



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2008-0036464
(43) 공개일자 2008년04월28일

(51) Int. Cl.

G06F 17/30 (2006.01) G06F 17/00 (2006.01)

(21) 출원번호 10-2006-0103091

(22) 출원일자 2006년10월23일

심사청구일자 없음

(71) 출원인

이우기

경기 군포시 산본동 한라2차아파트 414-1005

(72) 발명자

이우기

경기 군포시 산본동 한라2차아파트 414-1005

(74) 대리인

특허법인네이트

전체 청구항 수 : 총 1 항

(54) 동적인 계층적 웹 검색 트리 시스템

(57) 요약

본 발명은 인터넷 검색시스템 및 브라우징에 관한 것으로, 사용자가 검색엔진에 검색식을 입력하여 URL을 검색하는 단계와; 사용자가 상기 검색엔진에서 검색된 URL 중 루트노드를 선정하는 단계와; 상기 루트노드가 선정된 후, 해당 URL이 지칭하는 웹페이지의내용이 제시되는 단계와; 해당 웹 페이지의 웹사이트에의 노드와 아크를 분석하는 단계와; 각 아크의 가중치를 구하는 단계와; 검색요구조건에 맞는 검색트리를 정하고 사용자의 웹브라우저에 제시하는 단계를 포함하는 구조화된 동적 검색트리제공방법 및 그 검색시스템을 제공한다.

대표도 - 도14



특허청구의 범위

청구항 1

사용자가 검색엔진에 검색식을 입력하여 URL을 검색하는 단계와;
 사용자가 상기 검색엔진에서 검색된 URL 중 루트노드를 선정하는 단계와;
 상기 루트노드가 선정된 후, 해당 URL이 지칭하는 웹페이지의내용이 제시되는 단계와;
 해당 웹 페이지의 웹사이트의 노드와 아크를 분석하는 단계와;
 각 아크의 가중치를 구하는 단계와;
 검색요구조건에 맞는 검색트리를 정하고 사용자의 웹브라우저에 제시하는 단계
 를 포함하는 동적 검색트리제공방법

명 세 서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

- <17> 웹의 폭발적 성장은 양적 질적으로 괄목할 만하다. 양적으로 웹은 현재 조 단위의 페이지를 넘어선지 오래되었다. 질적으로는 단순한 HTML, XML 등의 웹 페이지뿐만 아니라, 전자화된 매뉴얼, 이메일, 그림, 동영상, 심지어 웹 서비스까지 포괄한다. 이렇게 커질수록, 웹 사용자는 점점 더 원하는 정확한 정보를 찾기가 더욱 어려워진다.
- <18> 일반적인 검색엔진을 통한 웹 정보검색의 절차는 다음과 같다. 우선, 사용자가 검색엔진 시스템에 검색질의를 입력한다. 그러면 검색엔진은 입력사항을 반영하여 개별적인 점수화기법(weighting)에 따라 계산하여 높은 값부터 순위를 매긴 URL목록을 사용자에게 제시한다. 마지막으로, 제시되는 목록을 보고 사용자는 URL의 하나를 선택하여 클릭하면, 해당 객체가 사용자에게 전송되어 브라우저에 표현된다. 이를 웹 정보검색이라고 한다.
- <19> 그런데 이러한 웹 정보 검색의 과정을 살펴보면, 우선 사용자의 검색이 도 1과 같은 웹 객체의 집합(이하 “웹 사이트”)에 대하여 이루어진다고 할 경우, 검색대상이 되는 웹사이트는 사각형과 화살표로 구성되는데, 이는 각각 웹객체와 링크가 된다. 예를 들어 도1과 같은 경우를 살펴보면, 검색결과로 A2, 메인(MAIN), B1, A3, ```` 등과 같은 방식으로 URL이 나열되어 제공된다.
- <20> 즉, 웹 객체들이 일차원적으로 나열되고, 웹객체들이 서로 연결되어 있는 링크정보는 나타나지 않는다. 따라서, 웹이 가지고 있는 링크의 풍부한 표현능력을 묵살하거나 최소한 제공하지 않는다는 것이다. 따라서 사용자는 전체웹사이트에서 현재 자신이 어느 위치에 있는지, 얼마나 더 검색해야 하는지, 단말노드(leaf node)인지 시작노드인지, 자식노드가 얼마나 연결되어 있는지, 등에 대한 정보가 없어 유사한 웹 객체 주변의 사이클을 맴돌 수 있으며, 다시 방문하는 경우에도 방향인지에 어려움이 있다. 이는 또한 사용자의 웹 객체들에 대한 검색 경로의 정보 또한 놓친다는 것을 의미한다. 이는 근본적으로 종래의 정보검색시스템이 도서검색에서 시작되어 문자 위주의 검색으로 이루어져 왔고, 또한 사용자의 화면에도 문자위주로 표현되어 온 기존의 표현기법이 개선되지 않고 있다는 점에 기인한다.

발명이 이루고자 하는 기술적 과제

- <21> 본 발명은 웹 정보를 구조화하여 표현하는 의미 있는 접근법을 제공한다.
- <22> 또한 필요한 정보수에 따라 노드를 최적으로 순위화하여 단순화하는 방법을 제공한다.
- <23> 또한 이를 웹 브라우저에 표현하는 브라우징 방법을 제공한다.

발명의 구성 및 작용

- <24> 웹을 유향그래프(a directed graph) $G = (N, A)$ 로 인식하되, 유향그래프 G 의 N 과 A 는 웹 페이지를 의미하는 노

트집합과 노드사이의 아크집합을 각각 의미하고, 이러한 웹 그래프에서 구조화 검색트리(structured search tree)를 형성한다. 또한 우선검색트리의 형성이 웹 사용자의 선택에 따라 동적으로 이루어질 수 있어서, 검색의 시작을 어느 노드에서 하느냐에 따라 그 위치를 루트로 하는 검색트리를 만들 수 있도록 하는 것이다. 또한 하위그래프 $G^* = (N^*, A^*)$ 는 그래프 G 로 부터 유도되는데, 만일 아크 $A^* \subseteq A$ 의 각 시작노드 및 끝 노드가 각각 $N^* \subseteq N$ 에 속해야 한다. 이 경우 하위 그래프 G^* 는 검색 대상 즉, 구조적 검색도메인이라고 부르게 된다. 이 도메인은 검색의 대상이 되는 노드와 아크를 가르키고, 도 3에서는 이 도메인을 각각 굵은 다각형으로 표시하였다. 예컨대, 도 3a의 경우, 전체 그래프영역이 검색대상이 되어 $N^* = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ 이 된다. 도 3b 및 도 3c는 각각 노드 $\{6, 8, 9\}$ 및 $\{6\}$ 이 검색도메인이 되는 것이다. 여기서 얻어질 수 있는 검색트리는 구체적으로는 루트를 가진 유향 스패닝 트리(rooted directed spanning tree) 혹은 루티드 아보레센스(rooted arborescence)라고 하는데, 여기서는 검색트리라고 일반화하여 줄여 부르도록 한다. 이러한 검색트리에서 개별노드로 입력되는 아크는 하나로서 모두 $n-1$ 개의 아크가 n 개의 노드에 연결되며 사이클이 없는 유향그래프가 되어야 하고, 결과적으로 검색트리는 단 하나의 루트로부터 모든 노드에 유일한 경로가 존재하는 경우를 의미한다. 예를 들어 도 3에서는 3개의 트리가 있는데 도 3a트리는 “0” 노드를 루트로 하는 경우이고, 도 3b는 “8” 노드가 루트로 선택되는 경우, 그리고, 도 3c는 “6” 노드가 루트로 선택된 경우를 각각 나타내고 있다.

<25> 또한 중요한 점은 검색대상 도메인 즉, 검색트리를 생성하게 되는 범위문제의 경우, 이러한 도메인은 선정된 루트노드로부터 도달가능한(reachable) 웹 페이지들의 집합으로써 정의된다. 도 3a, 3b, 3c에서는 각각 굵은 테두리로 표현되어 있다. 잠정적으로 인덱스 “0”을 가지는 노드를 특정 웹 사이트의 시작페이지(default page)로서 index.html이나 default.php 등을 검색대상의 루트로 결정하는 경우 루트에서부터 도달가능한 웹 사이트 전체가 대상 도메인이 될 수 있다. 다른 극단으로는 도 3c와 같이, 단 하나의 단말페이지(dangling page)가 대상 도메인이 될 수도 있다. 이러한 양 극단을 포함하는 일반적인 여러 웹 페이지가 하이퍼링크로 연결되어 있는 검색도메인을 잘 표현할 수 있는 검색트리를 찾는 것이 본 발명의 대상이 된다.

<26> 먼저 가중치 평가방법을 설명하면, 웹 노드(N) 및 웹 아크(A)를 표현하는 일반적인 가중치(weight) 혹은 거리 기준(distance measure)은 정보검색 도메인에서는 글로벌 가중치와 로컬 가중치로 표현된다. 이 두 가지 가중치는 어떤 의미에서 상호 독립적이다. 즉, 글로벌가중치는 개별적 검색 키워드를 고려하지 않고 하이퍼링크 정보만을 이용하여 구해지며, 한편 로컬가중치는 하이퍼링크를 배제하고 키워드 및 질의어를 사용하여 가중치를 구하기 때문이다. 현재 많이 쓰이는 글로벌가중치 방식에는 페이지랭크 방법, 제이클라인버그(J. Kleinberg)의 HITS방법 등이 있다. 로컬가중치로서는 벡터공간모형(VSM)에 기반한 코사인가중치, 확률적가중치, 퍼지가중치 및 흔히 쓰이는 tf-idf(term frequency and inverse document frequency)방법 등이 있다.

<27> 노드의 내용(키워드)에 기반한 노드 가중치 평가방법과 달리 아크정보를 활용한 아크기반 노드가중치 평가방법이 최근 구글의 성공과 함께 주목받고 있다. 이 방법에서는 인접행렬(adjacency matrix) M 에 대해 정의하기를 만일 아크 i 에서 j 로의 링크가 존재하면 $M[i, j] = 1$ 이고, 없으면 0으로 나타낸다. 이때 $i, j \in \bar{N}$. 이러한 인접행렬에 기초하여 여러 가지 가중치 계산기법들이 개발되고 있다.

<28> 페이지랭크와 같은 글로벌가중치 만으로는 개별적 검색결과가 지원되지 않으므로 다음과 같은 로컬가중치(local weight)가 필요하다. 이는 노드 기반의 노드 가중치 평가방식이라 명명될 수 있으며, 그 대부분이 전통적 정보 검색에서 사용하는 벡터공간모델(VSM: vector space model)에 기반하고 있고, 그 중에서도 가장 많이 사용하는 방법의 하나가 tf-idf이다. 본 발명에서는 글로벌가중치와 로컬가중치의 곱을 사용하여 가중치로 사용하였지만, 가중치방식의 선정과 그 결합방식이 본 발명에 제약요인이 되지는 않는다.

<29> 다음으로 웹 구조화 모델을 설명한다.

<30> 여기에서의 유향그래프는 루트노드를 가진다고 가정하는데, 이 노드는 시작노드 집합에서 얻고, 이 시작노드 집합을 얻는 방법은 여러 가지가 가능한데 가장 일반적인 방법은 기존의 검색엔진에 사용자가 질의하여 얻어지는 URL 리스트를 사용하는 것이다. 그러면 상기 유향그래프와 그로부터 얻어지는 검색트리는 루트노드와 하이퍼텍스트 링크 즉, 아크로 연결된 한 웹 사이트내의 다른 노드들의 집합으로 구성된다. 그리고 이 집합이 검색을 위한 구조화의 대상도메인이 된다.

<31> 이는 웹사이트를 구조화하기 위해, 아크 중요도의 합을 최대화하는 계층적인 구조를 만들어내는 해법인 것이다. 이러한 구조는 최대 중요도 트리생성 문제로 변형할 수 있으며, 이는 기본적으로 최소비용트리 문제는 다음과 같은 트리생성 알고리즘으로 생성할 수 있다. 본 방법은 본 발명자가 발표한 에이피웹 워크샵(APWeb Workshops, 2006)) pp. 1004-1011에 기반하여 구현하였으며, 주어진 그래프에서 모든 아크의 중요도 $w_{(i,j)}$ 그리고 루트 노드

r 을 입력으로 하여 다음 알고리즘은 중요도 합을 최대로 하는 r -rooted spanning 트리구조를 생성하는 것이다.

<32>

다음으로 본 발명에 따른 필터링 알고리즘을 설명한다.

<33>

이 단계는 상기 얻어지는 결과물의 크기를 줄이는 과정이다. 현실적으로 한 웹사이트 내의 웹페이지들은 수백 내지 수 만개 이상으로 구성되어 있으므로 단지 최적구조를 얻었다라고 실제 웹 검색에 사용될 수 있도록 하려면 상기 얻어지는 구성요소들의 크기를 줄이는 과정이 필요하다. 다시 말하면, 다이나믹 검색트리(DST:dynamic search tree)를 실제 검색에서도 효과적으로 사용될 수 있게 하려면 사용자의 관심영역에 맞추어 제공할 수 있어야 한다는 것이다. 여기서는 2가지 종류의 추출알고리즘을 제시하며, 이는 상기 얻어진 다이나믹 검색트리(DST)를 추출된 검색트리(DST:dynamic search tree)로 변화시키는 과정인 것이다. 우선 그 한 가지는 필터링을 수행할 기준을 정하고 그 기준에 미달하는 아크와 관련 노드를 제거하는 것이다. 다른 하나는 아크 웨이트를 정렬하여 최대 k 개를 제외하고 나머지를 제거하거나 혹은 차례차례 표현한다는 것이다. 본 연구에서는 첫 번째 방법을 *filter_tree* 알고리즘에서 설명하고, 두 번째 방법을 *top_k_tree* 알고리즘으로 각각 설명한다.

<34>

우선, 가장 자명한 경우로서 단 하나의 노드만이 검색대상이 되는 경우이다. 이를 *dangling node*라고도 표현한다. 이 경우 필터트리는 단 하나의 노드를 결과로 제시한다. 참고로 그림 3(d)를 보면 된다. 다른 관점에서 보면 단 하나의 노드만 제시되는 경우란 일반적인 검색엔진의 검색과 동일하다. 이는 우리가 가정하기를 검색엔진에서 시작노드집합을 그대로 가져와 검색을 시작하기로 했기 때문에 자명하다. 그러면 그 외의 경우를 살펴보자. 이는 출발노드와 다른 노드들이 아크를 통해 연결되어있다는 의미이다. 이 경우는 그림3 (a)-(c)등이 해당된다. 다음 두 알고리즘에서 앞선 DST를 통해 얻어지는 최적트리를 대상으로 다음과 같이 *Filter_tree* 알고리즘 및 *top_k* 알고리즘을 적용하게 된다.

<35>

한편, 도 4의 알고리즘에서는 다음 설명되는 기호를 사용한다.

<36>

● DST 다이나믹 검색트리(dynamic search tree)

<37>

● FST 필터된 검색트리(filtered search tree)

<38>

● N^* DST에 포함되는 노드집합

<39>

● C FST에 포함되는 노드집합

<40>

● u, v, n 집합 N 의 원소인 노드

<41>

● $succ(v)$ 노드 v 와 연결된 자녀

<42>

● $w_{(u,v)}$ 노드 u 에서 노드 v 로의 아크(u,v)에 대한 가중치

<43>

● f 필터 기준치

<44>

우선 다음에 제시되는 *filter_tree* 알고리즘을 살펴보자. 이 알고리즘의 핵심은 9.라인에 따라 아크 가중치가 필터링 기준보다 적으면 해당 아크와 노드가 제거되고, 다음으로 그 하위의 아크가 있을 경우 제거된 아크를 대신하여 그 부모노드를 변경해준다는 부분이 12.~15.라인에 설명되어 있다. 이 알고리즘은 단말노드를 찾는 부분 즉, 1.~4.라인에서 시작되며 그 해당 노드와 아크가 제거될 것인지의 유무를 판단하며 이를 단말노드에서 형제노드로, 형제노드가 없으면 그 부모노드로 진행하고 루트노드에서 이 과정이 종료된다. 이 알고리즘의 복잡도는 단순히 모든 노드를 한번만 찾아 필터기준에 부합되는 지를 비교하므로 $O(n)$ 이다.

<45>

도 5에서는 주어진 DST를 2개의 FST로 보여주는 것이다. 우선 도 5(a)는 상기 설명된 예제에 대한 최적트리 즉 DST이다. 이는 다시 표현으로 말하자면 필터를 하지 않은(즉, $f = 0.00$)인 상태의 트리인 것이다. 이때 *filter* 기준 $f = 1.0$ 이 제시된다면, 아크(1,3)과 (2,7) 및 관련노드 3과 7이 제거대상이 되고, 그 결과는 도 5(b)에 제시되어 있다. 그런데 만일 *filter* 기준 $f = 1.5$ 가 제시된다면, 그 처리절차는 다음과 같다. 우선 단말노드를 가지는 노드8에서 아크(8,9)와 (8,6) 및 관련노드 9과 6이 제거된다. 이와 같은 방법으로 진행하다가, 노드1의 경우 아크(1,3) 및 관련노드3을 제거한다음, 아크(0,1) 및 관련노드 1이 제거되면서는 후속아크 (1,4)가 제거되는 대신 아크(0,4)가 신설된다. 마지막으로 루트노드에서 알고리즘이 종료된다. 이상의 내용이 도 5(c)에 제시되어 있다.

<46>

다음은 *Top-k* 트리 알고리즘에 대하여 설명한다. 웹 사용자는 f 값을 모르는 상태에서 특정 k 개의 결과를 요청

하는 경우가 일반적이다. 이를 다음과 같은 top_k 트리 알고리즘에서 다음과 같이 지원한다.

- <47> 우선, 아크 가중치를 소트한다. 여기에서 최상위 k개를 선택하고(물론 노드 수는 k+1개가 된다), 바로 그 k번째 아크의 가중치 값을 구한다. 이 값이 상기 언급한 필터값이 되는 것이다. 만일 소트된 k번째 아크의 가중치에서 타이거 발생할 경우(즉, 필터기준이 되는 아크의 가중치를 동일한 아크가 여러 개 있는 경우) 동점해소(tie-breaking) 및 선택방법이 필요해진다. 이 경우 본 연구에서는 차례대로 노드 번호에 따라 처음으로 k개가 찰 때까지의 노드를 선택한다. 그 의미는 처음 노드번호를 부여할 때 넓이 우선 탐색방식을 사용하여 번호를 부여했기 때문에 루트로부터 가까운 좌측우선의 위상(Topology)을 우대하는 결과가 된다. 굳이 본문에서는 그 이상의 동점해소 및 선택방식을 생각하지 않는다. 최고 k개의 구조적 검색결과를 얻는 top-k 알고리즘은 다음과 같다.
- <48> 최고 k개의 구조적 검색 알고리즘을 적용하는 구체적인 예를 도 5에서 살펴보기로 하자. 일단 자명한 경우란 0개의 노드만을 요구할 때이다. 즉, $k = 0$ 인다, 이 경우는 루트노드만 결과로 보이는 것이다. 자명하지 않은 경우란 사용자가 1 이상의 “k”개의 구조적 검색결과를 요구하는 경우이다. 그러면 top_k_tree 알고리즘은 도 5(a)를 입력으로 받은 다음 그 각각의 가중치를 소트한다. 그러면 {2.28, 1.76, 1.62, 1.25, 1.15, 1.09, 1.05, 1.00, 0.62}와 같고, 만일 사용자가 이중에서 top-7 혹은 top-3를 요구하는 경우 그 결과는 각각 도 5(b) 및 도 5(c)가 될 것이다.
- <49> 이하, 실제 웹 사이트를 대상으로 구조적 검색방식을 구현한 내용을 설명하기로 한다. 대상이 되는 웹 사이트를 선정하는 방법은 우선 구글에서 다음 3개의 키워드를 입력하여 얻어진 결과를 가지고 구조적 검색을 구현한 것이다. 3개의 키워드는 “hurricane, kids, safety”였다.
- <50> 시작단계(Initiation): 구조적검색을 위한 시작집합을 얻는다.
- <51> 사용자가 검색엔진에 키워드를 입력한다. 이 내용이 표1에 나타나있다. 표 1에서는 노드id들과 관련된 URL들을 보여준다.
- <52> 단계1: 루트노드 선정
- <53> 도 7의 URL들은 각각 구조적 검색에서 루트노드가 될 수 있다. 도 3을 살펴보면, 노드 중에서 색깔이 칠해진 노드 “1” 이 도 3(c)의 루트가 된다는 의미이다. 본 실험에서는 10번째 탐색 URL인 “hurricand.lsu.edu”를 루트노드로 선정했다.
- <54> 단계2: 검색도메인(G^*)
- <55> 단계1에서 선정된 루트노드로부터 검색대상이 되는 것은 루트 노드로부터 도달가능(Reachable)한 URL 집합(Λ^*) 이 검색대상이 되는 도메인(G^*)이 된다. 루트노드를 결정하면 사용자의 브라우저에는 해당 URL이 지칭하는 웹 페이지의 내용이 보여지게 된다. 이와 동시에 해당 URL이 유향그래프 G^* 는 그림A.1처럼 표현된다. 각 URL은 표 A.1에 나타나있고, 웹사이트의 구조가 분석된다. 예를 들어, 도 7에서 첫 번째 URL인 www.maxpress.com/hkn/의 경우 199개의 노드와 1941개의 아크로 구성되어있다. 그리고 10th URL인 hurricand.lsu.edu 의 경우 140개의 노드와 333개의 아크로 구성되어있다. 이를 분석한 내용이 표 2에 정리되어있다.
- <56> 단계3: 아크 가중치
- <57> 아크 가중치는 여러 가지 방법으로 계산할 수 있으며, 여기서는 글로벌 가중치로서 PageRank와 로컬 가중치로서 tf-idf의 곱으로 정의된 식으로 구한 것이다.
- <58> 단계4: 다이내믹 검색트리(DST)
- <59> 상기 절차로 주어진 아크 가중치와 검색도메인에 대해 상기 모델링에 의해 혹은 상기 알고리즘에 의해 DST 즉, 최적트리를 구한다. 예컨대, 상기 분석된 예에 대해서는 도 8에 DST가 제시되었다.
- <60> 단계5: 추출된 검색트리(FST)
- <61> 단계4에서 구한 DST에 대하여, 상기 설명한 filter_tree 알고리즘이나 top_k_tree 알고리즘을 통해 다양한 필터기준 혹은 검색요구기준을 적용하여 FST를 구한다. 예컨대, 도 9는 은 필터기준 0.5에 대해 구한 것이며, 도 10에는 3가지 기준 $f = 1.00, 2.00$ 및 4.00 을 적용한 결과가 각각 제시되어있다. 다른 표현으로는 top-3결과가 그림A4(c)에 제시되었다고 할 수 있다.
- <62> 추가적으로 고려할 부분을 상기 언급한 바와 같이 사용자가 어떤 URL을 선정하는가에 따라 루트를 바꿀 수 있고, 그 루트로부터 상기 언급된 단계를 적용하여 새로운 검색결과를 얻을 수 있다. 그래서 다이내믹(Dynami

c)이라는 용어를 사용하였다. 예컨대, 그림A.5는 사용자가 6번 노드 즉, <http://hurricane.lsu.edu/publications.htm>를 선택한 경우에 대한 DST를 표현하고 있다. 최종적으로 사용자는 도 10 도 11에 나타난 구조적 검색결과를 얻을 수 있으며, 이것은 사용자에게 추천(recommendation) 혹은 향후 미리 검색할 방향을 제시하는 의미가 된다. 또한 만일 사용자가 상위 k개의 결과를 원할 경우, 아크 가중치가 소트되고 그 중에서 상위 k개가 상기 설명된 top_k_tree알고리즘에 의해 제시된다. 예컨대, 상위 4개를 요청하는 경우의 최종 결과가 도 12에 제시되어 있다. 그 결과는 여러 가지로 표현될 수 있는데, 해당 그림에서 보여지는 바는 화면을 프레임으로 나누고 좌측에는 URL의 구조를 보여주고 우측에는 보여진 상기 구조 중에서 한 가지 URL을 선택할 경우 해당 웹 페이지의 내용이 보여지는 시스템을 구현한 경우이다.

<63> 도 14 및 도 15는 다른 예로서, 특정 도메인에 대한 웹 페이지 정보들을 수집하여 그래프 형태의 구조를 트리 형태로 변환하고 웹 브라우저에 구현한 결과를 보여준 예시다.

<64> 도 14는 <http://www.dlf.org.uk/about>이 본 발명에 따른 웹검색시스템이 웹 브라우저에 나타난 상태이다. 특정 도메인 내의 웹 페이지의 계층 구조는 오른쪽 상단에 있는 트리뷰(Tree View) 컨트롤에 나타내어진다. 이 컨트롤은 위치의 이동과 크기의 축소와 확장이 가능하다. 컨트롤 내에서 웹 페이지는 웹 페이지 번호와 함께 트리노드(Tree Node)로 표현된다. 트리노드는 한 개의 부모 노드와 여러 개의 자식 노드를 갖는다. 현재 페이지의 노드 번호는 1번이며, 웹 페이지 번호가 4번인 노드 <http://www.dlf.org.uk/news>의 자식 노드이다.

<65> 상위구조정보는 트리뷰 컨트롤로 화면에 표시된다. 이 트리뷰는 현재 페이지와 현재 페이지의 하위 페이지를 제외한 웹 페이지 노드로 구성되어 있다. 도 15에서 표시되어 있는 1번 노드는 오른쪽의 트리뷰에는 표시가 되어 있지만 왼쪽의 트리뷰에는 1번 노드와 1번 노드의 자식 노드들은 표시되어 있지 않다.

<66> 이하에서는, 본 발명과 관련된 그동안의 연구와 비교하여 본다.

<67> 최근 웹의 구조를 분석하는 접근법은 관련된 정보의 검색의 효과성 및 효율성을 제고하는데 중요한 요소로 간주되고 있다. 웹의 구조적 검색 관점에서 하이퍼링크의 구조를 적용하는 연구는 크게 보아 웹 객체 가중치평가, 검색엔진에서의 웹 페이지 순위부여방식, 토폴로지 및 시각화 방식 등이 있다. 웹 객체 가중치평가에 있어서는 정보검색 연구자 및 개발자들의 경우 페이지랭크 및 HITS가 가장 많이 활용되고 있다. 웹의 아크 분석에서 HITS의 경우 매우 주목을 받은 알고리즘으로서 연결성 노드(hubs)와 의미를 가진 노드(authorities)를 매우 효율적인 반복연산을 통해 제시했다. 의미를 가진 노드란 특정 주제에 대해 좋은 내용을 담고 있는 노드를 의미하며 좋은 연결성 노드에 아크를 많이 가지며, 또한 좋은 연결성 노드는 의미를 가진 노드에 많이 연결된다는 선순환 구조에 가정하고 있다. 이 알고리즘은 여러 분야에 활용되었는데 상호강화 관계(mutually reinforcing relationships)라고 부르는 사회망(social network)을 찾는 문제에 주로 응용되고 있지만, 웹 페이지 검색에는 사용되지 않는다. 여기서는 페이지랭크와 아보레슨스 방식을 글로벌 가중치로서 사용했다. 아보레슨스 방식은 그 어떤 웹 검색 시스템에도 적용되지 않은 방식이다. 로컬 가중치는 기존의 정보검색 연구진영에서 매우 많은 시도를 해왔Dmau, 전통적인 벡터공간모형(vector space model)에 입각한 코사인 가중치, 확률 및 퍼지방식, 그리고 본 연구에 도입한 tf-idf등의 방식이 그것들이다. 그런데 본 연구에서 제안하는 글로벌 가중치 및 로컬 가중치의 통합 그리고 이의 아크 가중치로의 전환 등은 최초의 시도이다. 글로벌 및 로컬 가중치 통합의 또다른 시도는 후와 쟁(Hou and Zhang)이 개발한 LLI가 있으며 이는 singular value decomposition 방식을 활용한 것으로, 본 발명과는 다르다.

<68> 한편으로 웹 검색을 지원하는 시각화 도구를 개발하는데 있어서는 하이퍼볼릭(Hyperbolic) 브라우저의 경우 2차원 하이퍼볼릭 공간에 트리를 표현하였고, 차-차(Cha-Cha) 및 체셜(Cheshire)의 경우 인트라넷 검색엔진에서 키워드 정보 및 XML정보를 계층적으로 표현하려고 하였으며, 토요다와 기츄레가와(Toyota & Kitsuregawa)의 경우에도 사회망 및 웹 아카이브에 대한 시각화 도구를 제안하였다. 이러한 시각화 도구들은 그래프적 표현에 초점을 맞추었지, 웹 검색의 방향을 제안하거나 더구나 최적구조(optimal structure)을 고려한 경우는 없었다.

발명의 효과

<69> 본 발명에서는, 웹 검색에 있어서 새로운 접근법 즉, 구조화 검색이라는 효과적인 방식을 제안했다. 구조화 웹 검색은 크게 3부분으로 나누어 설명할 수 있는데, 우선 웹 객체에 있어서 노드(웹 페이지)와 아크(하이퍼링크)의 유향 그래프구조에서 아크의 가중치를 구하는 방법과 둘째, 최적화 기법을 활용하여 루티드 스페닝트리 즉, 최적트리를 생성하는 방법, 셋째, 이러한 최적트리에서 사용자 검색을 위해 가중치가 낮은 아크와 노드를 제거하는 필터링 알고리즘과 최우선 k개의 결과를 구조화하여 보여주는 top-k알고리즘이 그것이다. 이 과정을 간단한 예제로 설명하고 시스템을 구현하여 실제 시스템에 대해 보여주었다.

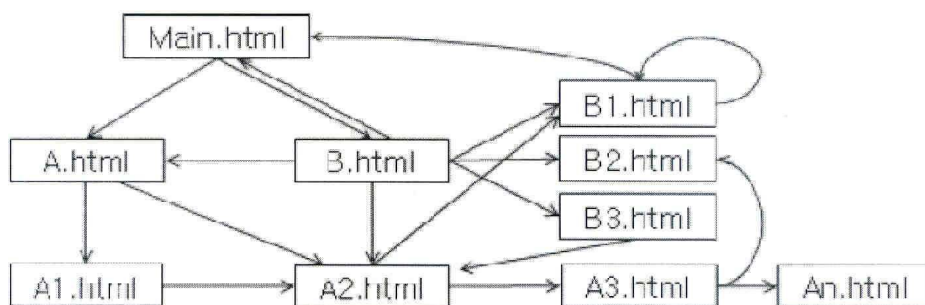
<70> 본 발명의 가장 큰 공헌은 다음과 같다. 첫째, 새로운 웹의 계층적 구조화를 통한 새로운 검색방식을 제안했다. 둘째, 노드 가중치를 구하는 방식을 통합하는 아크 가중치 평가방식을 제안했다. 셋째, 최적화 기법을 정보검색 분야에 적용하는 새로운 연구의 장을 열었다는 등이다.

도면의 간단한 설명

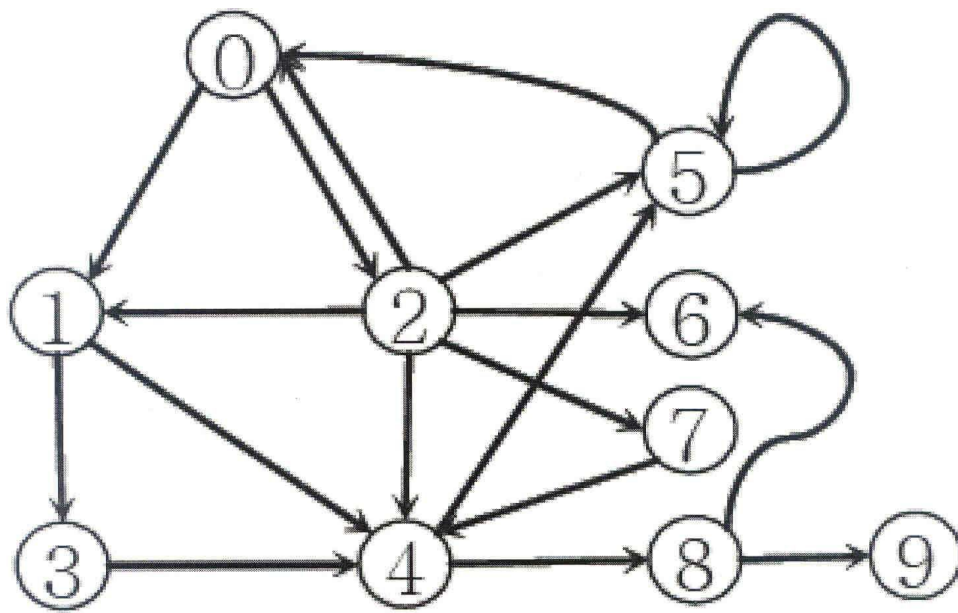
- <1> 도 1은 특정 웹사이트를 예시한 블록도
- <2> 도 2는 웹 유형 그래프
- <3> 도 3a, 3b, 3c는 서로다른 시작노드와 검색도메인을 가지는 세가지의 검색트리의 예를 도시한 구성도
- <4> 도 4는 본 발명에 따른 필터트리의 알고리즘
- <5> 도 5a는 본 발명에 따른 최적트리,
- <6> 도 5b는 본 발명에 따른 필터기준을 1.0으로 했을때의 출력트리,
- <7> 도 5c는 본 발명에 따른 필터기준을 1.5로 했을때의 출력트리
- <8> 도 6은 본 발명에 따른 top_k_tree 알고리즘
- <9> 도 7은 본 발명에 따라 노드 “0” 을 루트로 하는 hurricand.lsu.edu의 구조화된 검색도메인
- <10> 도 8은 도 7의 필터기준 0의 동적검색트리
- <11> 도 9는 도 7의 필터기준 0.5의 동적검색트리
- <12> 도 10a, 도 10b, 도 10c 는 각각 필터기준을 1.00, 2. 00, 4.00을 적용한 결과이다.
- <13> 도 11은 사용자가 특정노드를 선택한 경우의 계층적 동적검색트리.
- <14> 도 12a는 노드 6을 루트로 하는 필터된 검색트리, 도 12b는 상위 4개를 선택한 필터된 검색트리
- <15> 도 13은 도 12b에 따른 계층적 동적검색트리
- <16> 도 14 및 도 15는 본 발명에 따른 계층적 동적검색트리가 나타난 다른 예시도.

도면

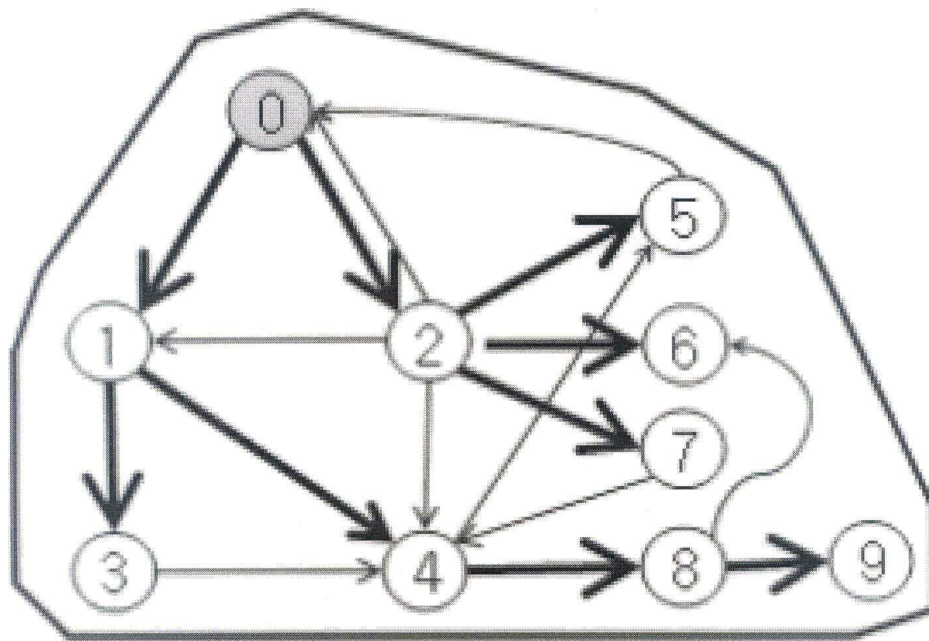
도면1



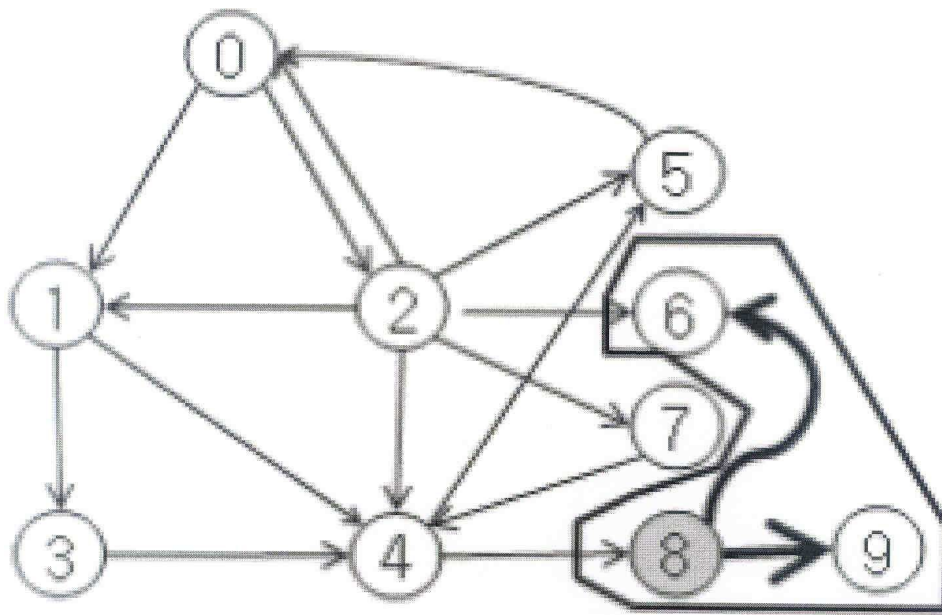
도면2



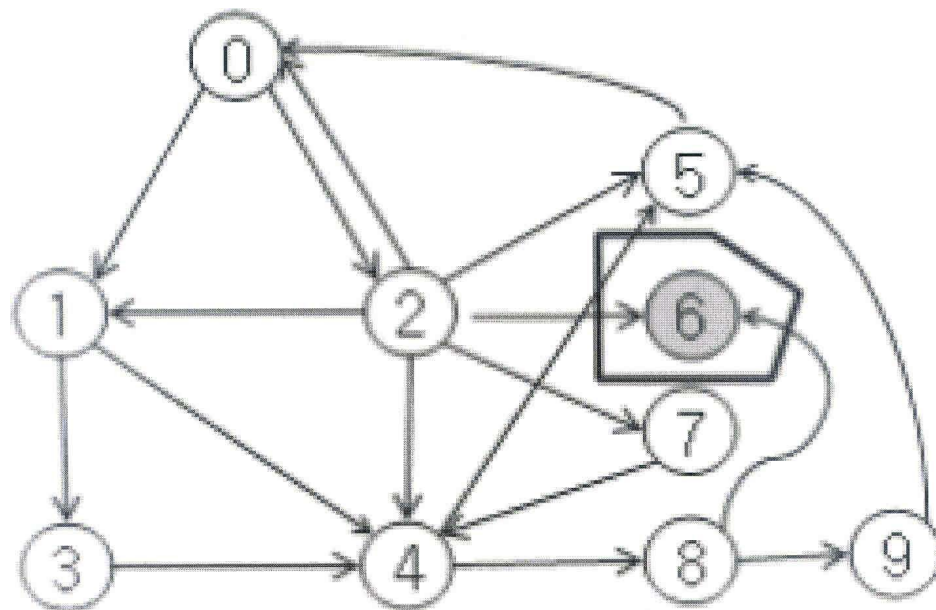
도면3a



도면3b



도면3c



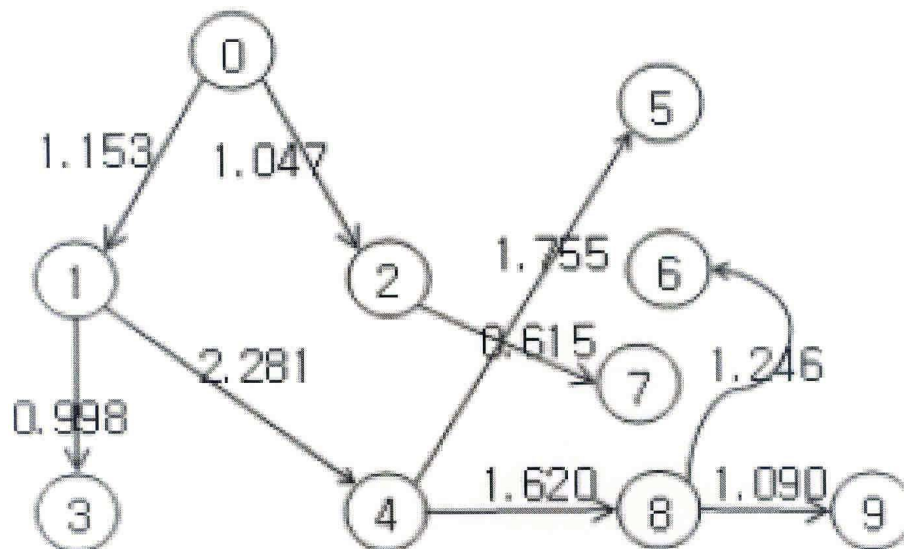
도면4

```

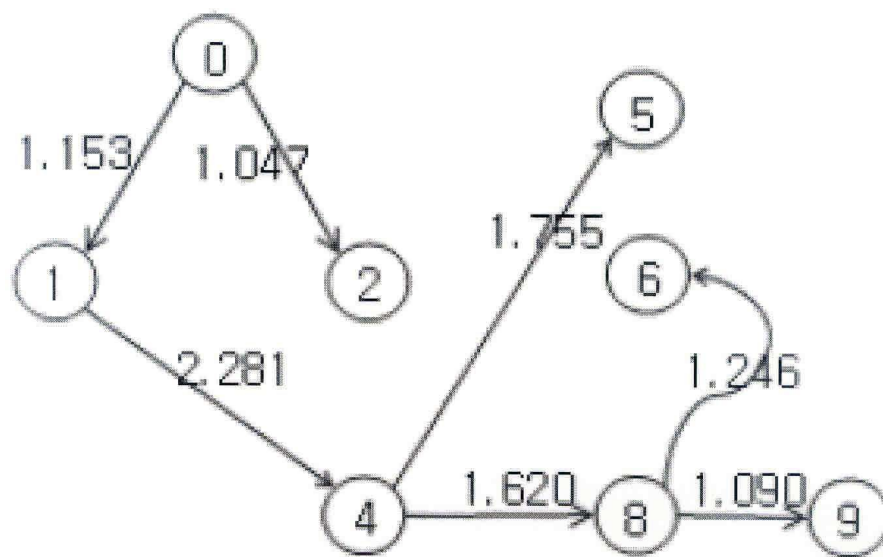
Algorithm: filter_tree
Input: DST,  $f$  output: FST
1.  filter_tree( $n, f$ ) {
2.    if |succ( $n$ )| == 0 then {
3.      return;
4.    }
5.    else {
6.       $C = \text{succ}(n)$ ; // set of the child nodes of  $n$ 
7.      for each  $u$ , where  $u \in C$  {
8.        filter_tree( $u, f$ );
9.        if  $w(n, u) > f$  then
10.       else {
11.         // delete the node  $u$ 
12.         add succ( $u$ ) to  $C$ 
13.       for each  $v$  where  $v \in \text{succ}(u)$ 
14.         create a directed arc from node  $n$  to node  $v$  with  $w(n, v)$ 
15.         delete node  $u$  and directed arc from node  $n$  to node  $u$ 
16.       } // else
17.     } // for
18.   } // if
19. } // filter_tree()

```

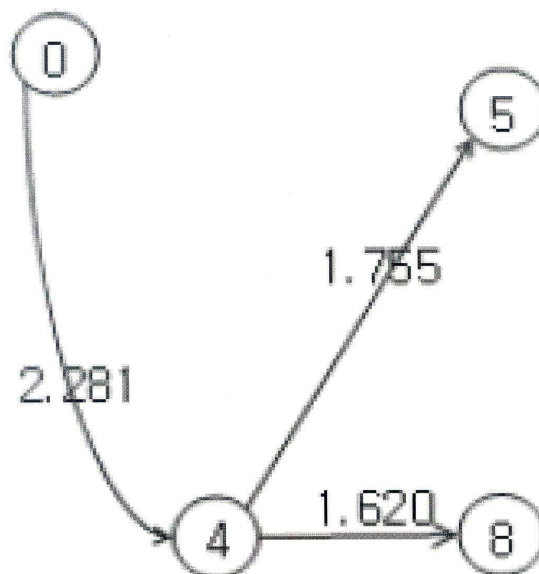
도면5a



도면5b



도면5c



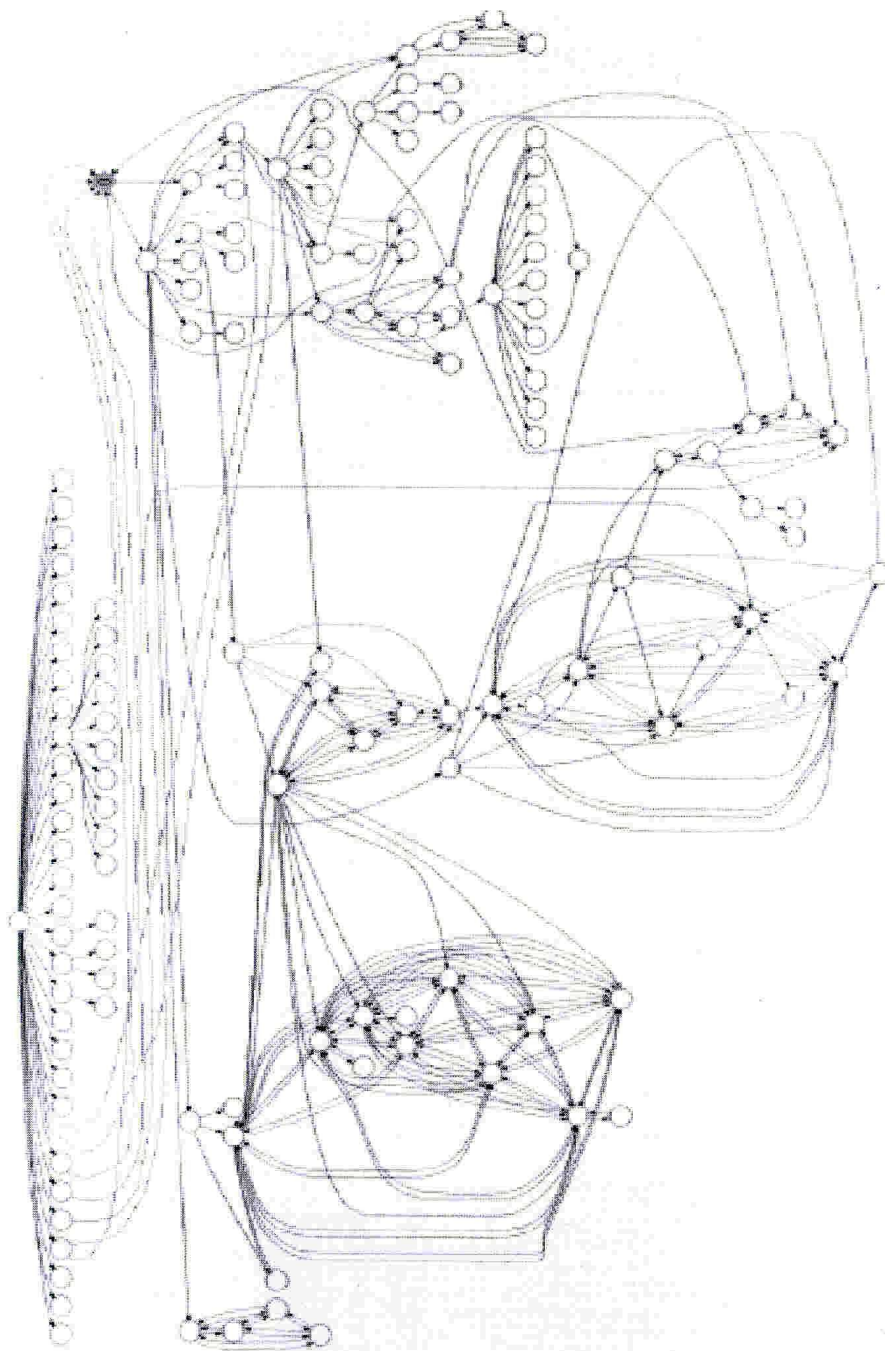
도면6

Algorithm: *top_k_tree*

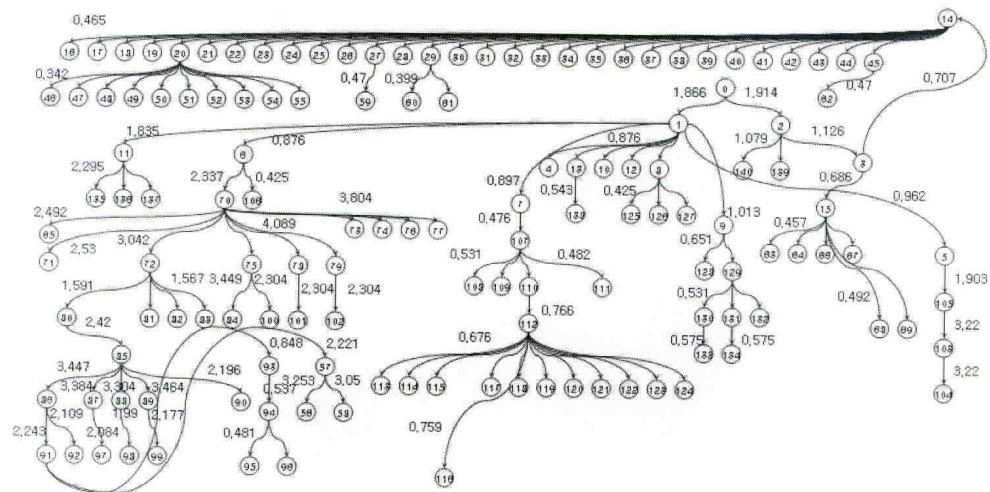
Input: DST, k ; output: FST

1. **top_k_tree**(n, k) {
2. make a list of the directed arcs included in DST, *ArcList*
3. sort *ArcList* in descending order of arc weights
4. select the first k directed arcs from the sorted *ArcList* and make FST with the selected k directed arcs and $k+1$ nodes that are connected by the selected directed arcs including the root node with the same method as *filter_tree* algorithm;
5. Tie-break if there exist multiple arcs within the top k limit.
6. } // **top_k_tree**()

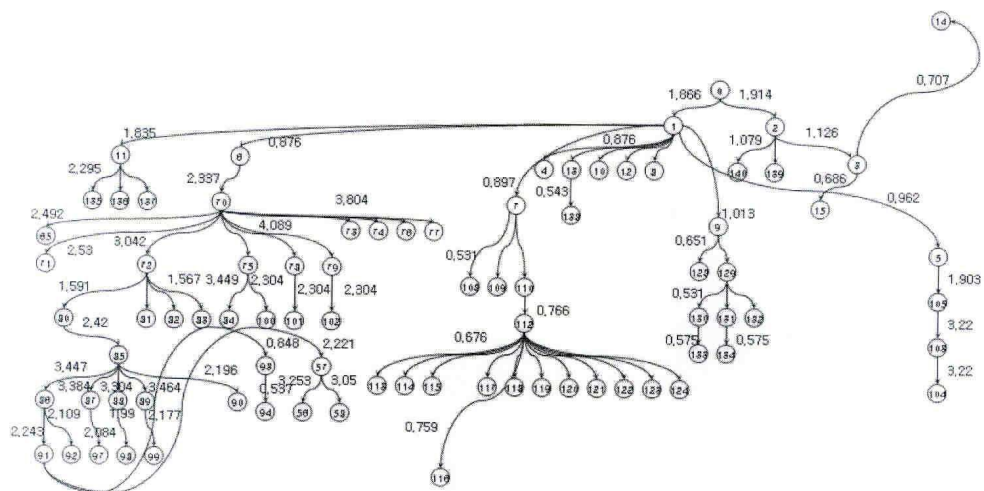
도면7



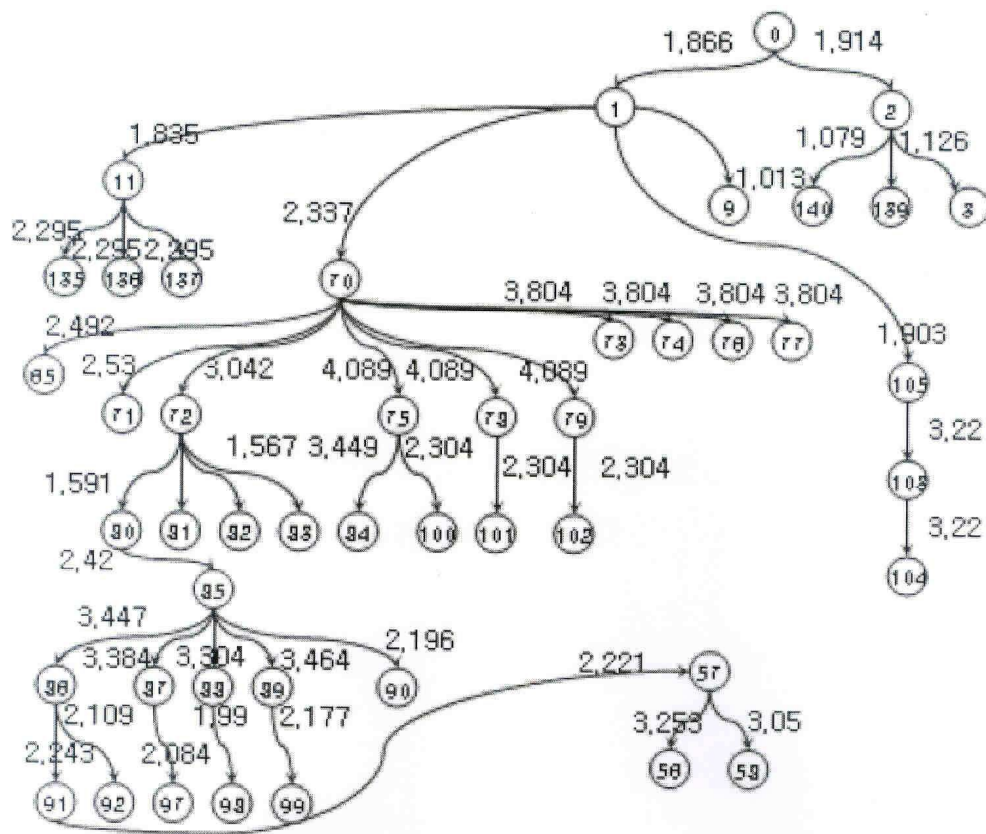
도면8



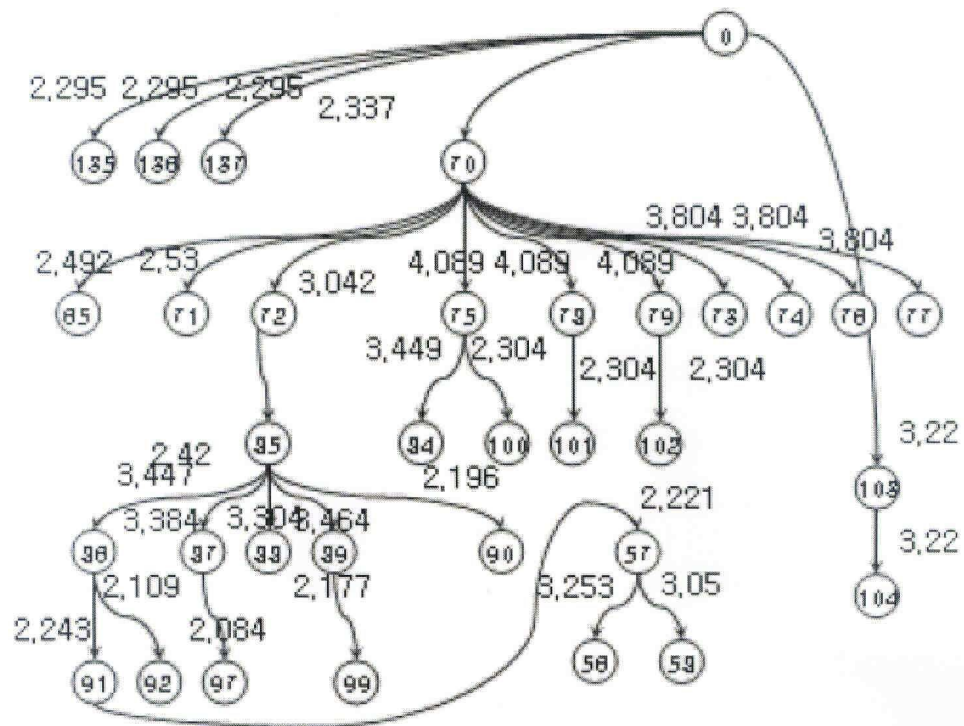
도면9



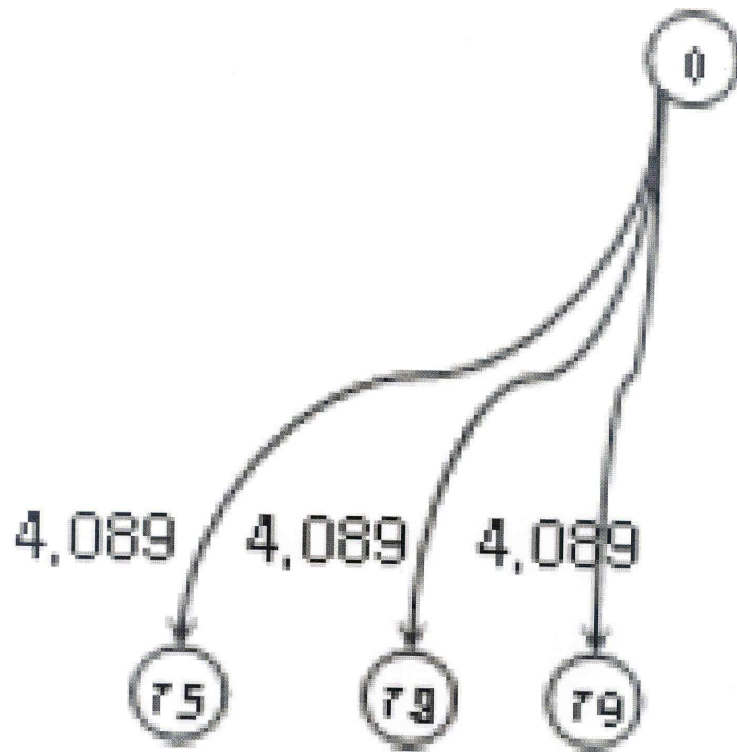
도면10a



도면10b



도면10c

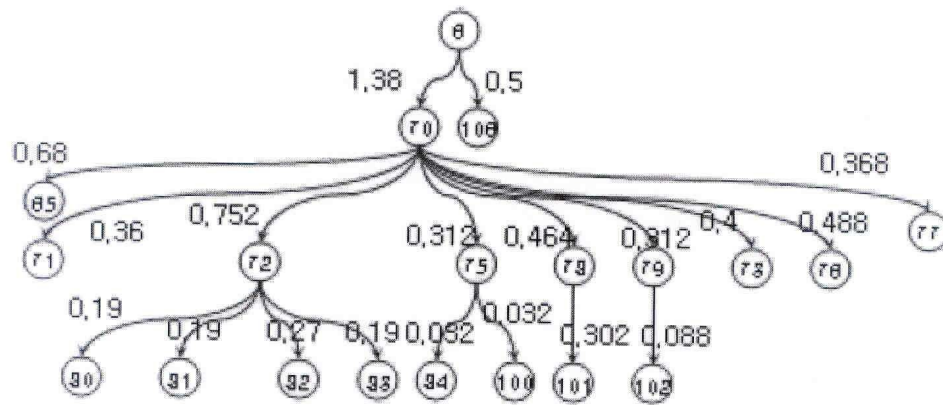


도면11

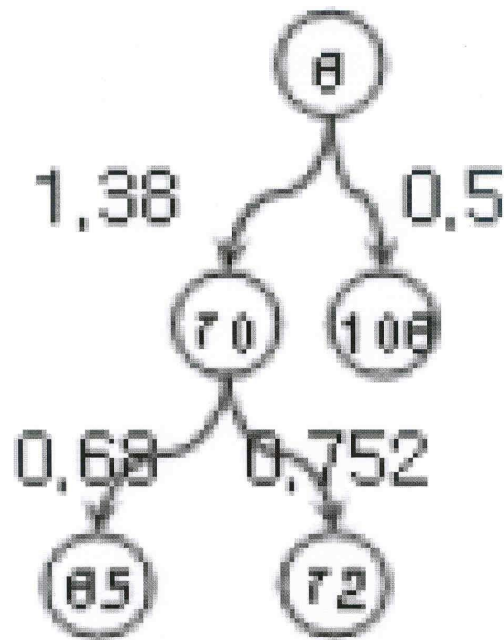
LSU Hurricane Experts
 Effectiveness of Building Codes in Reducing Hurricane Damage: A New Study for Mississippi
 Hurricane Center Appointed to Lead State Levee Investigation PhD and MS Assistantship Enhancements and Supplements
 Available to Incoming Grad Students Here For Hurricane Preparedness Tips last revised on 1/19/2006
<http://hurricane.lsu.edu/>

- ☐ Weather Information for Emergency Management
 Availability of weather information has improved in the last decade due to factors that include improved communication networks; updated observation, forecast, and warning systems;
<http://hurricane.lsu.edu/paper3a.htm>
- ☐ Are Chemical Plants at Risk from Hurricane Winds?
 The two biggest potential impacts from hurricane wind damage are release of hazardous materials into the environment and the economic impact of repairing damaged structures, including loss of production while the repairs are underway.
<http://hurricane.lsu.edu/paper6a.htm>
- ☐ Atmospheric Dispersion with Risk and Probability Estimates in Extreme Weather Conditions
 This paper compares the predicted hazard zones as function of wind speed following two different accidental release scenarios. One of the models uses a heavy gas dispersion algorithm, while the other employs a Gaussian model.
<http://hurricane.lsu.edu/paper7a.htm>

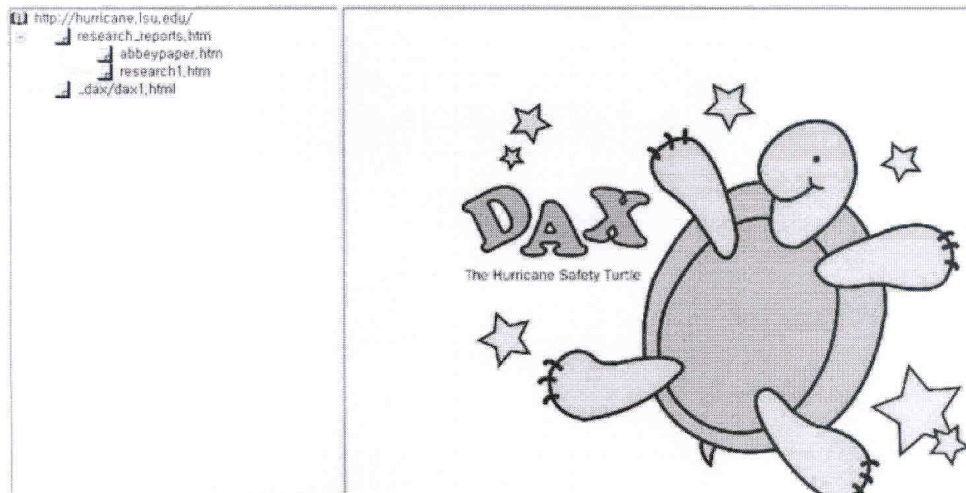
도면12a



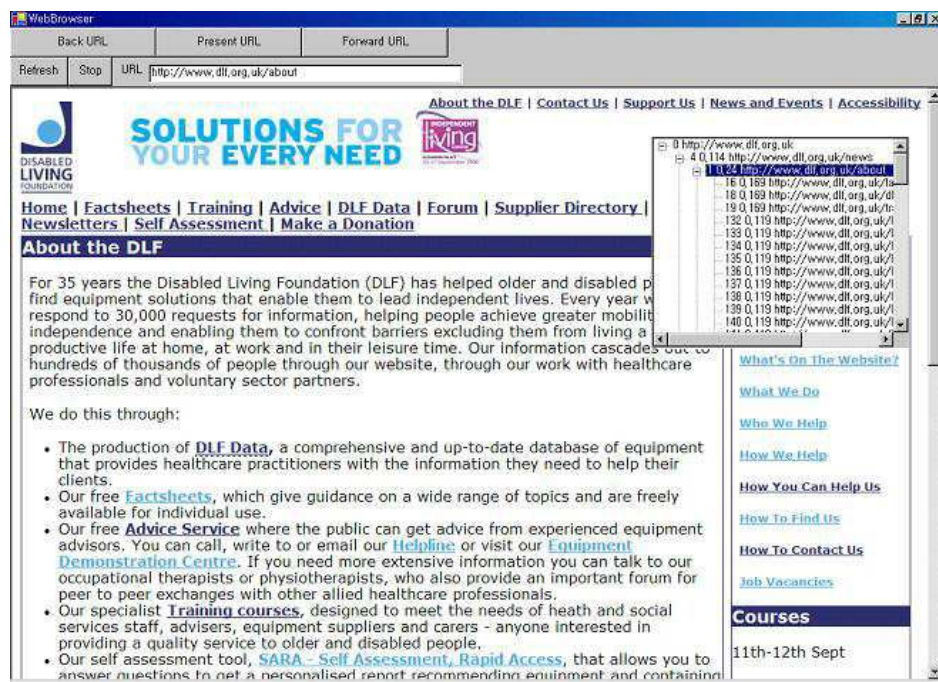
도면12b



도면13



도면14



도면15

