구성되고, 텍스트의 출현 빈도가 낮을수록 높은 순위를 부여한다. 예를 들어, 어떤 수집 채널에서 수집된 웹 페이지 표본으로부터 3개의 텍스트가 추출되었고 추출된 텍스트의 출현 빈도가 1, 100, 200이라면, 해당 텍스트의 출현 빈도 순위는 차례대로 1, 2, 3이 부여된다. 본 연구에서는 네이버 블로그 5,233개, 중앙일보 3,201개, 인스타그램 8,664개, 트위터 3,313개의 웹 페이지 표본에서 텍스트 출현 빈도를 분석하여 각 수집 채널에 대한 TAF 순위 사전을 구성하였다. 예외적으로 표현이 독특하거나 신조어가 섞인 텍스트처럼 TAF 순위 사전에 순위 정보가 존재하지 않는 텍스트는 1의 출현 빈도 순위를 부여하였다.

no	text	tp	taf_rank
1	# # # 1# 8# 1# 0 0 0# # # # # # # ...	html/head	1
2	0 7 141897737죄송합니다 블로그 서비스 점검 중입니다...	html/head	1
3	코로나 백신 접종 완료 20대 화이자 백신 4주 간격으로 맞은...	html/head	1
4	# 0 0 0# # 4# 70 70 70# b70 70 70# 70 70 ...	html/head	2
5	8140717a# # # # a# ai i 4 999 9995006 999 9995...	html/head	246
6	# # # uu# uu# uuu# uu# uu# uu# uuu# uuu#...	html/head	61
7	블로그 검색	html/body/div/div/div/form/fieldset	246
8	이 블로그에서 검색	html/body/div/div/div/form/fieldset/div	246
9	D뭉아요 시스템	html/body/div/div/div/div	258
10	D공감 아이콘이 더블 하트일 경우 u에클래스 추가	html/body/div/div/div/div/div	238
⋮			
84	코로나 백신 접종 완료 20대 화이자 백신 4주 간격으로 ...	html/body/div/div/div/div/div/div/div/...	1
85	이웃추가	html/body/div/div/div/div/div/div/div/...	249
86	본문 기타 기능	html/body/div/div/div/div/div/div/div/...	240
87	C 상품	html/body/div/div/div/div/div/div/div/...	241
88	19에 대응하기 위해 맞은	html/body/div/div/div/div/div/div/div/...	1
89	화이자 백신	html/body/div/div/div/div/div/div/div/...	8
90	2차까지 접종 완료한 후기를 써보려고 한다	html/body/div/div/div/div/div/div/div/...	1
91	요약해서 말하면	html/body/div/div/div/div/div/div/div/...	1
92	다행히도 나는 크게 이상반응이	html/body/div/div/div/div/div/div/div/...	1
93	나타나지 않았다	html/body/div/div/div/div/div/div/div/...	1
⋮			
294	블로그 마켓 가입 완료 레이어 닫기	html/body/div/div/div/button	246
295	z 0000 00 07 #000000 0403 1 # 92022 0 0 000 ...	html/body/div	246
296	완료되었습니다	html/body/div/div/div/div/div/div	1
297	문의사항은 블로그 마켓 고객센터로 연락주세요	html/body/div/div/div/div/div	246
298	z 0000 00 07 #000000 12 380 0 17 8 1 # 40 ...	html/body	246

Fig. 1 Example of web page converted to structured data

3.2. 시스템 구성

WCTT는 그림 2와 같이 크게 사용자 인터페이스(User Interface)와 웹 크롤러(Web Crawler), 모니터링 서버(Monitoring Server), 관계형 데이터베이스(Relational Database: RDB), 메시징 큐(Messaging Queue: MQ)로 구성된다. 먼저 사용자 인터페이스를 이용하여 데이터를 수집할 수집 채널과 수집 기간 및 수집 키워드 등 수집 조건을 선택할 수 있다.

WCTT의 웹 크롤러는 크게 URL 수집기(Collector)와 HTML 다운로더(Downloader), 본문 수집기(Text Collector), 본문 전처리기(Text Preprocessor)로 구성되며, 수집 채널별로 구성된 데이터 수집 정보(Data Collection Information: DCI)로부터 수집에 필요한 수집 정보를 읽어 URL과 본문을 수집한다. 이때 DCI는 외부에 JSON 포맷 파일로 존재하며, 표 1과 같은 주요 속성으로 구성된다. 이처럼 외부로부터 URL과 본문 수집에 필요한 정보를 가져오므로 태그 및 스타일 속성을 기반으로 본문을 수집하는 방법[4, 5]와는 달리 수집 로직을 수집 채널마다 별도로 구현하지 않아도 된다는 장점이 있다.

3.2.1. URL 수집기

URL 수집기는 DCI의 search_url에 명시된 수집 채널의 검색 페이지에 접근하여 키워드가 포함된 웹 페이지의 URL을 수집한다. 대부분의 웹 사이트에서는 수집 조건을 설정하여 웹 페이지를 검색할 수 있는 검색 페이지

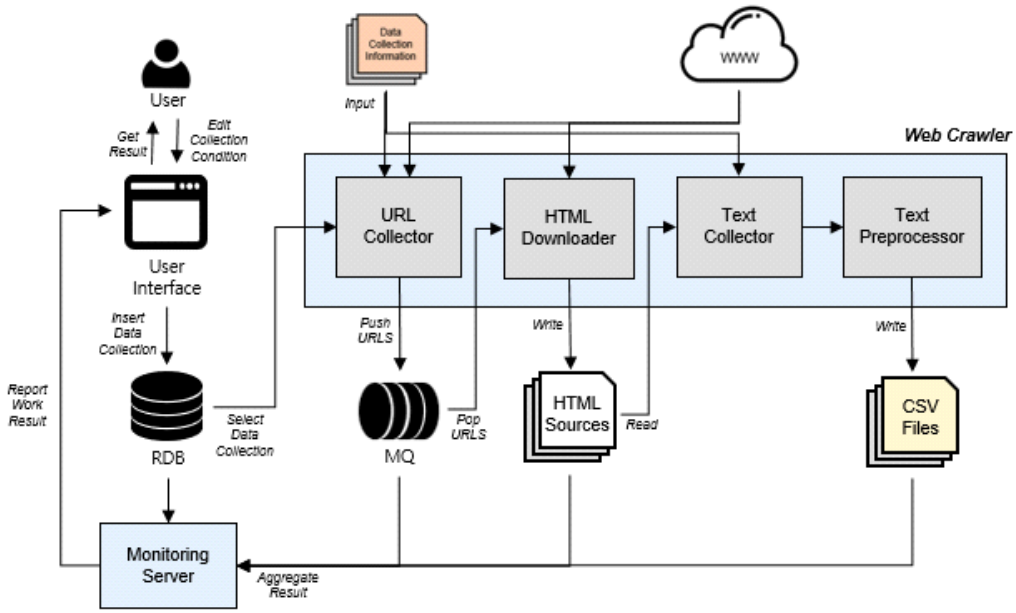


Fig. 2 WCTT structure

를 제공하고 있다. search_url은 검색 페이지에 연결된 URL로, 검색 기간과 키워드를 입력할 수 있는 파라미터를 갖는다. 이 파라미터들을 이용하면 사용자가 원하는 수집 조건에 맞는 웹 페이지의 URL을 검색 페이지에서 찾을 수 있다. 예를 들어, 네이버 블로그에서 search_url은 https://section.blog.naver.com/Search/Post.nhn?startDate=%s&endDate=%s&keyword=%s&pageNo=%s인데, 파라미터인 startDate, endDate, keyword, pageNo을 변경하여 수집 조건에 맞는 웹 페이지의 URL이 포함된 웹 페이지를 불러올 수 있다. 여기서 startDate와 endDate는 수집 기간을, keyword는 수집 키워드를, pageNo은 페이지 번호를 의미한다. 사용자는 WCTT의 사용자 인터페이스에서 수집 채널을 선택하고 수집 키워드와 수집 기간을 입력하면 수집 채널별 search_url에 수집 조건을 입력할 수 있다. URL 수집기는 search_url의 페이지 번호의 파라미터 값을 바꿔 검색 페이지를 갱신하여 URL을 수집한다. 본 연구에서는 예외적으로 인스타그램과 같이 검색 조건에 날짜를 입력할 수 없는 경우에는 Selenium[3]과 같은 브라우저 자동화 도구를 이용하여 스크롤을 조작하여 스크롤바의 제일 아래에 도달하는 순간 검색 페이지를 갱신하도록 하였다.

Table. 1 Main attributes of DCI

Attribute	Description
site_name	Web site name
site_code	Web site code
search_url	Search page URL
html_save_path	Path where HTML documents are saved
csv_save_path	Path where collected text is saved
content_tp_list	Tag path list of main text nodes
taf_boundary_rank	Text appearance frequency boundary rank
taf_rank_dictionary	Text appearance frequency rank per text

URL 수집기가 수집하는 URL의 양은 수집 조건에 따라 방대할 수 있다. 이 경우에 URL 수집기의 모든 작업이 완료된 후 웹 페이지를 내려 받는다면 본문 수집 시간이 전체적으로 지연되어 수집의 효율성이 저하될 수 있다. 이를 방지하기 위해서는 URL과 웹 페이지의 수집을 병행하여 수집 시간을 단축해야 한다. 이때 수집된 URL을 HTML 다운로더에게 효율적으로 전달하기 위해서는 URL 수집기와 HTML 다운로더 사이에 URL 전달을 중개하는 메시징 큐가 필요하다. 본 연구에서는 메시징 큐로써 아파치 카프카를 이용하였다. 아파치 카프카는 발행/구독(subscribe-and-publish) 모델의 분산 메

시징 시스템이자, 분산 환경에서 스트리밍 데이터를 효율적으로 처리하는 분산 스트리밍 플랫폼이다[15]. 아파치 카프카를 이용하면 소비자(consumer)는 원하는 시점에 데이터를 전달받을 수 있으며, 데이터의 오프셋(offset)이 카프카 내부에서 관리되기 때문에 마지막으로 데이터를 가져온 시점에서부터 새롭게 추가된 데이터를 가져오기 용이하다. 이러한 장점은 HTML 다운로더가 URL 수집과 병행하여 웹 페이지를 안정적으로 내려 받을 수 있도록 한다.

3.2.2. 본문 수집기

본문 수집기는 DCI에서 본문에 해당하는 텍스트 노드의 태그 경로 목록인 content_tp_list와 TAF 순위 사전의 위치를 나타내는 taf_rank_dictionary, TAF 경계 순위(TAF Boundary Rank)인 taf_boundary_rank를 참고하여 웹 페이지에서 본문을 수집한다. 본 연구에서는 네이버 블로그와 중앙일보, 동아일보, 인스타그램, 트위터 등의 수집 채널에서 수집한 웹 페이지 표본으로부터 태그 경로와 텍스트 출현 빈도를 분석하여 각 수집 채널에 대한 본문 수집 정보를 미리 구성하였다. 본문 수집기는 먼저 3.1에서 언급한 방법에 따라 웹 페이지를 정형화한 후, content_tp_list에 태그 경로가 존재하고 taf_boundary_rank 이하의 TAF 순위를 갖는 텍스트 블록에서 텍스트를 추출함으로써 본문을 수집한다.

3.2.3. HTML 다운로더

HTML 다운로더는 상시 동작하면서 주기적으로 메시징 큐에 접근하여 URL 수집기에 의해 수집된 URL을 가져온다. 이후 웹 페이지를 내려 받기 전에 메시징 큐의 URL을 집계하여 중복으로 수집된 URL을 제거한다. 본 연구에서는 일반적으로 검색 페이지 대부분의 검색 결과가 썸네일과 제목, 내용으로 구성되고, 구성 요소 각각에 해당 페이지로 이동할 수 있는 하이퍼링크가 하나씩 매칭되어 있다는 점을 감안하여 최소 3번 이상 수집된 URL을 중복으로 간주하였다. HTML 다운로더는 Python의 Selenium을 이용하여 수집된 URL의 웹 페이지에 접근한 후 리소스를 내려 받아 DCI의 html_save_path에 지정된 경로에 HTML 파일을 저장한다.

3.2.4. 모니터링 서버

모니터링 서버는 웹 크롤링 진행 상태를 집계하여 수집된 HTML 문서와 본문의 개수를 수집 채널별로 사용자에게 알려주기 위해, DCI의 html_save_path와 csv_save_path에 저장된 HTML 문서와 본문을 주기적으로 조회한다.

3.2.5. 본문 전처리기

본문 전처리기는 키워드 네트워크 분석 등에 활용할 수 있는 데이터를 생성하기 위해 DCI의 csv_save_path에 저장된 본문에서 불용어를 제거하고 명사를 추출한다. 본 연구에서는 형태소 분석의 정확도와 실행 속도를 고려하여 Python의 형태소 분석 라이브러리인 Mecab[16]를 이용하여 본문 전처리기를 구현하였다.

Ⅳ. WCTT 구현

WCTT의 구현 환경은 표 2와 같다. 사용자 인터페이스는 React.js와 Node.js를, 웹 크롤러는 Python을 이용하여 구현하였으며, Widnows 10 Pro 운영체제에서 시스템을 테스트하였다.

Table. 2 Implementation environment of WCTT

OS	Windows 10 Pro
User Interface	React.js 17.0 / Node.js 12.22
Web Crawler	Python 3.8
Message Queue	Apache Kafka 2.12
Database	SQLite3

WCTT의 모든 컴포넌트가 정상적으로 구동되면 그림 3과 같이 웹 브라우저가 실행된다. 사용자는 그림 3에서 수집 작업명과 키워드를 입력하고, 수집 기간과 수집 채널을 선택한 후 '등록하기' 버튼을 클릭하여 새로운 수집 작업을 추가할 수 있다. 키워드는 '추가' 버튼을 클릭하여 5개까지 추가할 수 있다.

수집 작업이 정상적으로 등록되면 그림 4의 (a)와 같이 수집 작업 목록에서 새로 생성된 작업 목록을 확인할 수 있으며, 사용자는 해당 목록에서 우측의 '시작하기' 버튼을 클릭하여 대기 중인 수집 작업을 실행시킬 수 있다. 작업이 시작되면 그림 4의 (b)와 같이 수집된 HTML 문서와 웹 텍스트의 라인 수가 실시간으로 확인할 수 있다. 수집 채널별 수집 현황을 확인하기 위해 목록에서 확인하고자 하는 작업을 선택하여 그림 5와 같은 수집 작업 상세보기 화면을 호출하여 상세한 진행 상황을 파

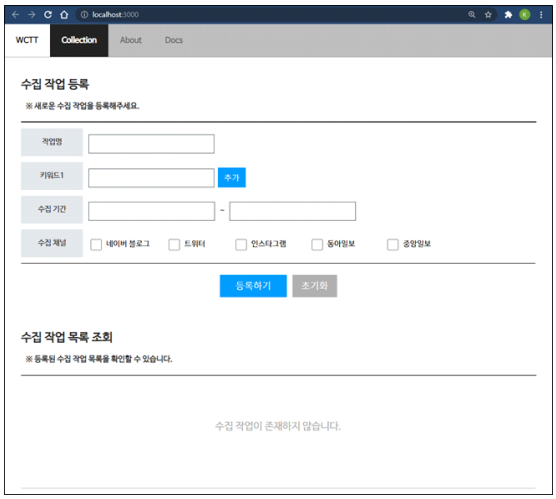
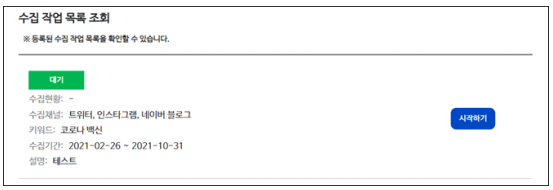
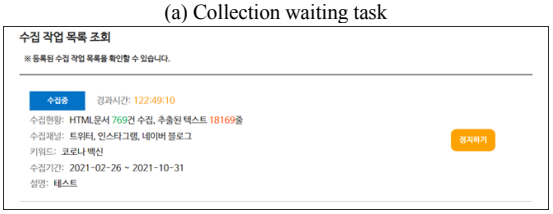


Fig. 3 User interface to register collection task



(a) Collection waiting task



(b) Collection execution task

Fig. 4 User interface to view a list of collection task

악할 수 있다. 또한, 수집된 본문의 전처리를 위해 그림 5의 좌측 하단에 위치한 '본문 전처리' 버튼을 클릭하여 전처리를 위한 사용자 인터페이스를 호출할 수 있다. 사용자는 그림 6의 사용자 인터페이스에서 '불용어 입력' 버튼을 클릭하여 본문에서 제거하고자 하는 텍스트를 등록할 수 있으며, '명사 추출하기' 버튼을 클릭하여 본문에서 불용어와 조사, 수치 등 데이터 분석 시에 중요도가 떨어지는 불필요한 텍스트를 제거하고 명사를 추출하여 그림 7의 하단과 같이 전처리된 본문을 저장할 수 있다.

Fig. 5 User Interface for detailed view of collection task

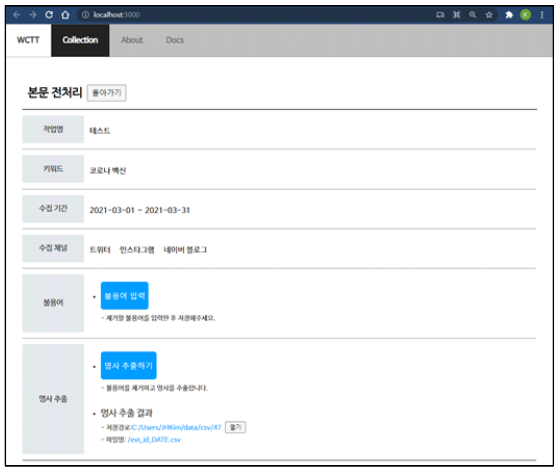


Fig. 6 User interface to preprocess text

Ⅴ. 결 론

본 논문에서는 태그 경로와 텍스트 출현 빈도 분석 방법을 이용하여 웹 페이지를 정형화한 후 본문을 수집하는 웹 크롤링 시스템인 WCTT를 설계 및 구현하였다. WCTT는 태그 및 스타일 속성 기반의 웹 크롤러와는 달리 모든 수집 채널에 대해 동일한 수집 로직으로 본문을

	본문
전처리 전 (원본)	백신 접종 후 2명 사망접종 기피 말라 요양병원에서 지내다가 백신을 맞은 후 숨졌습니다 보건 당국은 두 사람이 숨진 원인이 아스트라제네카 백신과 관계가 있는지 조사하고 있습니다 그러면서 아직 다른 나라에서도 백신 부작용으로 숨진 사례는 확인된 게 없다며 백신 접종을 피하지 말아 달라고 당부했습니다 첫 사망자는 경기도 일산 한 요양병원에 입원했던50대 남성입니다 어제 2일오전 9시 반 접종받았고 시간 경과 후 가슴 통증과 메스꺼움 호흡곤란 증상을 호소했습니다 병원에서 응급처치받았지만 접종 약22시간 만인 오늘 아침 7시 숨졌습니다 두 번째 사망자는 평택의 한 대 남성 환자로 일 시간 뒤 발열과 전신 근육통이 나타났습니다 장시 상태가 호전됐다가 악화해 오늘 시 숨을 거뒀습니다 사망자 두 명 모두 만성질환을 앓는 기저질환자로 예방접종 권고 대상이었습니다 보건당국은 백신과 사망 간 인과성 여부를 확인하기 위해 역학조사를 진행하고 있습니다 이들이 같은 제조번호의 백신을 맞았는지 어떤 증상이 있었고 검사 결과는 어땠는지 사망에 이른 다른 요인이 있는지 종합 조사해 판단하겠다고 설명했습니다 사망자가 발생한 요양병원 두 곳에서 한 달 평균 다섯 명 일곱 명이 숨진 것도 판단의 근거가 될 것으로 보입니다 질병관리청은 백신을 먼저 접종한 나라에서도 접종으로 인한 사망 사례가 확인된 건 아직 없다며 접종을 피하지 말라고 당부했습니다 정은경질병관리청장국민들께서 과도하게 불안감을 가지시고 접종을 피하거나 그러시지 않으셨으면 한다는 말씀드리고 백신 접종 후 급성 중증 알레르기 반응인 아나필락시스 의심 증상을 보인 사람도 3명 확인됐는데 2명은 회복해 귀가했고 1명은 관찰 중입니다 백신 접종 후 2명 사망접종 기피 말라 앵커요양병원에서 지내다가 백신을 맞은 50대와 60대가 숨졌습니다 보건 당국은 두 사람이 숨진 원인이 아스트라제네카 백신과 관계가 있는지 조사하고 있습니다 #코로나 #코로나백신 #한국백신 #백신접종 #백신사망 #백신접종사망 #코로나백신사망 #아스트라제네카 #아스트라제네카백신 #아스트라제네카사망 #백신사망자 #아스트라제네카사망자 #아스트라제네카백신 #코로나 #코로나백신 #한국백신 #백신접종 #백신사망 #백신접종사망 #코로나백신사망 #아스트라제네카 #아스트라제네카백신 #아스트라제네카사망 #백신사망자 #아스트라제네카사망자
전처리 후	백신 접종 사망 접종 기피 요양 병원 백신 보건 당국 원인 아스트라 백신 관계 조사 나라 백신 부작용 사례 백신 접종 당부 사망자 경기도 일산 요양 병원 입원 남성 오전 접종 경과 가슴 통증 호흡 곤란 증상 호소 병원 응급처치 접종 아침 사망자 평택 남성 환자 발열 전신 근육통 상태 호전 악화 사망자 만성 질환 저질 환자 예방 접종 권고 대상 보건 당국 백신 사망 인과성 여부 역학 조사 제조 번호 백신 증상 검사 결과 사망 요인 종합 조사 판단 설명 사망자 발생 요양 병원 평균 다섯 일곱 판단 근거 질병 관리 백신 접종 나라 접종 사망 사례 접종 당부 정은경 질병 관리 청장 국민 과도 불안감 접종 말씀 백신 접종 급성 중증 알레르기 반응 아나필락시스 의심 증상 회복 귀가 관찰 백신 접종 사망 접종 기피 앵커 요양 병원 백신 보건 당국 원인 아스트라 백신 관계 조사 코로나 코로나 백신 한국 백신 백신 접종 백신 사망 백신 접종 사망 코로나 백신 사망 아스트라 아스트라 백신 아스트라 사망 백신 사망자 아스트라 사망자 아스트라 백신 코로나 코로나 백신 한국 백신 백신 접종 백신 사망 백신 접종 사망 코로나 백신 사망 아스트라 아스트라 백신 아스트라 사망 백신 사망자 아스트라 사망자

Fig. 7 Example of preprocessed text

수집하기 때문에 수집 채널의 변경에 대한 유지 관리 및 확장이 용이하다. 또한 수집 채널에서 제공하는 API를 이용하는 웹 크롤러와는 달리 데이터에 제한 없이 본문을 수집할 수 있다는 장점이 있다. 뿐만 아니라, 수집한 본문을 키워드 네트워크 분석 등에 바로 활용할 수 있도록 불용어를 제거하고 명사만 추출하는 전처리 기능도 제공한다. 향후 보다 다양한 본문을 수집할 수 있도록 DBpia 등과 같은 학술 DB를 수집 채널에 추가하여 특정 키워드가 포함된 학술 자료를 수집하고, 키워드와 높은 연관성을 갖는 본문을 추출하는 기능 등을 추가할 계획이다.

ACKNOWLEDGEMENT

This paper was supported by the Education and Research Promotion Program of KOREATECH in 2022.

References

[1] Y. J. Kim, H. S. Kim, and H. S. Kim, "Understanding the Effects of COVID-19 on the Starbucks Perception through Big Data Analytics: A Comparative Study," *Culinary Science & Hospitality Research*, vol. 27, no. 6, pp. 276-279, Jun. 2021.

[2] Y. R. Suh, K. P. Koh, and J. W. Lee, "An analysis of the change in media's reports and attitudes about face masks during the COVID-19 pandemic in South Korea: a study using Big Data latent dirichlet allocation (LDA) topic modelling," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 25, no. 5, pp. 731-740, May. 2021.

[3] D. H. Han and Y. K. Koo, "Design of Action-Based Web Crawler Structural Configuration for Multi-Website Management," *KIISE Transactions on Computing Practices*, vol. 27, no. 2, pp. 98-103, Feb. 2021.

[4] J. H. Lee, "Building an SNS Crawling System Using Python," *Journal of the Korea Industrial Information Systems Research*, vol. 23, no. 5, pp. 61-76, Oct. 2018.

[5] S. Y. Park, J. H. Moon, Y. W. Kim, and H. G. Lee, "Design of Tree Structure Based Hypertext Crawler Using Jsoup," in *Proceedings of Symposium of the Korean Institute of communications and Information Sciences*, vol. 65, no. 1, pp. 896-897, Jan. 2018.

[6] C. Kohlschuer, P. Fankhauser, and W. Nejdl, "Boilerplate detection using shallow text features," in *Proceedings of the third ACM international conference on Web Search and Data Mining (WSDM)*, New York: NY, pp. 441-450, Feb. 2010.

[7] W. M. Song, W. S. Kim, and M. W. Kim, "Contents Extraction from HTML Documents using Text Block Context," *Journal of KISS : Software and Applications*, vol. 40, no. 3, pp. 155-163, 2013.

[8] S. H. Kim and H. J. Kim, "Logistic Regression Ensemble Method for Extracting Significant Information from Social Texts," *KIPS Transactions on Software and Data Engineering*, vol. 6, no. 5, pp. 279-284, May. 2017.

[9] J. H. Mo and J. M. Yu, "Korean Web Content Extraction using Tag Rank Position and Gradient Boosting," *Journal of KIISE*, vol. 44, no. 6, pp. 581-586, Jun. 2017.

[10] J. Leonhardt, A. Anand, and M. Khosla, "Boilerplate Removal using a Neural Sequence Labeling Model," in *Companion Proceedings of the Web Conference 2020 (WWW '20)*, New York: NY, pp. 226-229, 2020.

[11] J. H. Kim and E. G. Kim, "HTML Text Extraction Using Tag Path and Text Appearance Frequency," *Journal of the Korea Institute of Information and Communication Engineering*, vol. 25, no. 12, pp. 1709-1715, Dec. 2021.

[12] W. K. Kim, Y. H. Kim, and J. S. Park, “Digital Literacy Research Trend Analysis Using Keyword Network Analysis - 2011-2015 and 2016-2020 comparative analysis,” *The Korean Journal of Literacy Research*, vol. 12, no. 4, pp. 93-125, 2021.

[13] D. H. Kim, J. W. Koo, and U. M. Kim, "Design and Implementation of Automated Twitter Data Collecting System : Focus on Environmental Data," in *Proceedings of the Korea Information Processing Society Conference*, Online, vol. 27, no. 1, pp. 361-364, 2020.

[14] K. S. Yoon and Y. H. Kim, "Designing and implementing web crawling-based SNS web site," in *Proceedings of the Korean Society of Computer Information Conference*, Busan, vol. 26, no. 1, pp. 21-24, 2018.

[15] W. S. Ryu, “A System Design for Real-Time Monitoring of Patient Waiting Time based on Open-Source Platform,” *Journal of the Korea Institute of Information and Communication Engineering*, vol. 22, no. 4, pp. 575-580, Apr. 2018.

[16] H. S. Kang and J. H. Yang, "Selection of the Optimal Morphological Analyzer for a Korean Word2vec Model," in *Proceedings of the KIPS Conference*, Busan, vol. 25, no. 2, pp. 376-379, 2018.

김은경(Eun-Gyung Kim)

1983년 2월 : 숙명여자대학교 물리학과 졸업
1986년 2월 : 중앙대학교 전자계산학과 석사
1991년 2월 : 중앙대학교 컴퓨터공학과 박사
1992년 3월~ 현재 : 한국기술교육대학교 컴퓨터공학부 교수
※관심분야 : 빅데이터 분석, 딥러닝, 트리즈 등

김진환(Jin-Hwan Kim)

2016년 2월 : 한국기술교육대학교 컴퓨터공학부 졸업(학사)
2019년~현재 : 한국기술교육대학교 컴퓨터공학과 석사과정
※관심분야 : 빅데이터, 텍스트마이닝, 웹 크롤링, 기계학습