블로그에서 태그 그룹화를 이용한 트리형 Tag cloud 모델 설계 및 구현

최석순*
*고려대학교 컴퓨터정보통신대학원
e-mail : gildong0@gmail.com*

Design and Implementation for Tree Tag cloud model using tag grouping in blog

Seok-Soon Choi*
*Graduate School of Computer Information and Communication, Korea University

요 약

웹사이트의 블로그에서 등록된 게시물을 분류, 표현하는 방식으로 카테고리 분류방식과 Tag cloud 분류방식을 사용하고 있다. 그러나 카테고리분류방식은 같은 게시물이라도 블로그 관리자별로 해당 분류의 생성기준이 주관적인 판단에 따라 다른 분류에 속할 수 있어 이용자들이 찾고자 하는 게시물을 검색하는데 많은 시간이 소요될 수 있다는 단점이 있다. 또한 이를 보완하는 방안으로 사용되는 Tag cloud 방식은 태그들을 흩어놓아 원하는 정보를 빠르게 찾는데 한계가 있다. 이에 본 논문은 블로그에서 태그들을 그룹화하여 구현한 트리형 Tag cloud(이하 'TreeTag cloud') 모델을 통해 카테고리 분류방식의 트리 구조의 장점인 직관성 및 구조화와 Tag cloud 분류방식의 장점인 짧은 search depth 를 결합하여 구현하는 방법을 제안하였다.

1. 서론

2004 년 10 월 실리콘 밸리에서 개최한 WEB 2.0 컨퍼런스에서 오라이얼리의 Tim O'relly 가 WEB 2.0 이라는 용어를 사용한 이후로 인터넷은 급격히 WEB 2.0 으로 흐르고 있다[1]. WEB 2.0 은 Open, personality 이라는 큰 키워드를 가지고 현재 인터넷을 통해 급격히 변화하고 있다. 이 WEB 2.0 환경에서 블로그의 영향력은 점점 더 커지고 있는데 1 인 미디어로서의 블로그는 일상적인 글에서부터 전문적인 글까지 다양한 내용을 다루고 있으며[8] 또한 블로그는 서로 연결되어 다른 사람과 쉽게 컨텐츠를 공유할 수도 있고, 내용 또한 자주 업데이트가 된다[9]. 이러한 블로그에서 등록된 컨텐츠 분류방식으로 태그를 활용하는 방식의 사용 빈도가 늘어나고 있는데 그 중 Tag cloud 가 태깅된 정보들을 분류, 표현하는데 보편적으로 사용되고 있다.

블로그 서비스가 막 시작되는 시기에 게시물을 등록할 때 등록자가 이미 생성한 그룹항목 중 게시물 내용의 성격에 부합되는 항목을 선택하도록 하는 방식으로 카테고리 분류방식을 대부분 사용되어 왔다. 그러나 이 분류방식은 해당 컨텐츠를 생성하는 생성자의 주관적인 판단에 따라 서로 다른 그룹에 묶일 수 있어 컨텐츠 생성자와 다른 관점에서 컨텐츠의 성격을 판단하는 이용자에게는 찾고자 하는 게시물을 검색하는데 많은 시간이 소요될 수 있다는 단점이 있다. 이에 이를 개선하는 분류방식으로 온라인 쇼핑몰인 amazon.com, 온라인 소셜 북마크 사이트 del.icio.us 를 비롯하여 blogger.com, 테터툴스 등의 블로그 서비스 등 최근 많은 웹사이트와 블로그에서 구현되고 있는 새로운 개념의 분류방식으로 Tag cloud 분류방식이 있다[3][5]. 이 방식은 게시물을 등록할 때 등록자가 내용과 관련 있는 단어들을 기입하도록 하고 이용자는 이 단어들의 목록을 이용하여 찾고자 하는 게시물을 찾을 수 있다. 즉, search depth 를 0 로 줄임으로써 이용자들이 추가 작업 없이 원하는 게시물을 빠르게 찾을 수 있도록 하는 방식이다. 그러나 컨텐츠 별로 등록된 태그들이 중복되지 않고 별개의 태그로 구성되는 경향이 짙어질수록 오히려

태그 리스트에 표현할 태그가 많아져 도리어 이용자가 원하는 태그를 찾는데 어려움이 있을 수 있다는 문제가 있다. 이에 본 논문에서는 카테고리 방식의 특성 중 하나인 트리구조와 Tag cloud 방식의 장점을 결합하여 각 단점을 보완하는 TreeTag cloud 방식을 제안한다.

본 논문의 구성은 다음과 같다. 2 장에서는 관련 분류방식에 대해서 알아보고, 3 장에서 Tag cloud 방식의 문제점을 짚어보고, 4 장에서는 본 논문에서 제안하는 TreeTag cloud 모델에 대해 설명하고, 5 장에서 결론 및 향후 연구과제를 기술한다.

2. 관련분류 방식

본 장에서는 현재 분류방식 중 대표적인 카테고리 방식과 최근 많이 사용되고 있는 Tag cloud 방식에 대해 설명한다.

2.1. 카테고리 방식

카테고리란 범주라는 뜻이고 범주는 사물을 분류한다는 뜻으로 가장 오래된 분류방식이며 가장 일반적인 분류방식이다. 주위에서 흔히 볼 수 있는 서류철 분류방식에서부터 PC 의 디렉터리 관리까지 다양한 분야에서 사용되고 있으며[7][10] 가장 익숙한 분류방식 중 하나라고 하겠다.

(그림 1)은 흔히 볼 수 있는 블로그의 카테고리 방식의 게시물 분류 예이다.

(그림 1) 국내 블로그 사이트 테터툴스에서 제공하는 카테고리 예

2.2. Tag cloud 방식

태그는 꼬리표라는 말로 그 사물에 그것의 특징적인 성질의 단어를 부여한다는 뜻이다. 최근 블로그에서 카테고리 방식의 단점을 보완하기 위해 병행하여 사용되고 있는 분류방식으로 이 태그를 활용한 방식이 Tag cloud 방식이다[2][6][12].

사실 Tag cloud 는 정보검색 측면보다는 해당 블로그에 등록된 컨텐츠 종류의 빈도수를 보여주는 방식으로서 태그들을 일정 화면 영역에 나열함으로써 해당 블로그의 성격과 다루는 분야 등을 파악할 수 있으며 빈도수를 표현하는 방식으로 색깔을 달리하거나 또는 크기를 달리하기도 한다.

(그림 2)는 Tag cloud 의 예로 해당 블로그의 컨텐츠들에 각각 다수의 태그를 입력하고 이를 태그의 빈도에 따라 색깔과 글자크기를 달리하여 표현하고 있다.

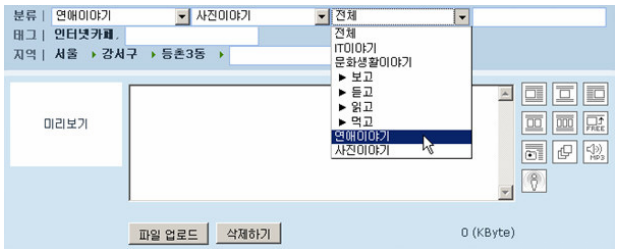
(그림 2) web2.0korea.org 의 Tag Cloud 화면

3. 카테고리 분류 방식과 Tag cloud 분류방식의 문제점

3.1 카테고리 방식의 문제점

게시물을 등록할 카테고리 방식에 부합되도록 하려면 이미 생성되어있거나 또는 생성할 하나의 단어를 선택하여 그 이름의 분류에 소속되어야 한다. 하지만 블로그에서 게시물을 작성할 때 정확히 하나의 분류에 집어넣을 수 없다는 것이 카테고리 방식의 가장 큰 한계이다. 어떤 목적에 의해 작성되는 문서들은 해당 목적에 맞는 분류가 명확하지만 블로그 같이 자유롭게 작성되는 글의 경우 등록시킬 카테고리 항목을 하나로 명확히 잡기 힘들다. 이렇듯 하나의 글이 하나의 카테고리 항목으로 표현할 수 없을 경우 컨텐츠 작성자가 가장 적합하다고 판단되는 카테고리 항목을 선택해야 하므로 방문자의 주관적인 판단과 다를 경우 해당 컨텐츠를 찾을 수 있는 방법이 없게 된다. 이를 보완하는 방안으로 두 개 이상의 카테고리 항목에 등록시키는 다중 카테고리 방식이 있다[2].

(그림 3)은 다중 카테고리 방식의 예이다.



(그림 3) 다중 카테고리 방식

3.2 Tag cloud 방식의 문제점

Tag cloud 를 통해서 방문자들은 해당 블로그에 등록된 컨텐츠들의 성격과 빈도를 알 수 있지만 어떤 자료를 찾기 위한 목적을 가지고 방문한 방문자인 경우 찾고자 하는 자료의 키워드를 찾기 위한 어떤 일관된 방법이 없다는 문제점이 있다. 이는 선형검색 방법을 이용해 검색하게 되므로 $O(n)$의 수행복잡도를 가진다[4].

4. TreeTag cloud 모델

4.1 TreeTag cloud 모델 구조

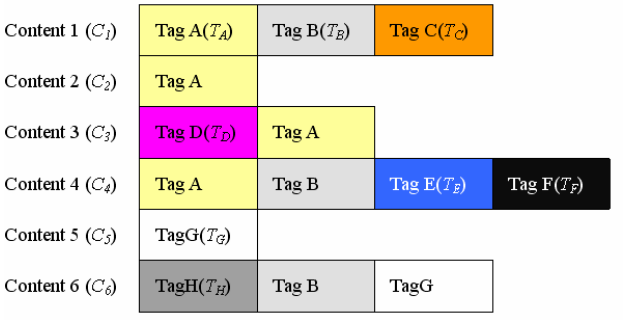
본 논문에서 제안하는 TreeTag cloud 방식은 해당 블로그에 등록된 컨텐츠 목록을 분류하는데 등록된 태그들을 그룹화하여 구현한 기법이다.

본 논문에서 사용되는 용어들은 (표 1)과 같다.

(표 1) 사용 용어 설명

용 어	설 명
C_n	태그가 붙은 content (n : 상수)
c	최상위 노드의 조건 태그 수
d	트리의 depth
T	태그
T_x	해당 블로그에 있는 태그 x 의 수
N_d	트리구조의 d depth 노드
$cnt(x)$	x 의 개수

블로그에 게시되는 컨텐츠는 (그림 4)와 같은 구조로 형성된다.

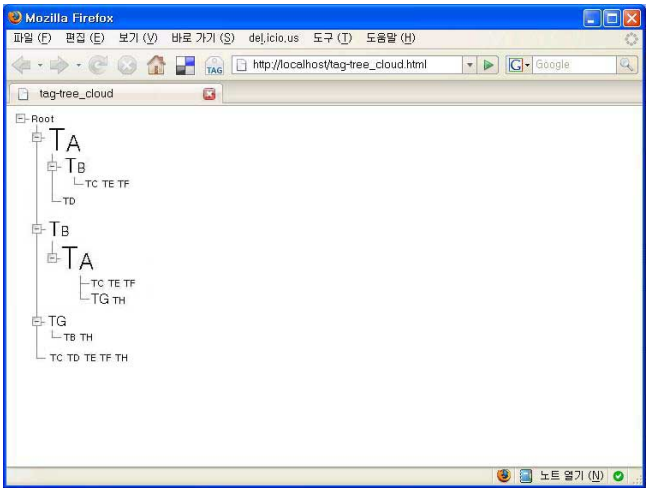


(그림 4) 블로그 컨텐츠 구조 예

방문자로부터 입력 받은 c 를 이용하여 $cnt(T_x) >= c$ 조건에 부합하는 T 를 최상위 노드(N_0)로 선택하고 나머지는 Tag cloud 로 구성한다. 그 뒤 각 N_0의 T 를 포함하는 C_n의 T 들을 그룹화하여 $cnt(T_x) > c$ 인 T 를 하위 노드(N_1)로 두고 나머지는 역시 Tag cloud 로 구성한 뒤 각 노드(N_1)의 T 를 포함하는 C_n 들을 그룹화하여 위를 반복한다. 만일 $N_d = d$ 이거나 그룹화한 T 가 없을 경우 반복을 종료한다. 이때 $N_d = d$ 로 종료되었을 경우 나머지 T 를 Tag cloud 로 구현한다. 다음의 (그림 5), (그림 6)은 각각 본 논문에서 제안한 TreeTag cloud 구성 데이터 XML, 구현결과 화면이다. 구현결과는 $c = 2, d = 3$ 의 조건으로 구현하였다.

```
<?xml version="1.0" encoding="UTF-8"?>
<root>
        <node>TA<count>4</count>
                <node>TB<count>2</count>
                        <node>TC<count>1</count></node>
                        <node>TE<count>1</count></node>
                        <node>TF<count>1</count></node>
                </node>
                <node>TD<count>1</count></node>
        </node>
        <node>TB<count>3</count>
                <node>TA<count>2</count>
                        <node>TC<count>1</count></node>
                        <node>TE<count>1</count></node>
                        <node>TF<count>1</count></node>
                </node>
                <node>TG<count>2</count></node>
                <node>TH<count>1</count></node>
        </node>
        <node>TG<count>2</count>
                <node>TB<count>1</count></node>
                <node>TH<count>1</count></node>
        </node>
        <node>TC<count>1</count></node>
        <node>TD<count>1</count></node>
        <node>TE<count>1</count></node>
        <node>TF<count>1</count></node>
        <node>TH<count>1</count></node>
</root>
```

(그림 5) TreeTag cloud 구성 데이터 XML



(그림 6) Tag-Tree cloud 화면

4.2 TreeTag cloud 모델 장점

한 연구원이 연구한 결과에 의하면 유저인터페이

스 설계 시 매뉴항목 따위를 나열 할 때 2 단 이상의 가로-세로 배열은 세로 배열보다 상대적으로 긴 시선 이동 경로가 복잡하여 사용자가 화면의 내용을 인식하는 속도를 느리게 하는 원인이 된다[11]. 이 연구의 결과를 적용해본다면 현재 Tag cloud 방식의 표현형식으로 사용되고 있는 일반 텍스트문서 단어나열방식은 시선이동경로가 좌에서 우로, 그리고 다시 좌로 돌아와야 하는 긴 시선이동 경로를 가지고 있어 필요한 태그를 검색하는데 어려움이 있다는 것을 알 수 있다. 그렇다고 모든 태그를 세로로 나열한다면 시간이 경과함에 따라 계속 누적되는 태그의 특성으로 볼 때 긴 페이지를 만들어 스크롤이 필요하게 되어 더 불편하게 된다.

본 논문에서 제안하는 TreeTag cloud 는 트리구조의 특징으로 세로배열 형식을 가지고 있으면서 오른쪽 사선방향으로 시선이동을 유도함으로써 Tag cloud 방식보다 짧은 시선이동 경로를 유지할 수 있으며 자식노드를 닫거나 열 수 있어 화면에 필요한 부분만 검색할 수 있으며 등록된 태그를 구조화 함으로써 원하는 정보를 검색하는데 좀 더 용이하다.

5. 결론 및 향후 연구과제

Tag cloud 분류방식을 사용하는 웹사이트들은 단점을 보완하고자 카테고리 방식을 병행하여 사용하고 있다. 하지만 각각 단점을 가지고 있는 상태로 사용하기 보다 본 논문에서 제안하는 이 둘을 결합하여 서로의 단점을 보완한 분류방식을 사용한다면 좀 더 효율적으로 컨텐츠들을 분류할 수 있고 방문자 입장에서는 찾고자 하는 정보를 좀 더 쉽게 찾을 수 있을 것이다. 그러나 게시물 게시자의 비 연관성 태그 및 모든 게시물에 별도의 태그 하나씩만 입력을 할 경우 효용성이 떨어지는 문제가 있으며 컨텐츠에 태깅 하는 주체는 컨텐츠 생산자(작성자)가 되므로 태그 값은 전적으로 작성자의 주관적인 판단에 의해 작성되기 때문에 타 블로그 간 같은 부류의 컨텐츠라도 다른 분류에 속할 수 있다는 문제도 있다. 또한 띄어쓰기나 영문과 한글의 중복표현으로 실제 같은 뜻인데도 두 개 이상의 태그가 생성될 수 있는 문제 또한 안고 있는데 이 부분은 현재 자동 태깅 시스템을 통해 어느 정도 보완된 상태이다[6][12]. 이 부분은 앞으로 계속적인 연구가 필요하다.

또한 로컬 파일을 하드디스크에 저장 시 디렉터리 분류방식을 대체할 수 있는 분류방식으로도 연구해 볼 가치가 있다.

참고문헌

[1] Tim O'Reilly, "What is Web2.0", http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

[2] subee's blog, "태그와 카테고리의 갈등", http://kindsubi.com/blog/38?TSSESSION=5289a21d79294b131380bf8b521ec738, 2006.6

[3] wikipedia, "Tag cloud", http://en.wikipedia.org/wiki/Tag_cloud

[4] 이지영, "IT CookBook, C 로 배우는 쉬운 자료구조", 한빛미디어, 2005

[5] 전종호, 이승윤, 이강찬, 이원석, "웹 2.0 기술 동향에관한 연구", 제 26 회 한국정보처리학회 추계학술발표대회 논문집 제 13 권 제 2 호, 2006.11

[6] 박영욱, "웹 2.0 구현의 핵심, 태그", 마이크로소프트웨어 2006 년 5 월호, pp. 124-129, 2006.5

[7] Jerzy Lewak, Slawek Grzechnik, Jon Matousek, "Method for accessing computer files and data, using linked categories assigned to each data file record on entry of the data file record" United States Patent, pp. 6-10, 1996.8

[8] wikipedia, "Blog", http;://en.wikipedia.org/wiki/blog

[9] S.C. Herring, L.A. Scheidt, S. Bonus, and E. Wright, "Bridging the Gap: A Genre Analysis of Weblog", Proc. 37th Hawaii Int'l conf. System Sciences, IEEE Press, 2004.

[10] Intelligence and security informatics / IEEE International Conference ; Sharad Mehrotra...[et al.] (eds.) Springer 2006 006.330973 I22i pp. 772, 2006

[11] Matteo Penzo, "Label Placement in Froms" http://www.uxmatters.com/MT/arthives/000107.php

[12] 김중태, "웹 2.0 시대의 기회 시맨틱 웹", 디지털미디어 리서치, pp. 172-195, 2006