|  |  |  |  |
| --- | --- | --- | --- |
| **2016년도 2학기 My Lab연구 최종 보고서**  **-My Lab 연구 Ⅱ 수업-** | | | |
| **지원연구실** |  | **지원자 학번** | **20123417** |
| **교수님** | **이 경 용** | **지원자 이름** | **조 성 룡** |
| **연구실 연락처** |  | **지원자 연락처** | **010-7119-5296** |
| **멘토 대학원생** |  |
| **지원 동기** | **웹의 최신기술을 활용하여 빅데이터를 분석하고 결과를 도출해 내보고 싶었다.** | | |
| **연구 주제** | **1. 웹 최신 기술인 Crawling – Scrapy**  **2. 페이지의 분석 & PageRank Algorythm 활용 - R** | | |
| **연구 배경 및 내용**  **요약** | **1. 웹 최신 기술인 Crawling을 활용하여 국민대학교 홈페이지인** [**www.kookmin.ac.kr**](http://www.kookmin.ac.kr)**에서 모든 페이지를 가져온다.**  **2. 가져온 페이지를 가지고 R을 통해 분석을 한다. Graph와 PageRank를 통해 결과를 도출해 낸다.** | | |
| **예상 소요비용 및 내역** | **-** | | |
| **연구 과정 요약** | **1.**   * **Crawling을 활용하기 위한 Scrapy 구축** * **Scrapy를 통해 페이지 가져오기**   **2.**   * **추출해 낸 Data를 R을 통해 분석 (Graph , PageRank)** | | |
| **연구 활동 내역 요약** | **1. Scrapy를 통한 kook.ac.kr의 연관된 모든 페이지를 저장하고 이에 대한 DB를 만들고 저장.**  **2. DB를 R를 통해 Graph적으로 분석하고 PageRank Algorythm을 적용하여 페이지간 중요도를 도출해 낸다.** | | |
| **연구 결과 요약** | **1. Scrapy를 통해 국민대학교 홈페이지를 긁어온 결과 약 15,000 페이지가 존재하며 이 중 약 300 페이지는 죽은 페이지 인 것으로 확인 되었다.**  **2. 각 페이지간의 상관관계를 분석한 결과 국민대학교 대문 페이지인 kook.ac.kr이 상위권에 위치하고 있었으며 컴퓨터공학부 페이지인 eecs.kookmin.ac.kr는 하위권에 위치하고 있었다.** | | |
| **활용 및 기대효과** | **1. Crawling을 통해 어떤 페이지가 죽었는지 확인 할 수 있으며 이를 활용해 해당 페이지에 대한 유지보수가 가능 할 것으로 기대된다.**  **2. PageRank를 통해 중요도를 표시하고 특정 Contents를 중요한 페이지에 담는 등의 효과를 기대 할 수 있을 것이다.** | | |
| **느낀 점** | **웹의 최신기술인 Crawling을 활용하기에 앞서 Python을 이용한 Beautiful soup을 써보았으나 동적인 페이지에 대한 활용이 불가능 했다. 후에 Scrapy라는 프레임워크를 이용하여 모든 링크에 대해 동적으로 Crawling이 가능했다. 또 평소에는 듣기만 했던 R을 이용해 Data를 분석했다. 이를 통해 웹에 대한 기술과 Data를 다루는 기법 등에 대해 숙지하게 되었으며 실제 웹 페이지에 이용되고 있던 PageRank Algorythm을 통해 검색엔진이 구동되는 방식에 대해 알게 되었다.** | | |

※페이지 제한 없음

**1장. 연구 배경**

**웹 기술과 데이터 분석을 위한 것을 주제로 삼고자 했다. 이에 웹 기술인 Crawling을 이용하여 특정 데이터를 추출하고 추출된 데이터를 기반으로 분석하는 했다.**

**학부 교과 과정에는 없는 다양한 기술과 언어를 사용하면서 많을 것을 느끼고 배우고 싶었다.**

**2장. 연구내용**

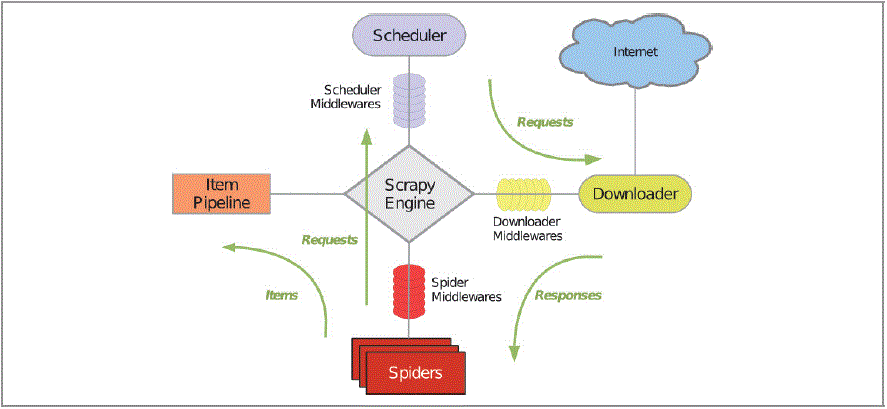
Kookmin.ac.kr와 연관된 모든 페이지를 추출해내고 이에 대한 분석을 통해 그래프를 그리고 rank를 매기는 등의 다양한 활동을 하였다. 후에 추출해낸 페이지의 title등을 활용 하여 각 페이지의 단어를 통한 연관관계를 분석하는 등의 효과도 기대 할 수 있을 것이다.

ㅇ Crawler 선택

크롤러는 많은 종류가 존재하나 최근에 가장 주목받고 있는 Scrapy를 사용하기로 결정했다. Scrapy는 프레임워크 이기에 확장 가능한 기반 코드를 제공한다. 연구 초기 Beatiful soup을 사용을 했지만 라이브러리라는 한계로 HTML을 파싱하는 기능 외 다른 기능은 없어 연구에 적합하지 않다고 생각했다.

ㅇ Scrapy

Scrapy는 프레임워크로서 일련의 구조를 가지고 있다. 그 구조는 다음과 같다.



<Scrapy의 구조>

-스케쥴러: Scrapy 엔진의 수집에 관련된 정책사항을 설정하는 역할을 담당한다.

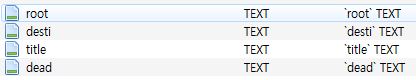
-아이템 & 파이프라인 : 수집하려는 데이터의 입출력을 담당한다. 수집하려는 항목을 아이템으로 정의하고 수집한 데이터의 형태를 파일 혹은 DBMS로 직접 입력이 가능하도록 설정 할 수도 있다.

-스파이더 : 수집하는 데이터를 크롤링하는 역할을 한다. 설정에 따라 데이터를 아이템 형태로 파이프라인에 전송하기도 한다. 이 연구에서는 파이프라인을 통한 전송은 하지 않고 바로 DB로 만들어 내는 과정을 거친다.

-다운로더 : HTTP , FTP 프로토콜을 해석하여 웹에 있는 데이터를 다운로드 하는 역할을 담당한다.

ㅇData

다음은 실제 구축된 Scrapy를 통해 추출해낸 DB의 TABLE을 나타낸다.



<실제 사용된 DB의 table 구조>

Root – Scrapy가 request를 보낸 페이지이다.

Desti – request를 받는 페이지이다.

Title – Desti의 쓰여진 title 태그를 기반으로 파싱을 한 값이다.

Dead – requet가 어떠한 이유로 불가능할 경우 죽은 page로 간주한다.

ㅇR

주어진 DB를 기준으로 R studion를 활용해서 시각적인 분석을 한다.시각적인 분석에는 igraph를 활용한 노드간 그래프의 연결성을 보여준다. 또 시각적인 분석 외에도 각 페이지의 중요도를 나타내는 PageRank Alogrythm을 통해 각 페이지의 rank를 구하고 내림차순으로 정렬하여 결과를 도출해낸다.

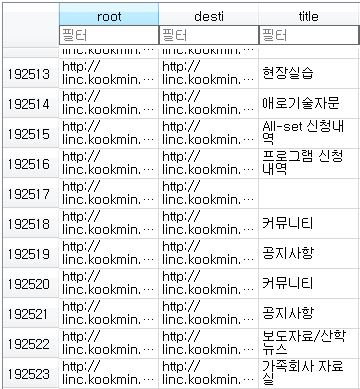
ㅇPageRank Alogorythm

이 알고리즘은 월드 와이드 웹과 같은 하이퍼링크 구조를 가지는 문서에 상대적 중요도에 따라 가중치를 부여하는 방법이다. 서로간의 인용과 참조 (여기서는 root -> desti를 기준으로)로 연결된 임의의 묶음에 적용 할 수 있다. 페이지 랭크는 더 중요한 페이지는 더 많은 다른 사이트로부터 링크를 받는다는 관찰에 기초하고 있다. 예를 들어 페이지 A가 페이지 B,C,D 로 또한 페이지 랭크에서는 랜덤 서퍼(Random Surfer)라는 페이지를 임의로 방문하며 탐색하는 모델을 가정한다. 이 모델에서는 위 예의 A페이지를 방문한 서퍼는 A페이지를 보고 만족하여 탐색을 중단하거나, 혹은 A페이지에서 만족하지 못하여 다른 페이지를 방문할 것이다. 이러한 확률을 a라 한다면, B페이지는 a\*(1/3)만큼 페이지 랭크를 받게 된다.

페이지 랭크는 이와 같은 방법을 통해 페이지간 페이지 랭크 값을 주고 받는 것을 반복하다 보면, 전체 웹 페이지가 특정한 페이지 랭크 값을 수렴한다는 사실을 통해 각 페이지의 최종 페이지 랭크를 계산 할 수 있게 된다.

**3장. 연구결과**

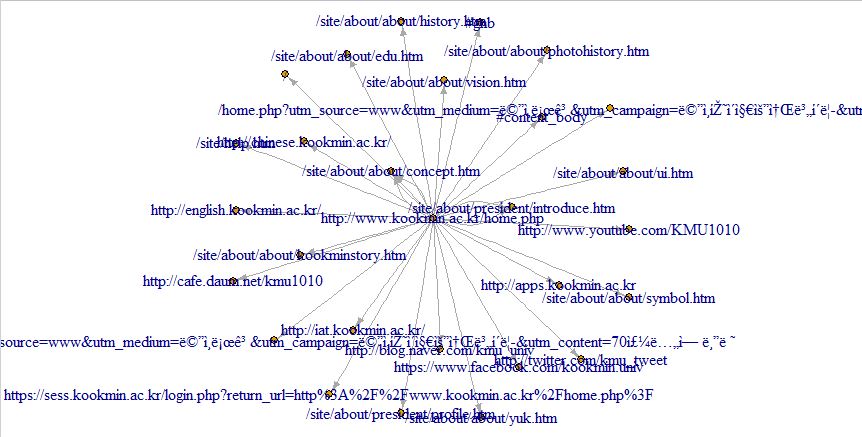
ㅇDATA

****

<실제 추출된 Data의 일부>

약 15,000의 페이지를 Craling 했다. 15,000에 대한 데이터는 약 190,000개가 추출 되었으며 이 중 죽은 페이지 (Dead에 체크가 된 페이지)는 약300개로 추정된다.

추출된 DATA를 기반으로 R을 이용해 graph를 그려보고자 했다. DB의 root 와 desti를 edge list로 활용하여 30개의 DATA의 graph를 그리면 다음과 같다. (컴퓨터 성능상의 문제로 많은 data에 대한 그래프는 그릴 수가 없었다.)

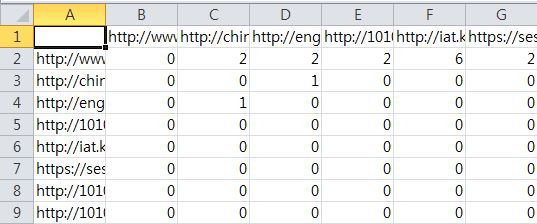


<data 30개 대한 Graph>

Graph를 그리는 과정에서 data가 넘어 올 때 한글이 깨지는 경우가 발생한다. 이는 UTF-8인코딩을 거치지 않는 것으로 보여진다.

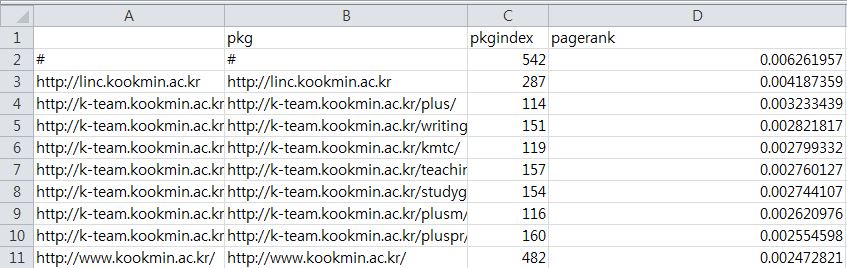
아래의 그림은 root , detsi의 edge list를 adjacency matrix(이하 인접행렬)로 만든 결과이다. 컴퓨터의 Memory제한으로 인해 일부분의 Data만을 인접행렬로 만들었다. 10,000개의 data에 대한 결과로 약

3000 x 3000의 행렬을 도출해 낼 수 있었다.



<10,000개의 data에 대한 인접행렬 중 일부분>

만들어진 인접행렬의 값을 기준으로 각 값에 대한 PageRank Algorythm을 적용한 결과는 다음과 같다.



<추출된 Data 중 일부분>

위의 결과에서 국민대학교의 대문 페이지인 kookmin.ac.kr 11위라는 순위를 보이고 있음을 확인 할 수 있다. 3 ~ 10위를 차지하고 있는 k - team페이지의 결과로 보아 많은 페이지에서 링크가 걸려있고 학교에서 밀어주고 홍보하고자 하는 페이지로 판단이 된다.

**4장. 결론**

이번 연구를 통해 웹과 데이터 분석적인 면에서 많은 지식을 쌓을 수 있었다. 웹 최신 기술 중 하나인 Crawling을 통해 Data를 추출하고 R을 통해 분석했다. 이러한 분석의 결과로서 페이지의 생사유무에 대해 알 수 있었고 어떤 페이지가 많은 링크가 걸려있는지 PageRank를 통해 알 수 있었다. 추 후 기대되는 결과로는 많은 링크가 걸린 페이지를 통해 중요도가 높은 페이지임을 알 수 있고 이 페이지에 Contents를 추가하여 광고, 홍보의 목적으로 쓸 수 있을 것으로 보인다. 데이터를 보면 대문페이지인 kookmin.ac.kr이 예상외로 11위권에 미치고 linc.kookmin.ac.kr이 최상위권에 있음을 알 수 있다. 이러한 결과는 각 페이지에서 linc에 대한 링크가 많이 걸려있다는 것을 의미한다. 즉, 국민대학교는 linc사업을 많이 홍보하고 있다고 생각 할 수 있겠다.

데이터를 Crawling 하면서 링크에 대한 title도 같이 추출하였다. 이 title을 통해 페이지간의 연관성에 대해 좀 더 높은 정확도를 부여 할 수 있으며 각 title을 묶어 여러 개의 군집을 생성 할 수 있을 것이다. 시간의 제약으로 이러한 연구까지 하지 못한 것은 아쉬운 점으로 남는다. 후에 기회가 있다면 이 연구에 쓰인 Scrapy를 보안하고 다양한 방면에서의 분석을 해보고 싶다.. 많은 배움과 아쉬움이 공존하는 연구였다.