

# 웹2.0의 참여형 아키텍처 환경에서 그래픽 기반 포크소노미 태그 연관 검색의 설계 및 구현

## Design and Implementation of the Graphical Relational Searching for Folksonomy Tags in the Participational Architecture of Web 2.0

김운용\*  
Woon-Yong Kim

박석규\*\*  
Seok-Gyu Park

### 요약

최근 인터넷의 급격한 확장을 통해 대두되는 웹2.0 기술은 웹의 구조적 진화인 질적 변화와 사용자 증가라는 양적인 변화로 해석할 수 있으며, 이 구조는 사용자 참여형 아키텍처를 근간으로 하고 있다. 블로그, UCC, SNS(Social Networking Service), 매쉬업, 롱테일 등으로 대표되는 웹2.0기술은 웹의 구조화에 중심적인 역할을 담당하고 있으며, 이곳에 포함된 사용자 참여형 자료의 분류와 검색은 포크소노미(Folksonomy)방식을 통해 널리 이용되고 있다. 포크소노미는 웹 페이지에 공개되고 있는 정보나 관련 주제를 고전적 분류학 기술로 나누는 것이 아니라 꼬리표(태그)에 따라 구분하는 새로운 분류방식으로 사용자 참여로 구성된다. 현재 이러한 분류를 통한 검색은 단순 텍스트 태그검색이나 태그클라우드 방식 등을 통해 특정 태그에 대한 검색이 이루어지고 있으나 태그들 간의 관계를 표현하고, 이들 관계를 고려한 검색은 미비한 실정이다. 이에 본 논문에서는 등록되는 태그들 간의 관계를 고려하여 태그들 간의 연관 그래프를 동적으로 형성하고 이를 기반으로 연관 검색을 제공한다. 이를 통해 태그 검색의 신뢰성을 높이고 검색의 편의성을 제공할 수 있을 것이다.

### Abstract

Recently, the web 2.0 services which appear by exponential extension of the Internet can be expressed with the changes in the quality of structural evolution and in the quantity of increasing users. The structural base is in user participational architecture. The web 2.0 services such as Blog, UCC, SNS(Social Networking Service), Mash-up, Long tail, etc. play a important role in organization of web, and grouping and searching of user participational data in web 2.0 is broadly used by folksonomy. Folksonomy is a new form that categorizes by tags, not classic taxonomy skill. It is made by user participation. Searching based on tag is now done by a simple text or a tag cloud method. But searching to consider and express the relations among each tags is imperfect yet. Thus, this paper provides the relational searching based on tags using the relational graph of tags. It should improve the trust of the searching and provide the convenience of the searching.

☞ keywords: Web 2.0, Folksonomy, Tag, Participational Architecture, Relational Searching

## 1. 서론

최근의 웹 환경은 인터넷의 급격한 확장을 통해 과거와는 비교도 되지 않을 정도로 방대한 양의 정보를 손쉽게 얻을 수 있을 뿐만 아니라 사

용자의 적극적인 참여를 유도하는 참여형 아키텍처를 지향하고 있으며, 블로그나 SNS, 위키(Wiki) 등이 사용자 참여형 서비스 전반으로 확대됨으로써 UCC(User Created Contents)라는 새로운 미디어를 만들어 내고 있다. 또한 웹의 구조화를 통한 끊임 없는 네트워크를 구성하고 웹서비스 기반의 매쉬업 기술과 롱테일(Long-Tail), Ajax(Asynchronous Java Script And XML), RSS(Really Simple Syndication) 피드 등을 이용해 사용자에게 풍부한 사용자 경험과 편의성을 제공하는 데 많은 노력을 기울이

\* 정 회 원 : 강원도립대학 컴퓨터 응용과 교수  
wykim@gw.ac.kr

\*\* 종신회원 : 강원도립대학 컴퓨터 응용과 교수  
skpark@gw.ac.kr

[2007/04/04 투고 - 2007/04/20 심사 - 2007/06/19 심사완료]

고 있다[1][2][3].

검색서비스 역시 많은 진화를 가져오고 있다. 웹 검색서비스의 개척자 역할을 한 야후에 의한 웹의 디렉토리 서비스[4]로 시작한 검색서비스는 이후 로봇형 검색 서비스를 제공하는 구글의 검색 엔진으로 이어지고 있으며 HTML기반의 웹문서 역시 XML기반의 구조화를 받아들임으로써 더욱 구조적이고 편리한 검색방법으로 진화하고 있다.

그러나 이러한 검색 결과는 타당성을 증명하기 위해 사용자 스스로 확인하는 것 외에는 방법이 존재하지 않는다. 이러한 문제를 해결하는 방법으로 등장한 개념이 소셜 태깅(Social Tagging)에 의한 포크소노미(Folksonomy)방식이다[5]. 이 방식은 블로그, 기사, 그림, 동영상 등과 같은 웹상의 정보에 사용자가 임의의 꼬리표(태그)를 붙이는 방식으로 검색엔진에 의해 인덱스를 만드는 것에 비해 사람의 눈에 의해 이루어지는 인덱스 방식이다. 이 기술은 웹2.0의 참여형 아키텍처를 구성하는 핵심요소로 인식되고 있으며 이러한 태그들은 모든 인터넷 사용자의 참여를 통해 만들어지고 태그와 태그와의 연관성은 사회적인 의미를 갖게 된다.

그러나 다양한 사용자가 참여하여 만들어지는 태그들은 동적인 구조를 가짐으로써 소수의 집단에 의해 가능한 체계적인 분류체계를 형성하는 것이 어렵다. 그 결과 태그 기반의 검색은 텍스트 기반이나 태그 클라우드[6][7]와 같은 방식으로 이루어지고 있다. 이러한 방식은 특정 태그에 대한 단편적인 관련 콘텐츠를 검색하기는 용이하지만 태그와 태그와의 연관성을 고려하여 정확한 결과를 검색하기가 어렵다. 또한 텍스트 기반의 검색은 모바일환경이나 소형 디바이스 환경에서 활용하기에는 어려움이 존재한다.

이러한 문제를 해결하기 위해 본 논문에서는 포크소노미 방식의 태그들을 이용해 태그와 태그간의 연관성을 고려한 연관 그래프의 동적 구성 기법을 제시하고 이를 기반으로 그래픽환경에서 태그간의 관계성을 고려한 효율적 연관 검색 기

법을 소개한다. 이를 통해 태그 검색에 대한 사용자의 편의성을 증대시키고 보다 정확한 관련성을 가진 검색 결과를 얻을 수 있을 것이다.

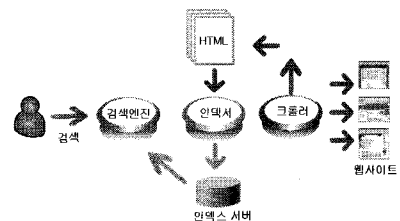
본 논문의 구성은 다음과 같다. 먼저 2절에서 태그 기반의 검색에 대한 관련 연구들에 대해 살펴본다. 3절에서는 사용자 참여형 아키텍처 환경에서 태그간의 동적 연관 그래프 구성 및 태그 연관 검색 기법들을 논의한다. 4절에서는 그래픽 기반의 연관 검색 기법 활용 예를 보인 후에, 5절에서 결론 및 향후 과제로 논문을 마무리한다.

## 2. 관련 연구

### 2.1 전통적 웹 검색 기법

웹 검색은 주로 포털 사이트를 통해 이루어지고 있으며 초기의 웹 검색은 스탠포드대학의 데이빗 파일로(David Filo)와 제리 양(Jerry Yang)으로부터 시작한 웹 디렉토리 서비스 일 것이다. 이러한 방식은 콘텐츠의 주제나 내용에 따라 계층 구조로 된 카테고리로 분류하며 사용자가 카테고리를 따라 원하는 콘텐츠를 검색할 수 있게 하는 택소노미(Taxsonomy)방식의 트리형 계층구조를 지원한다[4]. 이러한 웹 디렉토리 서비스는 사람의 손을 거친 검색사이트로 시작하여 점차 로봇형 검색 사이트로 진화하였다[8][9].

로봇형 검색 서비스는 그림 1에서와 같이 크게 검색대상이 되는 정보를 수집하는 크롤러와 모여진 정보를 정리 보존하는 인덱서 그리고 정리된 정보로부터 적절한 정보를 찾는 검색엔진인 3부분으로 구성된다.



〈그림 2〉 로봇형 검색엔진 구조

또한 이들의 검색 기법은 N-그램과 형태소 해석 방법을 이용한다[10]. N-그램은 텍스트의 글자 하나하나를 최소단위로 하여 N개의 글자를 대상으로 검색대상을 정하는 방법이며, 형태소 해석은 어구의 의미를 생각해서 잘라내는 방식이다. 그러나 이들은 기본적으로 HTML문서와 작성된 텍스트를 바탕으로 이루어지기 때문에 비구조적인 특성과 의도하지 않는 많은 정보가 검색될 수 있다는 한계점을 가진다.

## 2.2 태그 기반 웹 검색

대중에 의한 분류의 의미를 가진 포크소노미는 계층 구조가 아니라 사용자의 주관에 따른 태그라는 메타데이터를 붙여서 분류하는 방식으로 하나의 콘텐츠에 여러개의 태그를 붙일 수 있으며 시간의 흐름에 따라 관련 콘텐츠간의 연관 관계가 항상 변화될 수 있는 구조를 가진다[6][7]. 포크소노미 방식은 XML기반의 사용자 참여형 관련 콘텐츠 증가에 따라 활용도가 증가 되고 있으며 특히 태그 검색 서비스뿐만 아니라 소셜 북마크, 소셜 네트워킹서비스, 사진 공유 서비스 등에서 자주 활용되고 있다[11].

태그 기반의 검색에서, 등록된 태그들은 사회성을 반영한다는 특징때문에 사용자에게 원하는 정보를 보다 정확하게 제공할 수 있다. 이러한 특징 때문에 최근 포크소노미의 활용도를 높이려는 노력들이 활발하게 진행되고 있다. 태그를 이용한 서비스의 대표적인 예로 보다 많은 링크가 되어 있는 태그의 텍스트를 크게 표시하는 태그 클라우드(Tag cloud)형태의 시각적인 방식이 있다[6]. 그림 2는 이러한 태그 클라우드 예를 보여준다.

이러한 기존 태그 검색은 콘텐츠에 등록된 태그들을 직접적으로 검색하고 있거나, 빈번하게 사용되는 태그들을 기록함으로써 사용자들에게 자주 사용하는 태그들을 쉽게 접근할 수 있는 태그 클라우드 방식으로 운영되고 있다. 이는 특정 태그 검색에 대해서는 유용하지만 태그와 태그간의



〈그림 3〉 태그 클라우드 활용 예

연관성을 고려한 검색처리는 비효율적이다. 예로 검색어 ‘웹서비스’로 검색 후 보다 세부적인 검색 대상 ‘자바’을 태그를 검색할 경우 사용자가 웹서비스를 포함하고 자바로 작성될 수 있는 내용을 보고자 할 의도 일지라도 이들 웹서비스와 자바의 연관성을 고려한 검색을 수행하지 않는다. 또한 중복 검색을 통해 ‘웹서비스, 자바’로 검색할 경우 해당 두 태그를 포함한 요소는 검색할 수 있으나 이들 역시 전체 콘텐츠를 대상으로 수행되므로 검색의 효율성이 떨어진다. 그러므로 사용자들에게 보다 효과적인 검색을 위해서는 검색 결과를 바탕으로 이후 검색에 영향을 미칠 수 있는 검색 방법이 필요하다.

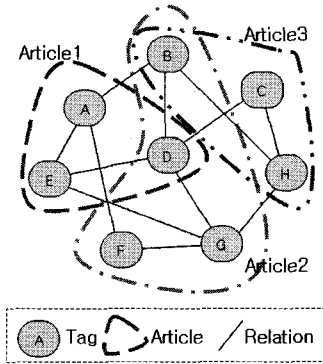
이를 위해 3절에서는 사용자 참여형 구조에서 태그들 간의 연관성을 고려한 검색 기법을 제시한다. 이를 통해 사용자에게 보다 높은 편리성과 접근성을 제공하고 신뢰성 있는 검색 결과를 얻을 수 있을 것이다.

## 3. 동적 태그 연관 그래프를 이용한 연관 검색 기법

이절에서는 태그들 간의 연관 검색을 제공하기 위해, 먼저 사용자에게 의해 만들어지는 태그들을 이용해 동적 연관 관계를 고려한 연관 그래프 형성 기법을 다룬 후 이 생성된 그래프를 기반으로 태그들 간의 연관검색 기법을 제시한다.

### 3.1 연관성을 고려한 동적 태그 연관 그래프 구조

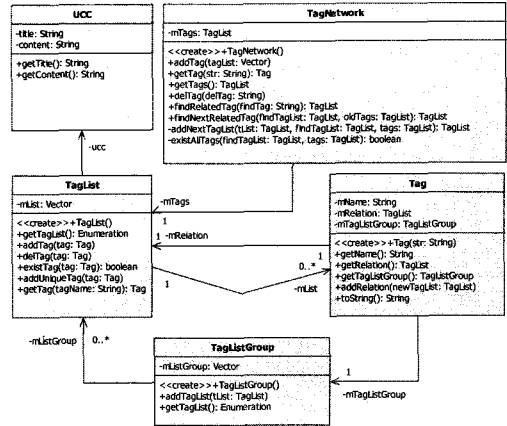
사용자에 의해 만들어지는 태그들은 사회적인 성격을 반영하고 있기 때문에 전통적인 웹 검색 방법에 비해 더욱 정확한 검색을 제공할 수 있다. 또한 이들 블로그, 기사, 그림, 동영상과 같은 정보검색을 위해 제공되는 태그들은 각각 하나 이상으로 구성될 수 있으며, 언제든지 추가되거나 변경될 수 있는 특징을 가진다. 그리고 하나의 특정 정보에 존재하는 태그들은 서로간의 밀접한 연관성을 가질 뿐만 아니라 다른 정보에 제공되는 태그들과도 연관성을 가진다. 그러므로 이들 간의 관계를 고려하여 태그들 간의 연관 그래프를 구성할 필요성이 존재한다.



(그림 4) 태그들의 연관성

위의 (그림 3)은 일반적인 포크소노미 환경에서 태그와 콘텐츠 그리고 그들 간의 연관성을 보여준다. 하나의 콘텐츠는 그 내용을 설명하기 위해 하나 이상의 태그들로 구성되고, 그들 태그들은 서로 간에 밀접한 연관성을 가진다. 이들 관계는 크게 하나의 정보에 소속된 태그들의 집합성 관계와 다양한 정보에 동일한 태그를 가진 동일성 관계로 나눌 수 있다. 이러한 관계성을 고려한 태그 연관 그래프 구조는 (그림 4)와 같다.

(그림 4)의 클래스 다이어그램은 태그간의 연관성을 표현하고 있다. 태그 네트워크(TagNetwork)는 사용자들에 의해 만들어지는 태그들을 관리하기 위해 태그리스트(TagList)을 포함하고 있으며 이 태그리스트(TagList)는 하나 이상의



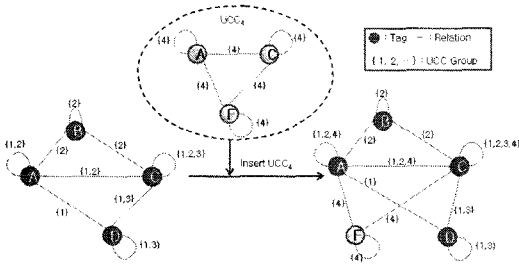
(그림 5) 태그 간의 연관성을 고려한 그래프 구조

태그(Tag)들을 포함한다. 또한 각 태그(Tag)는 자신과 연관된 태그들의 집합을 관리하기 위해 태그리스트(TagList)를 포함하고 있으며, 자신이 속한 콘텐츠단위의 태그 정보를 관리하는 태그리스트 그룹(TagListGroup)을 가진다. 이 태그리스트 그룹(TagListGroup)은 각 콘텐츠 단위 태그리스트(TagList)정보를 포함하고 있다. 또한 태그리스트(TagList)는 자신이 속한 콘텐츠(UCC)정보를 관리한다. 이렇게 구성된 태그간의 연관 그래프는 각 태그들의 집단성과 동일성간의 연관관계를 표현한다.

포크소노미 환경에서 태그들은 사용자들에 의해 빈번하게 추가되고 수정되어진다. 그러므로 태그들 간의 연관 그래프는 이러한 특성을 반영하여 재구성될 수 있어야한다.

### 3.2.1 태그 연관 그래프에 새로운 사용자 생성 콘텐츠 추가에 따른 태그 추가

참여형 아키텍처 구조에서 사용자 생성 콘텐츠는 언제든지 추가될 수 있다. 이렇게 추가된 콘텐츠는 그 내용을 식별할 수 있는 태그 리스트를 포함한다. 그 결과 이 태그 리스트들을 기존 태그 연관 그래프에 반영시켜야한다. 일반적인 태그 리스트 추가과정은 다음과 같다.

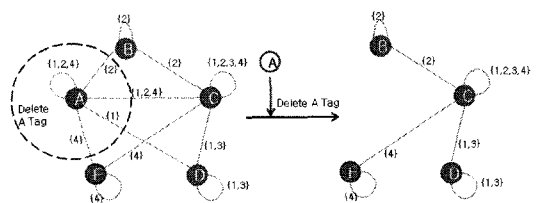


〈그림 6〉 콘텐츠 추가에 따른 태그 추가 개념도

(그림 5)는 UCC에 포함된 태그들을 태그 연관 그래프에 추가하는 개념을 보여주고 있다. 새로운 콘텐츠는 자신의 태그리스트를 가지고 기존의 그래프에 추가될 때, 동일 태그는 추가되지 않으며 단지 기존 태그와 해당 UCC가 새롭게 연관관계를 형성한다. 반면 기존에 등록되지 않는 태그는 태그리스트에 새롭게 추가되며 이들 역시 UCC와 연관성을 유지한다. 이러한 개념을 이용해 (그림 4)에서 제시된 그래프 구조를 기반 한 새로운 UCC 추가 알고리즘은 다음과 같다.

### 3.2.2 기존 태그 연관 그래프에서 특정 태그 삭제

태그 연관 그래프에서 특정 태그 삭제 역시 사용자 참여형 구조에서 빈번하게 발생하는 현상이다. 특정 태그를 삭제하기 위해서는 먼저 삭제 대상이 되는 태그와 연관성을 가진 태그들을 검색한 후 이들과의 연관관계를 제거한 후 자신을 삭제하는 방식으로 이루어진다. 일반적인 태그 삭제 과정의 개념도는 다음과 같다.



〈그림 7〉 태그 연관 그래프에서 특정 태그 삭제 개념도

특정 태그의 삭제는 이 태그를 포함한 UCC의 태그들을 삭제하는 과정이다. 이러한 과정을 수행하기 위해 먼저 삭제 대상 태그와 연관된 UCC를

```

class TagNetwork
  local mTag : TagList
  feature {ANY} - Public Commands
  addTag(tagList:Tags) is
    - Add new the set of Tag on UCC into Tag Network
    require tagList - The new set of tags on UCC
    do
      from tagList.getNextTag() until tagList.getNextTag()==null
      loop
        - Find a same Tag in mTag with the tag that found from tagList
        if not find tag then
          - Create new Tag for a tag searched from tag list
          - Add the new Tag into mTag for the Tag Network
        end if
      end loop
      from mTag.getTagList() until mTag.getTagList() == null
      loop
        - Add relation of Tags from the new tagList into mTag
      end loop
    end do
  
```

〈그림 6〉 새로운 UCC 추가에 따른 태그 리스트 추가 알고리즘

```

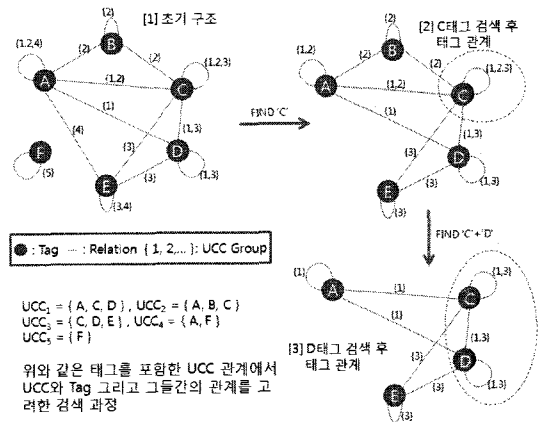
class TagNetwork
  local mTag : TagList
  feature {ANY} -- Public Commands
  delTag(delTag: the name of Tag) is
    - Delete tag and tag relation from Tag Network
  require delTag -- The tag name that is deleted from the tag network
  do
    dTag := getTag(delTag) --Find the Tag from mTag for deletion
    rTags := dTags.getRelation() --Find the relational tags from dTag
    from rTags.getTag() until rTags.getTag()==null
    loop
      rTagList := tag.getRelation() --Get the relational tag list from the tag
      rTagList.delTag(dTag) --Delete relation from the relational tag
      - get UCC group included the tag
      -- delete UCC
    end loop
    mTags.delTag(dTag) -- Delete dTag from mTags (Tag Network)
  end do
  
```

〈그림 8〉 태그 연관 그래프에서 특정 태그 삭제 알고리즘

검색할 필요가 있다. 논문에서 제시된 연관 그래프에서 이러한 과정은 특정 태그와 연관 태그를 검색함으로써 가능하다. 연관 태그들은 자신이 포함된 그룹을 관리하고 있으며 이 그룹에서 제거할 대상의 태그 관계를 삭제한다. 모든 연관성 관계를 제거한 후 삭제대상 태그가 삭제된다. 이러한 과정의 알고리즘은 다음과 같다.

### 3.2 태그 연관 그래프를 이용한 연관 검색 기법

태그 간의 연관성은 동적으로 유지되기 때문에 태그간의 디렉토리 구조를 정적으로 형성하기가 어렵다. 이러한 문제를 해결하기위해 본 논문에서는 사용자의 검색 과정에서 선택된 태그를 기반으로 동적 디렉토리 구조를 형성한다. 이 과정은 선택된 태그의 연관관계를 고려하여 이루어진다. 그 결과 사용자는 특정 상황에 적합한 태그들의 연관 검색을 가능하게 한다. 이러한 태그 검색의 개념도는 아래와 같다.



〈그림 9〉 태그 연관 그래프를 이용한 연관 검색 개념도

태그 연관 그래프에서 태그 연관 검색은 UCC에 포함된 태그들을 기반으로 도달 가능한 태그들을 추적하고 이후 검색결과에 필요한 태그들을 구성한다. 이를 통해 다음 연관검색 시 불필요한 검색 과정을 줄일 수 있다. 그림 9는 다음과 같은 가장아래에서 C 태그 검색 후 D 태그 연관 검색 과정을 보여준다.

UG : UCC 그룹 (사용자 정의 콘텐츠 그룹),  
 TG : Tag 그룹 (UCC에 포함되어 있는 태그 리스트)로 정의할 때, 초기 UCC 그룹은  $UG = \{UCC_1, UCC_2, UCC_3, UCC_4, UCC_5\}$  이고 태그 그룹은  $TG = \{A, B, C, D, E, F\}$ 이다. 또한 각각의 UCC는  $UCC_1 = \{A, C, D\}$ ,  $UCC_2 = \{A, B, C\}$ ,  $UCC_3 = \{C, D, E\}$ ,  $UCC_4 = \{A, E\}$ ,  $UCC_5 = \{F\}$  태그를 포함한다. 태그 네트워크 구조는 태그와 태그 간의 연관성을 나타내는 선들 그리고 각각의 선들이 어떤 UCC와 관련성을 제공하는지를 나타내는 {} UCC집합으로 구성된다.

이때 최초 검색태그 C는 {1,2,3}의 UCC와 연관성을 가진다. 이때 연관 그래프에서 이들 {1,2,3}과의 연관성을 가진 연결선을 제외한 나머지를 삭제(1,2,3중 어느 하나라도 연관성을 가진다면 태그간의 연관 선은 지속된다.) 하면 그림 9의 [2]의 구조와 같다. 그 결과 태그 F는 도달 불가능한 태그로 다음 검색대상에서 제외된다. 이후

D태그의 검색 시 D 태그는 {1,3}의 UCC와 연관성을 가지므로 첫 번째 검색 후 구성된 태그 연관 구조에서 이들을 포함하는 연결선을 제외한 나머지 선들을 제거하면 그림 9의 [3]과 같이 표현된다. 이때 B 태그는 더 이상 접근 불가능한 태그가 되며 다음 검색대상에서 제외된다. 정규화된 표현식은 다음과 같다.

$$UG = \sum_{i=1}^n UCC_i$$

$$UCC = \sum_{i=1}^n T_i, T = Tag \in UCC$$

$$TG = \bigcup_{i=1}^n T_i, (T_i \in UCC)$$

$$Find(x) = UG(x) = \sum_{i=1}^n UCC_i, (x \in UCC_i)$$

$$Find(x+y) = Find(x) \cap Find(y)$$

$$Find(x+y) = Find(x) \cap \sum_{i=1}^n UCC_i, (y \in UCC_i)$$

이를 통해 태그 검색과정은 불필요한 태그 검

```

class TagNetwork
  local mTag : TagList
  feature {ANY} -- Public Commands
  TagList findRelatedTag(findTag:String, oldTags:TagList) is
    -- Find the next relational search list from tag lists
  require findTag -- The tag name for search to the tag network
         oldTags -- The searched tag list before this search
  do
    if is_first_search then
      from mTag.getTagList() until mTag.getTagList()==null
      loop
        -- Find the next search list from tag network (mTag)
      end loop
    else
      from oldTag.getTagList() until oldTag.getTagList()==null
      loop
        -- Get the tag list group included the tag
        -- Find the each tag from the tag list group
        -- Include into the next search list the each tag except the tag that
           exist the previous tag list
      end loop
    return The next search list
  end do
  
```

<그림 10> 태그 연관 그래프를 이용한 연관 검색 알고리즘

색 기회를 줄임으로써 사용자의 의도에 적합한 콘텐츠를 얻을 수 있도록 한다. 태그 연관 검색 알고리즘은 다음과 같다.

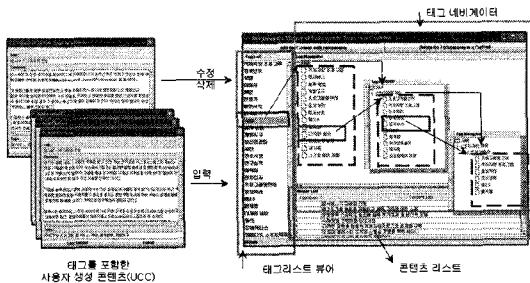
태그 연관 검색은 특정 태그 검색 시 사용자 참여 콘텐츠에 등록된 태그들을 기반으로 형성된 연관 그래프를 이용하여 특정 태그와 연관된 태그 리스트를 검색한 후 사용자에게 다음에 검색 가능한 태그 대상을 알려줌으로써 검색의 신뢰성과 효율성을 증대시킬 수 있다. 검색 방법은 먼저 최초 태그 검색 시 태그 연관 그래프에 포함된 검색 대상 태그의 연관 태그 리스트를 제공함으로써 이루어진다. 이후 검색은 이전 검색 결과에 기반을 두고 반복적으로 이루어진다.

#### 4. 그래픽 기반 태그 연관 검색 활용

그래픽 기반 사용자 인터페이스는 디바이스의 소형화와 모바일화를 추구하는 유비쿼터스 환경에 적합한 구조를 가진다. 검색과정 역시 텍스트 기반의 사용자 검색에 비해 보다 효율적인 접근성과 편의성을 제공할 수 있을 것이다. 검색과정의 핵심은 등록된 사용자 생성 콘텐츠를 빠짐없이 쉽게 검색가능하고 불필요한 검색과정을 줄이는데 있다. 여기서 제시되는 프로토타입은 3절에서 제시된 태그 연관 그래프를 이용한 연관검색을 바탕으로 구성하고 있으며 이러한 구성형태는 다음과 같다.

되어지고 이들은 필요에 따라 추가 및 수정 삭제가 이루어진다. 그림 11에서 태그 리스트 뷰어는 중복 성을 배제한 등록된 사용자 생성 콘텐츠의 태그 리스트를 보여준다. 이 태그 리스트에서 특정 태그를 선택 시 태그 네비게이터에 선택 태그가 나타나며 그와 연관된 태그 리스트들이 디렉토리 구조로 표현된다.

사용자는 이후 연관 태그 리스트에서 추가적인 특정 태그를 선택할 수 있으며, 이 선택된 추가 태그를 기반으로 연관성을 가지고 도달 가능한 태그 리스트를 다음 선택가능 태그 집합으로 만드는 과정을 반복한다. 그림 11의 예는 ‘자바’라는 태그 선택 시 태그 연관 그래프를 통해 해당 태그와 연관된 리스트를 가져와 ‘자바’태그를 부모노드로 하여 연관된 리스트를 자식노드들로 하여 트리 구조를 형성하고 있다. 이후 검색은 이들 자식 노드에서 ‘자바’와의 연관성을 포함하면서 추가적인 태그를 선택할 수 있다. 이후 ‘디자인 패턴’을 선택할 경우 이전 검색 결과의 태그 그래프를 대상으로 ‘디자인 패턴’ 태그를 포함한 태그들만 검색하여 구성함으로써 태그들 간의 연관 검색이 가능하게 된다. 이때의 검색결과가 콘텐츠 리스트에 표현됨으로써 사용자는 자신이 원하는 태그에 대한 보다 세밀하고 신뢰성 있는 검색이 가능하게 된다. 이렇게 등록된 콘텐츠를 기반으로 사용자의 검색 요구에 대한 태그 간의 디렉토리 구조를 동적으로 생성함으로써 사용자에게 검색의 효율성을 제공할 수 있을 뿐만 아니라 신뢰성 있는 결과를 얻을 수 있을 것이다.



〈그림 11〉 그래픽 기반 태그 연관 검색 과정

사용자 생성 콘텐츠는 태그들을 포함하여 입력

#### 4.1 태그 검색 기법 비교

제시된 태그 기반 연관검색 비교를 위해 일반 태그검색과 태그 클라우드 방식 그리고 제안된 형태의 특징을 분석하면 다음과 같다. 일반 태그나 태그 클라우드방식은 단어 검색을 통해 전체 콘텐츠를 대상으로 매번 검색을 수행하지만 제안된 방식은 이전 결과에 대한 연관검색을 수행함



〈표 1〉 태그 검색 기법 비교

비교대상	일반 Tag 검색	Tag 클라우드 기법	제안된 기법
검색방식	단어검색 다중 단어 검색	단일어 검색	이전결과에 대한 연관 검색
그래픽 표현	텍스트 기반	태그 클라우드 형식 (빈도수 나 참여도에 따른 텍스트 크기 변화)	태그들의 연관성을 고려한 Tree 구조
검색 시간	매번 검색 시 전체 콘텐츠 대상	매번 검색 시 전체 콘텐츠 대상	연관 그래프를 통해 해당 태그와 연관 선을 가진 대상 검색, 이후 검색 시 이전 검색 결과를 대상으로 검색 수행
초기 구축 비용	없음	사용빈도수 나 참여도 기록	연관성을 고려한 태그 연관 그래프 구성
신뢰성	연관성 부재를 통한 검색 결과 불필요한 내용 포함	연관성 부재를 통한 검색 결과 불필요한 내용 포함	+태그간의 연관성 고려를 통한 불필요한 태그 검색 방지

으로써 불필요한 검색대상을 줄일 수 있다. 또한 텍스트 기반의 형태에서 트리 형태의 그래픽 기반으로 운영함으로써 사용자 들이 쉽게 태그 연관성을 검색할 수 있도록 한다. 이들의 특징은 표 1과 같다.

## 5. 결론 및 향후과제

최근 인터넷의 급속한 진화와 함께 이슈가 되고 있는 웹2.0 기술의 근간은 사용자 참여형 아키텍처라 할 수 있을 것이다. 또한 이러한 참여형 구조에 포함된 콘텐츠의 분류와 검색은 포크소노미 방식이 널리 이용되고 있다. 포크소노미는 고전적 분류방식 보다는 태그에 따라 구분하는 새로운 방식으로 사용자 참여를 통해 이루어진다. 현재 이러한 분류를 통한 검색은 단순 텍스트나 태그클라우드 방식으로 이루어지며 이들 태그 간의 관계를 고려한 검색이 미비한 상태이다. 본 논문에서는 사용자에게 의해 등록된 태그간의 연관성을 고려한 동적 연관 검색 기법을 제시하였다. 이를 위해 태그들 간의 연관 그래프를 형성하고, 태그의 등록 및 삭제의 동적관리기법과 이를 기반으로 하는 태그 연관 검색기법을 제시하였다. 이를 통해 등록된 태그간의 신뢰성 있는 검색을 제공하고 사용자의 편의성을 증대시킬 수 있을 것

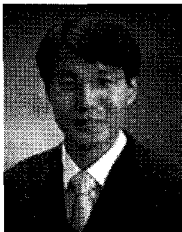
이다. 향후 태그 사용 빈도수를 고려한 연관 검색 기법 및 온톨로지 기반의 의미 분석을 이용한 연관 검색 과 활용에 대한 연구가 필요하다.

## 참 고 문 헌

- [1] Anderson, Chris, The Long Tail: Why the Future of Business is Selling Less of More. Hyperion, 2006
- [2] Jesse James Garrett, Ajax: A New Approach to Web Applications, <http://www.adaptivepath.com/publications/essays/archives/000385.php>, 2005
- [3] O'Reilly, Tim, What is Web 2.0, O'Reilly.com. O'Reilly Media, Inc. <http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html>, 2005
- [4] Rowley, J., Organizing Knowledge, 2nd Ed. Brookfield, VT:Gower, 1995
- [5] Scott A. Golder and Bernardo A. Huberman, Using Patterns of Collaborative Tagging System, Journal of Information Science, 32(2).198-208, 2006
- [6] Hassan-Montero, Y., Herrero-Solana, V. Improving Tag-Clouds as Visual Information Retrieval Interfaces, International Conference on

- Multidisciplinary Information Sciences and Technologies, InSciT2006, Media, Spain, October pages 25-28, 2006
- [7] Owen Kaser and Daniel Lemire, Tag-Cloud Drawing: Algorithms for Cloud Visualization, Tagging and Metadata for Social Information Organization, WWW 2007, May8-12, 2007
- [8] Edwards, J., McCurley, K. S., and Tomlin, J. A. "An adaptive model for optimizing performance of an incremental web crawler". In Proceedings of the Tenth Conference on World Wide Web: 106-113, 2001
- [9] Shkapenyuk, V. and Suel, T., Design and implementation of a high performance distributed web crawler. In Proceedings of the 18th International Conference on Data Engineering (ICDE), pages 357-368, San Jose, California. IEEE CS Press, 2002
- [10] Christopher D. Manning, Hinrich Schütze, Foundations of Statistical Natural Language Processing, MIT Press, 1999
- [11] Laura Gordon-Murnane, Social Bookmarking, Folksonomies, and Web 2.0 Tools, Searcher, Vol. 14, Iss. 6, pages 26-39, Jun 2006

## ○ 저 자 소 개 ○



### 김 운 용(Woon-Yong Kim)

1999년 광운대학교 전자계산학과 졸업 (석사)  
2003년 광운대학교 컴퓨터과학과 졸업 (박사)  
2004년~2005년 Maxsoft 기술연구소 책임연구원  
2006년~현재 강원도립대학 컴퓨터응용과 교수  
관심분야 : OOP, 분산컴퓨팅, 웹2.0, 웹서비스, 임베디드 소프트웨어  
E-mail : wykim@gw.ac.kr



### 박 석 규(Seok-Gyu Park)

1988년~2001년 진주산업대학교 전산실장  
1992년 경남대학교 컴퓨터공학과 졸업 (석사)  
2005년 경상대학교 컴퓨터과학과 졸업 (박사)  
2001년~현재 강원도립대학 컴퓨터응용과 교수  
관심분야 : 소프트웨어 신뢰성, 시스템 분석 및 설계, 멀티미디어  
E-mail : skpark@gw.ac.kr