효율적인 웹 마이닝 시스템의 설계 및 구현

김형욱 최익규 김민구

Design and Implementation for the Effective Web Mining System

HyoungUk-Kim IkKou-Choi Minkoo-Kim

{wizard, ikchoi, minkoo}@ajou.ac.kr

요약

효율적인 웹 마이닝을 위해서는 방대한 인터넷 공간에서 사용자가 원하는 정보를 찾아 내고, 이들 중 보다 유용하다고 판단되어진 자료를 선별적으로 제시할 수 있어야 한다. 본 논문에서는 웹 컨텐츠 분석과 HTML 문서들 사이의 링크 연결의 패턴 분석을 기반으로 하는 웹 구조 분석 방법들을 검토하고, 웹 검색 시스템을 구현하여 결과를 분석하였다. 이를 위해 웹 문서의 내용을 인덱싱한 뒤 질의와의 관련성의 확률을 구하는 랭귀지 검색 모델에 링크 구조 분석을 이용한 순위 알고리즘을 사용하여 좋은 결과를 얻고자 하였다. 또한 기존의 링크 관련 알고리즘에서 알려진 문제점을 해결하기 위한 몇 가지 테크닉을 사용하였다

Key words : 정보 검색, 웹 검색, 검색 모델, 인공 지능

1. 서론

웹 환경에서의 정보 검색은 이전의 검색 환경과 비교해 상대적으로 방대한 규모를 갖는 데이터로부터 사용자가 제시하는 적은 수의 질의어들을 이용하여, 효과적인 검색을 수행해야 하는 어려움을 갖는다. 일반적으로 사용자들은 자신이 이용하고 있는 검색 시스템의 구조와 수행 방식을 알고 있지 못하기 때문에, 원하는 문서를 정확하게 가져올 수 있는 질의어들을 사용하는 대신 평균적으로 3~4개 이내의 관련된 주제를 포괄하는 질의어들을 사용하는 경향을 가지고 있다. 사용자가 제시한 부정확하거나 광의의 질의어들에 의존하여 검색 엔진들은 아주 작은 수거나, 사용자가 결과를 살펴보기에 지나치게 많은 수의 문서들을 검색 결과로 가져오게 된다. 이러한 문제는 전체 문서 데이터의 크기가 큰 웹 검색에서의 경우 좋지 못한 검색 효율을 야기하는 원인이 된다. 따라서 웹 환경에서의 검색 엔진은 질의어와 관련된 문서들을 찾아낼 뿐만 아니라, 구해진 문서들 중에서 보다 질의에 관련이 있고, 귀중한 정보를 갖는 문서들의 우선순위에 따라 사용자에게 제시해야만 하며, 이를 위해 어떻게 문서의 가치를 판단할 것인가가 중요한 의미를 갖게 된다.

웹 환경을 구성하고 있는 문서들은 HTML (HyperText Markup Language) 문법의 형식을 따르며, 다른 문서들과 방향성을 갖는 링크로 연결되어 있는 하이퍼텍스트(HyperText)로서의 특성을 갖는다. 웹 문서를 만드는 저작자는 자신이 만든 웹 문서로부터 다른 웹 문서들을 연결하게 하는 링크를 삽입하게 되며, 일반적으로 웹 문서의 내용과 관련하여 중요한 정보를 가지고 있다고 판단되거나, 자신의 웹 문서와 연관성을 갖는 웹 문서들을 링크가 가리키고 있는 대상으로 선택하게 된다. 따라서 많은 수의 웹 문서들로 부터의 링크를 가지고 있는 웹 문서는 적은 수의 웹 문서들로부터 참조되고 있는 웹 문서보다 상대적으로 높은 중요도를 갖는다고 가정할 수 있다. 이로부터 웹 문서들을 노드로 하여 형성된 웹 그래프로부터 각 노드에 연결되어 있는 링크들의 정보는 해당하는 웹 문서의 중요도를 평가하는데 있어 주요한 근거가 될 수 있다.

Kleinberg의 HITS 알고리즘이나 SALSA, PageRank 알고리즘 등은 이와 같은 하이퍼링크 구조에 대한 가정으로부터 웹 문서들의 내용에 포함되어진 링크 정보를 수집하여, 각각의 웹 문서들에게 우선순위 혹은 가중치를 부여한다. 이러한 링크 구조 분석 알고리즘들을 이용하여 검색을 수행한 결과는 질의어에 대한 정보만을 이용하는 기존의 검색모델들의 검색 결과보다 향상된 성능을 보여주고 있다.

대부분의 링크 분석 알고리즘들은 독립적인 검색 모델이라 기보다는, 질의어에 해당하는 용어가 문서 내에 출현하는 빈도수에 의존적인 전

통적인 내용 기반의 검색 모델에서의 결과를 보완하는 형태로 사용된다. 예를 들어 HITS 알고리즘의 경우 먼저 질의어로부터 기존의 검색 엔진에서 얻어진 검색 결과에서 상위의 문서 집합을 초기값으로 하여, 링크 정보를 이용하여 초기 집합을 확장한 뒤, 각 문서들에 대한 authority 및 hub score를 구한다. PageRank 알고리즘은 전체 문서들에 대하여, 각 문서마다 inlink의 빈도수를 이용하여 PageRank값을 계산한 뒤, 질의어 기반의 검색 결과에 대하여 우선순위를 고려하기 위한 가중치로 반영한다.

그러므로 웹 검색에서 전체적인 검색 성능은 질의어를 이용한 내용 기반의 검색 결과와 링크 정보나 HTML 태그 분석을 위주로 하는 문서 구조 분석의 결과에 모두 좌우하게 된다. 본 논문에서는 비교적 최근에 등장한 내용 기반 검색 모델인 Language 검색 모델을 소개하고, 전통적인 모델인 벡터 및 확률 모델과의 검색 성능을 비교한다. 또한 웹 환경을 위한 검색 모델로서 문서들의 링크 정보를 분석하여 확장된 형태의 새로운 Language 모델을 제시하고, 이를 적용한 검색 시스템을 구현한다.

2. 관련 연구

2.1 Language Model

Language 검색 모델에서는 전체 문서들의 Collection과 각각의 문서들을 독립된 Language 모델로서 정의한다. 이렇게 정의된 각 Language Model은 문서 Collection이나 문서가 포함하고 있는 어휘들의 집합으로 구성된다. 질의와 문서 사이의 유사도를 구하기 위해서 Language Model이 주어진 질의를 생성해 낼 수 있는 확률 값을 사용한다. 질의는 하나 이상의 질의어들로 이루어져 있으며, 각 질의어는 다른 질의어들에 대해 독립적이라고 가정하여 질의와 문서사이의 관련 정도에 대한 확률은 각각의 질의어들과 문서사이에 대한 확률을 고려함으로써 구할 수 있다.

$$P(D|Q) \propto \frac{P(Q|D)P(D)}{P(Q)}$$
$$\propto P(Q|D)P(D)$$
$$\propto P(q,,q2...qn|D)$$
$$\therefore P(D|Q) = P(D)\,\Pi P(qi|D)$$

Language 검색 모델에서는 각 Language 모델이 주어진 질의를 생성해낼 수 있는 확률 값으로부터 질의와 문서 사이의 유사도를 구한다. 질의를 구성하는 질의어에 해당하는 용어를 문서의 Language 모델이 많은 빈도수로 포함하고 있는 경우에 높은 확률 값을 갖게 되며, 문서 내에서의 용어들의 최대 빈도수에 의해서 정규화된 후, 각 질의어에 대한 확률의 곱으로서 질의에 대한 확률이 구해진다. 만일 어떤 질의어가 문서 내에 포함되어 있지 않을 때에는 질의에 대한 문서의 확률, 즉 유사도가 0이 되므로, 이러한 질의어에 대한 확률을 전체 문서의 Language 모델에서의 빈도수로부터 구하게 되고, 적당한 smoothing값으로 보정하게 된다.

$$P(D|Q) = P(D)\,\Pi P(qi|D)$$
$$= P(D)\,\Pi\,(\lambda * P(qi|D) + (1-\lambda) * P(qi|C)$$

일반적으로 Language 검색 모델의 성능은 smoothing 값인 λ에 의해 좌우되므로, 이러한 λ 값을 어떻게 정해줄 것인가 중요한 문제가 된다. <표-1>은 Language 검색 모델에서 smoothing 값을 정해줄 수 있는 method들을 보여준다. 아래에서 smoothing method는 전체 집합의 다른 문서들에 독립적으로 문서와 질의어 사의의 관계에 의해 smoothing 값을 결정하게 된다. 각각의 method들에 의한 smoothing은 실험하는 문서 collection에 따라서 다른 성능을 보여준다. 이들 중 문서 길이에 대한 정규화 방법을 이용하는 Dirichlet method가 다른 method들에 비해 일반적으로 나은 성능을 보여준다.

Methods	smoothing value	parameter
Jelinek-Mercer	λ	λ
Dirichlet	$\frac{\mu}{\sum tf(t,d)+\mu}$	μ
Absolute discount	$\frac{\delta \mid d \mid u}{\mid d \mid}$	δ

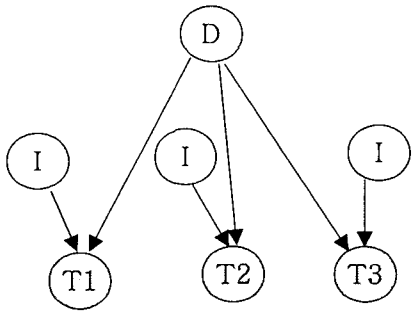
<표1- Smoothing Methods>

용어에 대한 smoothing은 문서가 용어를 포함하고 있지 않더라도 문서에 대한 확률을 구할 수 있도록 하지만, 이러한 개념을 확장하여 용어의 중요도를 이용하는 새로운 Language 모델을 정의할 수 있다. 즉, 만일 중요한 용어라면 문서로부터의 용어의 확률을 구할 수 있고, 중요하지 않다면 전체 문서 집합으로부터 용어의 확률이 구해질 수 있다. <그림-1>은 용어의 중요도 개념을 이용하여 확장된 Language 모델을 간단한 Bayesian Network으로 표현한 그래프이다. 그래프에서 색이 없는 노드들은 문서와 용어들의 중요도에 대한 독립적인 random 변수들을 나타내며, 각 노드들은 문서와 용어의 중요도에 대한 용어의 확률의 의존적인 정도를 의미한다. 각 용어들에 대한 중요도의 확률은 다른 용어들이나 문서에 대해 독립적으로 가정된다. 따라서

확장된 Language 모델에서 각 질의어에 대한 확률은 아래와 같이 중요한 질의어인 경우와 중요하지 않은 질의어인 경우의 각각에 대한 확률의 합으로서 얻어진다.

$$P(D, I_1, \ldots, I_n, T_1, \ldots, T_n) = P(D)\prod P(I_i)P(T_i \mid I_i, D)$$

$$\therefore P(D, T_1, \ldots, T_n) = P(D)\Pi P(T_i \mid D) = P(D)\Pi\sum_{i=0}^{1} P(I_i = k)P(T_i \mid I_i = k, D)$$

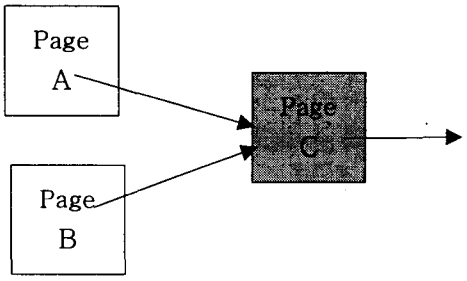


<그림-1. Bayesian Network>

2.2 Page Rank Algorithm

PageRank 알고리즘은 웹 검색을 위해 하이퍼텍스트의 링크 구조를 분석하는 대표적인 알고리즘이며 Brin과 Page에 의해 현재 가장 성능이 우수하다고 알려져 있는 웹 검색엔진인 Google을 구현하기 위해 사용되었다.

웹 문서는 다른 웹 문서로 연결하기 위한 forward 링크와 다른 웹 문서들로부터 연결되는 backward 링크들을 갖는다(<그림-2>). 보다 많은 수의 웹 문서들로부터 참조되고 있는, 즉 backward 링크들을 가진 웹 문서가 적은 수의 backward 링크를 가지고 있는 웹 문서보다 중요한 문서라는 가정은 웹 문서들마다 backward 링크의 빈도수만을 고려하게 된다. 그러나 backward 링크의 웹 문서가 다른 backward 링크의 웹 문서보다 큰 중요도를 가질 경우, 즉 예를 들어 카테고리 기반의 검색엔진인 Yahoo로부터의 backward 링크는 다른 웹 문서들로부터의 backward 링크보다 높은 가중치를 가져야만 한다. 이로부터 Kleinberg의 HITS 알고리즘이나 PageRank 알고리즘은 링크 분석의 대상이 되는 전체 웹 문서들내에서 각각의 문서들에 대한 중요도를 구하는 과정을 연속적으로 수행하여, 각각의 웹 문서들에 대한 중요도가 수렴되어진 값을 최종적인 중요도의 결과 값으로 사용한다.



<그림-2. forward와 backward 링크>

PageRank 알고리즘은 웹 문서에 대한 PageRank 값을 구하기 위해 우선 backward 링크들을 가지고 있는 웹 문서들을 찾고, 이들의 PageRank 값을 forward 링크의 수로 나눈 값들의 합을 구한다. 즉, 높은 PageRank 값을 갖는 웹 문서로부터의 backward 링크를 가질 경우, 구하려는 웹 문서의 PageRank 값에 유리하게 되며, 이것은 backward 문서의 forward 링크들의 수에 의해서 상쇄된다. 다른 웹 문서들에 영향을 주는 웹 문서의 PageRank값은 문서가 갖는 forward 링크들에게 균등하게 나누어진다. 아래의 식은 PageRank 값을 구하는 과정을 간단하게 묘사한 것이다. 매개 변수 c는 모든 웹 문서들에 대한 전체 PageRank들의 합이 상수가 되도록 정규화를 위해 사용된다.

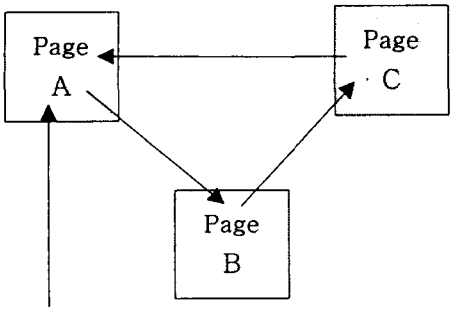
$$PR(x) = c\sum_{y \in B_x} \frac{PR(y)}{\#outlinks(y)}$$

위와 같이 PageRank 알고리즘을 수행하는 것은 웹 문서의 PageRank 값이 연결되어 있는 다른 웹 문서들에 영향을 주어 PageRank 값을 증가시켜준다. 따라서 만일 어떤 웹 문서들이 서로 연결되어져 있는 circular 그래프를 형성하게 될 때, 외부로부터 연결되어져 그래프로 들어오는 링크들만이 존재하고, 그래프로부터 외부의 웹 문서들로 나가는 링크가 존재하지 않을 경우, 그래프 내의 웹 문서들에 대한 PageRank 값은 수렴하지 않고 지속적으로 증가하게 되는 "rank sink"의 문제를 갖게 된다(<그림-3>). 이를 해결하기 위해 random surfer 모델을 이용하여, PageRank 알고리즘을 확장한다. Random surfer 모델은 웹 서핑 중인 사용자가 현재의 웹 문서에 연결되어 있는 링크를 따라 이동하다가, 직접 URL을 입력하여 관련없는 다른 웹 문서로 이동하게 되는 과정을 모델링한 것이다. 확장된 PageRank 알고리즘은 randomly surfing 확률을 나타내는 상수에 의해 backward 링크들에 대한 PageRank 값들의 합을 보정한다.

$$PR(x) = (1-)c\sum_{y \in B_x} \frac{PR(y)}{\#outlinks(y)} + c\frac{1}{N}$$

N : *the number of all webpages*

c : *the probabilty of randomly surfing*



<그림-3. rank sink problem>

3. 시스템의 설계 및 구현

3.1 확장된 검색 모델

Language 모델에서 문서의 관련 도에 대한 확률과 용어의 중요도에 대한 확률은 독립적인 random 변수들로 정의되며, 질의어에 대한 확률은 이들에 의존적이므로, 문서와 질의 사이의 관련 정도의 확률을 결정하게 된다. 따라서 문서의 관련 확률과 각 용어의 중요도에 대한 확률을 어떤 값으로 결정하는 가에 따라 Language 모델을 이용한 검색의 결과에 주요한 영향을 줄 수 있다. 이러한 문서의 관련도와 용어의 중요도에 대한 확률에 대한 초기값은 임의의 상수 값으로 정의된다. 용어의 중요도 확률은 검색을 반복한 뒤 얻어지는 feedback 정보를 이용하여 검색의 단계마다 다른 값으로 변경될 수 있다. EM(Expectation Maximization) 알고리즘을 이용하는 경우 검색이 끝난 뒤 얻어지는 질의어에 관련 있는 문서 집합을 분석하여 각 질의어에 대한 확률을 최대화시키며, 검색이 진행되면서 feedback 정보가 얻어질 때마다 각 용어를 위한 중요도의 확률을 보다 최적의 값으로 변경시킬 수 있다.

그러나 Language 모델에서 문서의 관련 있는 정도에 대한 확률은 전체 문서 집합의 모든 문서들에 균등한 값을 부여한다. 즉 문서와 질의어들과의 확률을 고려하기 전 단계에서 문서가 관련이 있을 확률은 전체 문서집합내의 모든 문서들에 대해서 모두 같다고 가정한다. 이로부터 실제적인 확률 값은 전체 문서들의 수의 역수로 구한다. 혹은 보다 많은 수의 용어를 포함하고 있는 문서가 상대적으로 중요하다는 전통적인 정보 이론의 가정으로부터 문서의 길이를 전체 문서들의 길이의 합으로 정규화한 값을 사용한다.

본 논문에서는 모든 문서들에 동일한 확률을 부여하는 대신, 문서가 갖는 링크 구조에 대하여 분석된 정보를 문서의 기본적인 관련 정도로 가정하며, 이를 위해 PageRank 알고리즘을 적용하였다. PageRank 알고리즘을 수행한 뒤 얻어진 문서의 PageRank 값이 클수록 문서가 관련이 있을 확률은 높아진다. 문서의 PageRank 값을 Language 모델에서의 확률로 반영하기 위해 전체 문서 집합의 문서들이 갖는 PageRank 값들 중 최대 값으로 정규 화하였다. 만일 두 개의 문서가 문서와 질의어 사이의 관계로부터 얻어진 확률이 같을 경우, 즉 두 개의 문서에 대해 이루어진 내용 분석의 결과가 같다면, 두 문서에 대한 우선순위는 링크 구조의 분석된 결과에 따르게 되며, PageRank 값으로부터 보다 중요한 문서를 판단할 수 있게 된다. 확장된 Language 모델은 아래와 같다.

$$P(D|Q) = P(D)\prod P(q_i|D)$$
$$= \frac{PageRank(D)}{\max PageRank}\prod P(q_i|D)$$

3.2 시스템의 구현

검색 시스템은 3.1에서의 Language 모델과 PageRank 알고리즘을 이용하여 확장한 검색 모델을 적용하여 구현되었으며, 실험 및 평가를 위해 TREC11의 웹 문서를 사용하였다. 구현된 검색 시스템은 데이터베이스의 문서들에 대한 전처리(preprocessing)를 담당하는 모듈과 문서들의 링크 정보의 취합과 PageRank 알고리즘을 수행하는 링크 구조 분석 모듈, 확장된 Language 검색 모델에 의한 우선순위를 처리하는 랭킹 모듈의 세 개의 서브시스템들로 구성된다.

문서들의 전처리 모듈에서는 TREC11의 문서 collection으로부터 웹 문서들을 가져오고, 문서들의 내용을 tokenization과 불용어의 제거, Porter 알고리즘을 사용한 stemming 과정을 거쳐서 용어들에 대한 indexing 데이터베이스를 구축하였다. TREC11로부터 가져온 웹 문서의 수는 1247753개이며, 문서들로부터 indexing 되어진 용어의 수는 3379618개였다. 링크 구조 분석 모듈에서는 문서들이 가지고 있는 링크들을 취합한 뒤, PageRank 알고리즘을 수행하였다. 문서와 질의 사이의 확률을 구하기 위한 랭킹 모델은 아래와 같이 구현되었다. 용어의 smoothing을 위한 λ의 값은 0.15를 사용하였다. ρ는 링크 분석으로 부터의 결과와 내용 분석의 결과에 가중치를 조정하기 위해 사용하였다.

$$score(d,q) = \rho\ link\ score(d) + content\ score(d,q)$$

$$= \rho \frac{PageRank(d)}{\max(PageRank_c)} + \sum_{k=1}^{m} d_k \cdot q_k$$

$$q_k = qtf$$

$$d_k = log(1 + \frac{tf_k * \sum_{(t,k)} df(t,k)}{df_k * \sum_t tf(t,d)} * \frac{\lambda_k}{1-\lambda_k})$$

4. 실험 및 평가

링크 구조 분석으로 확장된 검색 모델의 성능을 비교하기 위해서 아래와 같은 4종류의 다른 버전을 구현하였다. 첫 번째의 모델은 Language 모델에서의 기본적인 식이다. 두 번째의 모델은 첫 번째 모델에서의 결과에 문서의 길이의 로그값을 적용하였다. 세 번째의 모델은 정규 화된 문서의 PageRank값을 첫 번째 모델의 결과에 적용하였다. 네 번째의 모델은 Okapi 시스템에서 구현된 BM25 모델을 사용하였다. BM25 모델은 기존의 확률 기반의 검색 모델 중에서 가장 좋은 검색 성능을 보이는 것으로 알려져 있다.

$$score_1(d,q) = log(1 + \frac{tf_k * \sum_{(t,k)} df(t,k)}{df_k * \sum_t tf(t,d)} * \frac{\lambda_k}{1-\lambda_k})$$

$$score_2(d,q) = log(\sum_t tf(t,d)) + score_1(d,q)$$

$$score_3(d,q) = \rho \frac{PageRank(d)}{\max(PageRank_c)} + score_1(d,q)$$

TREC11의 topic중 10여개를 선택하여 질의로 사용하였으며, 3종류의 모델들에 대하여 검색을 시도하였다. 이들의 성능을 평가하기 위해 상위의 5개, 10개, 30개의 문서들에 대한 precision을 측정하여 각각에 대한 평균 정확도를 구하였다. <표-1>의 결과에서와 같이 링크 정보를 이용한 모델의 경우가 상위 5개와 10개의 문서들 중에서 가장 높은 정확도를 가졌지만, 10개까지의 문서들에 대해서 낮은 정확도를 갖는 정규화된 문서 길이를 사용하는 모델이 상위 문서의 범위를 확대 할수록 좋은 결과를 보였다.

run	precision at doc5	precision at doc10	precision at doc30
version1	0.4300	0.4000	0.2667
version2	0.3000	0.3000	0.3333
version3	0.4500	0.4200	0.2667

<표-1. 상위 문서들에 대한 정확도>

5. 결론 및 향후 과제

웹 환경을 기반을 이루고 있는 웹 페이지들은 HTML이라는 공통된 문서 형식과 웹 페이지들 사이에 방향성을 갖는 링크로 연결된 구조를 가지고 있다. 이러한 웹 환경에서의 특수성을 이용하여 검색 모델에 적용하는 것은 사용자의 요청으로부터 정확하고 정보를 찾고, 얻어진 문서들 중 보다 중요한 정보를 판단하여 제시하기 위해 도움을 줄 수 있다. 향후 과제로는 보다 좋은 결과를 얻기 위해서 링크 구조 분석으로부터 얻어진 결과와 내용 분석으로부터 얻어진 결과들에 대한 가중치 부여 방법을 개선해야 할 필요가 있다.

6. 참고 문헌

1. Jon Kleinberg. Authoritative sources in a hyperlinked environment. In Proceedings of the Nineth Annual ACM-SIAM Symposium on Discrete Algorithms.,1998.
2. [Brin98] S.Brin, L.Page, The anatomy of a large-scale hypertextual web search engine, WWW8, 1998
3. K.Ng. A maximum likelihood ratio information retrieval model, In Proceedings of the 8th Text Retrieval Conference, TREC-8, NIST Special Publications, 1999
4. C. Zhai and J.Lafferty, A study of smoothing methods for language models applied to ad hoc information retreival, In Proceedings of the 24th ACM Conference on Research and Development in Information Retrieval(SIGIR'01), pages 334-342, 2001
5. J.M. Ponte and W.B. Croft, A language modeling approach to information retrieval, In Proceedings of the 21st ACM Conference on Research and Development in Information