Journal of Korea Multimedia Society Vol. 12, No. 3, March 2009(pp. 427-435)

웹 콘텐츠에서 모바일 디바이스 기반 아이템 블록을 추출하기 위한 세그먼트 알고리즘

김수도[†], 박태진[††], 박만곤[†††]

요 약

사용자들은 웹 콘텐츠의 세부 내용단위인 메뉴, 로그인, 뉴스, 동영상 등 다양한 아이템에서 자신이 흥미있는 아이템을 찾아 읽고 아이템에 연결된 하이퍼링크를 클릭한다. 모바일 디바이스처럼 작은 스크린에서는 데스크탑 기반의 웹 콘텐츠를 동시에 보여주기 힘들어 사용자는 좌우 또는 상하 스크롤링을 통해 아이템을 찾아 헤매는 검색의 불편함이 발생한다. 사용자가 자주 사용하거나 또는 원하는 아이템을 찾아 먼저 표현하여 모바일 인터페이스 조작의 불편함을 감소시킬 수 있다. 그러기 위해 웹 콘텐츠는 세부 내용단위인 아이템별로 세그먼트되어야 한다. 기존 대부분의 세그먼트 알고리즘들은 웹 콘텐츠의 HTML 코드와 모바일 사이즈에 기반하여 세그먼트하고 있어 다양한 아이템들이 구조적으로 더욱 복잡하게 형성되고 있는 현대의 웹 콘텐츠에서 내용 단위인 아이템 블록으로 세그먼트하는데 여러 어려움이 있다. 본 논문에서는 데스크탑 웹 콘텐츠에서 내용 단위에 기반한 아이템 블록 추출을 위한 세그먼트 알고리즘을 제안한다.

A Segment Algorithm for Extracting Item Blocks based on Mobile Devices in the Web Contents

Su-Do Kim[†], Tae-Jin Park[††], Man-Gon Park[†††]

ABSTRACT

Users are able to search and read interesting items and hence click hyperlink linked to the item which is detailed content unit such as menu, login, news, video, etc. Small screen like mobile device is very difficult to viewing all web contents at once. Browsing and searching for interesting items by scrolling to left and right or up and down is discomfort to users in small screen. Searching and displaying directly the item preferred by users can reduces difficulty of interface manipulation of mobile device. To archive it, web contents based on desktop will be segmented on a per-item basis which component unit of web contents. Most segment algorithms are based on segment method through analysis of HTML code or mobile size. However, it is difficult to extract item blocks. Because present web content is getting more complicated and diversified in structure and content like web portal services. A web content segment algorithm suggested in this paper is based on extracting item blocks is component units of web contents.

Key words: Item Block(아이템 블록), Segment Algorithm(세그먼트 알고리즘), Mobile Web(모바일 웹), Adapting Algorithm(적응화 알고리즘)

※ 교신저자(Corresponding Author) : 김수도, 주소 : 부산광역시 남구 대연3동 599-1(608-737), 전화 : 051)629-6240, FAX : 051)628-6155, E-mail : kim-sudo@hanmail.net
접수일 : 2008년 8월 5일, 완료일 : 2008년 12월 17일
[†] 준회원, 부경대학교 누리사업단 계약교수
[††] 정회원, 마산대학 조선메카트로닉스학과 강의교수 (E-mail : csptj2@naver.com)
[†††] 종신회원, 부경대학교 전자컴퓨터정보통신공학부 교수 (E-mail : mpark@pknu.ac.kr)

1. 서 론

오늘날 인터넷은 교육, 비즈니스분야 뿐 아니라 다양한 분야에서 정보를 습득하기 위한 기본 통신 도구로 활용되고 있다. 사용자들은 언제 어디서나 필요할 때마다 바로 사용할 수 있는 모바일을 통해서도 데스크탑처럼 웹의 다양한 정보를 검색하고 사용할 수 있기를 원한다. 모바일 디바이스를 통해 월드와이드웹에 접근하여 웹 콘텐츠를 볼 수 있는 것을 모바일 브라우징이라고 하며 전문가들은 미래의 웹은 모바일 디바이스에 놓이게 될 것으로 예상하고 있다[1].

그러나 작은 스크린 화면을 가지는 모바일 디바이스에서 데스크탑에 최적화되어 있는 웹 콘텐츠를 보기에는 여러 가지 어려움이 있다. 기존의 네이트(Nate), 메직엔(Magicⓝ), 이지아이(ez-i) 등은 모바일 디바이스에 맞게 별도로 만들어 제공한 인터넷 사이트로 이런 상황은 사용자에게 이동통신이라는 제한적인 환경에서의 서비스만 가능하도록 하여 데스크탑 기반의 다양하고 풍부한 정보와 서비스의 이용을 가로막고 있다[2].

데스크탑 기반의 웹 콘텐츠는 전체가 하나의 단일 내용이 아닌 메뉴, 로그인, 뉴스, 동영상 등 다양한 아이템들로 구성된다. 사용자들은 웹 콘텐츠에서 자신이 흥미있는 아이템을 찾아 읽고 아이템에 연결된 하이퍼링크를 클릭한다. 모바일 디바이스처럼 작은 스크린 사이즈에서는 데스크탑의 전체 웹 콘텐츠를 보여주기 힘들기 때문에 사용자가 좌우 또는 상하 스크롤링을 통해 아이템을 찾아 헤매는 검색의 불편함이 발생된다. 사용자가 원하는 아이템 단위로 찾아 먼저 표현하여 인터페이스 조작의 불편함을 감소시켜줄 수 있다[3-6]. 그러기 위해 웹 콘텐츠는 전체가 하나의 단위가 아닌 세부내용단위인 아이템 단위로 세그먼트되어져 사용자가 원하는 아이템들을 선별하여 표현할 수 있어야 한다.

웹 콘텐츠를 모바일 디바이스에 적절한 단위로 세그먼트하기 위한 다양한 알고리즘들이 제안되었다[7-9]. 대부분의 세그먼트 알고리즘은 HTML 코드 분석을 통한 휴리스틱 알고리즘에 기반하고 있다. 휴리스틱 알고리즘은 웹 콘텐츠의 코드 분석을 통해 패턴을 찾아내고 패턴을 이용하여 모바일 사이즈에 맞게 블록 단위로 세그먼트하는 방식이다[10-13]. 그러나 웹 콘텐츠를 코드 또는 크기에만 의존할 경우 포털과 같이 다양한 아이템들이 구조적으로 더욱 복잡하게 형성되고 있는 현대의 웹 콘텐츠에서 내용단위인 아이템 단위별로 세그먼트하는데 많은 어려움이 있다.

본 논문에서는 사용자가 원하는 아이템을 찾아 표현하기 위해 데스크탑 기반의 웹 콘텐츠에서 세부내용을 표현하는 아이템 단위로 세그먼트하기 위한 알고리즘을 제안한다.

2. 관련연구

모바일 디바이스처럼 작은 스크린 사이즈에서 웹 콘텐츠를 표현하기 위해 적절한 단위의 블록으로 세그먼트하기 위한 다양한 알고리즘들이 제안되었다.

Timo Laakko와 Tapio Hiltunen가 제안한 [10]은 Kontti 프로젝트의 한 부분으로 메타데이타를 이용하여 웹 콘텐츠를 모바일 디바이스에서 인지가능한 단위로 분할하는 알고리즘을 제안하였다. 웹 콘텐츠를 전달하기 적절한 단위로 분할하기 위해 헤더와 테이블 같은 내용을 나눌수 있는 요소에 브레이크 포인트(Break Points)를 생성한다. 분해(Decomposition)를 위해 각 단위에 우선권 값을 할당하고 우선권 값에 의해 분할 지점이 결정된다. 아래 그림 1은 웹 콘텐츠

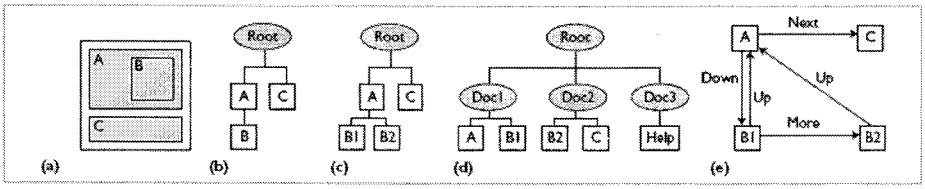


그림 1. 프록시의 적응화 방법. 박스는 인지할수 있는 단위를 표현하고 있다. (a) 원본 웹 콘텐츠 (b) 적응화 규칙에 의해 분해된 문서 (c) 분할한 후 단위 구조. B를 B1과 B2로 쪼개는 브레이크 포인트는 단위 B의 중간위치에 있다. (d) 마지막 분할된 전달 단위들(Doc1에서 Doc3). (e) 프레임워크는 단위들 사이에서 네비게이션이 가능하도록 링크를 생성한다.

가 분해되는 과정을 보여주고 있다.

Hwang과 동료들이 [11]에서 제안한 구조-인식 웹 트랜스코딩(Structure-Aware Web Transcoding) 알고리즘은 일반 윤곽 변환과 선택 제거 변환을 이용하여 원본 웹 페이지의 레이아웃을 최대한 보존하면서 서브페이지로 제거할 블록을 추출한다. 일반 윤곽 변환에서는 패턴 매칭 알고리즘(Prefix Pattern-Matching Algorithm)을 이용하여 웹 콘텐츠를 효과적으로 전달하기 위해 많이 사용된 반복 레이아웃 패턴들을 찾아 서브페이지 블록으로 변환한다. 선택 제거 변환에서는 테이블의 셀의 중요도를 검사하여 중요하지 않은 셀을 서브페이지 블록으로 변환한다.

Deng Cai와 동료들이 [12]에서 제안한 VIPS (Vision-based Page Segmentation Algorithm) 알고리즘은 사람의 인지단위인 시각에 기반하여 웹 페이지를 분할되고 계층적으로 구조화하는 방식이다. 웹 페이지를 DOM 트리로 구성하고 트리의 잎노드(Leaf Node)에서 블록들을 추출하고 블록들을 수직 또는 수평으로 나누는 시각 분리대(Separator)를 이용하여 페이지를 세그먼트하는 알고리즘이다. 아래 그림 2는 웹 페이지를 세그먼트하는 과정을 그림으로 표현하였다.

Yu Chen과 그 동료들은 [13,14]에서 웹 페이지 디자인에 따라 5종류의 하이-레벨 콘텐츠 블록들("Header, Footer, Left Sidebar, Right Sidebar")을 구분하여 검출한다. HTML의 DOM 트리에서 웹 페이지의 위쪽 Header 부분 N 픽셀 범위안에 블록이 있으면 "Header" 블록으로 분류하고 "Footer" 블록, "Left"와 "Right" 사이드바를 검출하는 데에도 유사한 접근을 사용하고 어디에도 포함되지 않으면 "Body"로 분류한다. 다시 블록들을 모바일 디바이스에 맞게 쪼개기 위해 <HR><TABLE><TD><DIV>와 이미지 바처럼 블록들을 세부적으로 나눌 수 있는 명시적 분리대와 줄간격 또는 갭(Gab)처럼 함축적 분리대를 이용하여 웹 콘텐츠를 세그먼트한다.

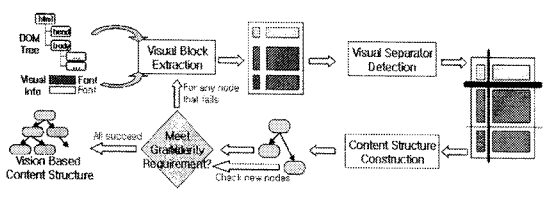


그림 2. VIPS 알고리즘 과정

위의 알고리즘처럼 대부분의 세그먼트 알고리즘이 표처럼 특수한 형태에 기반하거나 색상 또는 글자 크기처럼 HTML 코드에 기반하여 웹 콘텐츠를 세그먼트하고 있어 같은 아이템 주제를 표현하는 아이템이 코드에 의해 여러개의 블록으로 나누어질 수 있다. 본 논문에서는 웹 콘텐츠의 HTML 코드뿐 아니라 내용분석에 기반한 아이템 단위의 블록으로 세그먼트한다.

3. 아이템 기반 세그먼트 알고리즘

사용자들은 웹 콘텐츠 전체를 하나의 단위로 인지하지 않는다. 웹 콘텐츠는 각 주제를 표현하는 아이템들로 나누어져 있고 사용자들은 자신이 흥미있는 내용을 표현하는 아이템을 선별하여 읽는다. 현대의 웹 콘텐츠는 다양하고 많은 종류의 아이템들을 하나의 웹페이지에 표현하기 위해 구조적으로 시각적으로 점점 더 복잡하게 구성되고 있다. 기존의 대부분의 알고리즘들이 표나 글자크기와 같은 HTML 코드나 크기 등에 기반한 블록 단위로 세그먼트하고 있어 현대처럼 복잡하게 구성된 웹 콘텐츠에서 코드에 의존하여 세그먼트할 경우 하나의 주제를 표현하는 아이템이 여러 개의 블록으로 분리되어 개인화 서비스처럼 내용 단위로 세그먼트하여 제공하는 경우 문제를 발생시킨다. 모바일 사용자들이 가장 많이 이용하는 포탈의 경우 많은 아이템들이 더욱 복잡한 구조로 표현되고 있어 아이템 단위의 세그먼트의 실패가 더욱 두드러진다. 본 논문에서는 아이템 단위의 블록을 추출하기 위한 세그먼트 알고리즘을 제안한다. 웹 콘텐츠는 HTML 파서 단계, 블록 추출 단계 그리고 아이템 추출 단계를 거친다.

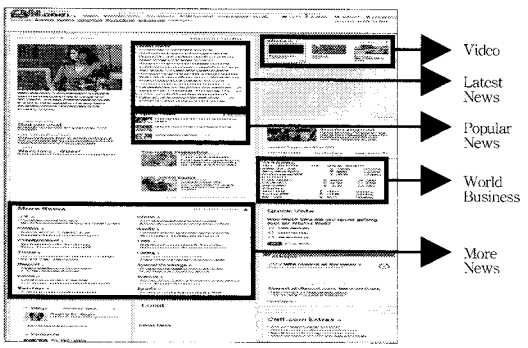


그림 3. CNN 웹 페이지의 아이템들

HTML 파서 단계에서는 웹서버로부터 HTML 소스를 읽어 DOM 트리를 생성한다. 블록 추출 단계에서는 먼저 HTML 코드에 기반하여 작은 단위의 블록들로 세그먼트한다. 아이템 추출 단계에서는 각 블록이 어떤 내용을 나타내는 종류인지 분석한 후 블록과 블록의 의미적 관계를 조사하여 같은 아이템을 표현하는 경우 통합하고, 아닌 경우 통합하지 않는다. 아이템 추출 단계후의 블록 단위가 아이템 블록으로 처리된다.

3.1 HTML 파서 단계

사용자가 웹페이지를 요청하면 웹브라우저에서는 웹서버로부터 다운받은 HTML 문서를 문서해석 프로그램을 통해 태그와 내용을 분리하여 해석된 내용을 화면에 출력한다. 이런 문서해석 프로그램을 파서(Parser)라 한다. 본 논문에서는 JAXP 파서 중에서 DOM 파서를 이용하여 HTML 문서를 XML형식처럼 읽어 태그, 속성, 속성 값 등으로 분리한 DOM(Document Object Model) 트리를 생성한다. HTML DOM 트리의 각 노드는 HTML의 태그로, 속성은 태그의 속성으로, 그리고 노드의 데이터 값은 문서의 내용들로 구성된다.

3.2 블록 추출 단계

DOM 트리의 각 노드들은 HTML 태그로 구성되어 있고, 노드의 속성들은 HTML 태그의 속성이름들로 구성된다. 웹 콘텐츠의 아이템들을 디자인하기 위해 HTML 태그들을 이용하여 구조적으로 형성하고 있다. 블록 추출 단계에서는 먼저 HTML 태그를 분석하여 웹 콘텐츠를 세부적으로 나누는 태그 기반의 블록을 추출하여 DOM 트리를 블록 트리로 재구성한다. 웹 콘텐츠를 구조적으로 구분하는데 사용되는 HTML 태그 리스트를 관리하고, 전위순회(Preorder) 방식으로 각 노드들을 탐색하여 만약 블록 추출 규칙(표 1)에 일치하는 HTML 태그를 만나면 블록 노드를 생성하여 태그의 서브루트 위치로 삽입한다. 모든 노드의 탐색이 끝나면 DOM 트리는 블록 트리 구조로 재구성된다. 아래 표 1은 블록 추출 규칙에 사용되는 HTML 태그 리스트이고 아래 그림4는 블록 추출 단계를 통해 CNN의 "More News" 아이템이 블록 트리 구조로 재구성된 것을 표현한 그림이다.

표 1. 블록 추출 규칙에 사용된 HTML 태그들

태그명	태그 역할
<DIV>	영역 구분 또는 분할
<TABLE>	표 형식
<FORM>	사용자 입력 양식
<UL>, <OL>, <DL>	목록형식
<HR>	수평선
<P>	문단나누기
<H1> ~ <H6>	제목형식

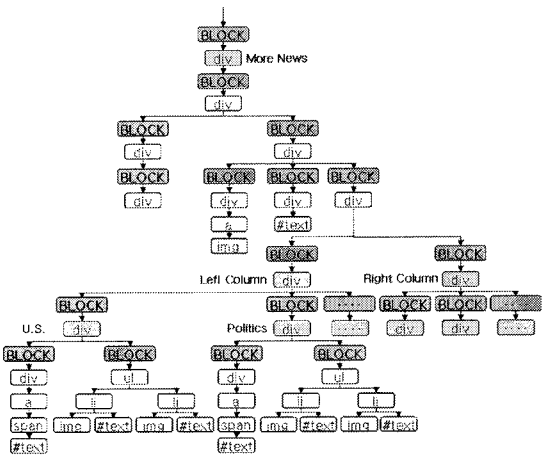


그림 4. CNN의 "More News" 블록 트리 구조

3.3 아이템 블록 추출 단계

블록 추출 단계에서 추출된 태그 기반의 블록은 하나의 주제를 표현하는 아이템일 수 있고 또는 아이템의 일부분에 해당될 수도 있다. 대부분의 경우 하나의 아이템이 다양한 태그들을 이용하여 표현되기 때문에 추출된 블록은 아이템의 일부분인 세부 단위인 경우가 많다. 위의 그림 4처럼 "More News" 아이템을 표현하기 위해 많은 태그들이 구조적으로 복잡하게 구성되고 있다. 아이템 블록 추출 단계에서는 블록 추출 단계에서 추출된 각 블록의 내용을 분석하여 블록의 종류를 구분한 후, 블록과 블록의 의미적 관계에 따라 아이템 블록의 조건에 해당될 경우 하나의 블록으로 통합하고 그렇지 않으면 각각 다른 아이템으로 구분하여 통합하지 않는다. 예로 앞에 블록이 제목을 표시하고 뒤의 블록이 문단 내용을 표현하고 있는 경우 두 개의 블록은 하나의 주제를 표현하는 같은 아이템이라고 예상하고 통합한다.

표 2. 카테고리별 분류

우선권	명칭	분류 기준
1	Invalid Block	블록에 텍스트 또는 이미지 등과 같은 유효한 콘텐츠 내용이 없을때
2	Head Block	블록에 Head 특성을 가진 HTML 태그로 구성된 노드를 포함할 때, 예로 <H1>~<H6> 또는 Head 속성으로 지정되어 있을때
3	Image Block	블록에 Image 태그만 또는 Image 태그와 Image를 설명하는 제목 텍스트만을 포함하고 있을 때
4	Form Block	Form 양식을 구성하는 태그만으로 지정되어 있을 때, 예로 <form>, <input>, <TextArea> 등
5	Content Block	그 외 내용을 표현하기 위해 구성된 태그, 예로 , ,<p> 등

아이템을 추출하기 전 먼저 블록들을 블록의 내용 분석에 따라 카테고리별로 분류한다. 분류기준은 블록안의 HTML 태그들을 분석하여 표 2처럼 분류한다. 만약 중복될 경우 우선권 순서로 분류된다. 아래 그림 5는 표 2에 의해 분류된 블록의 명칭을 표시한 것이다.

각 블록의 내용에 따라 블록의 종류가 결정된 후 블록과 블록의 의미적 관계를 조사하여 아이템 블록으로 유추해 나간다. 아이템 블록을 추출하기 위해

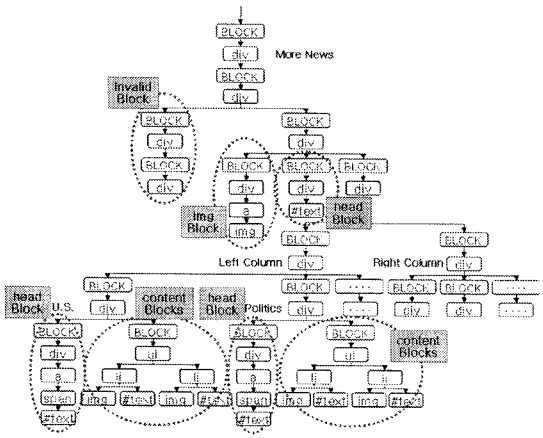


그림 5. "More News" 아이템의 블록 명칭

아래의 4단계로 진행한다. 4단계를 모두 거친 후 최상위 Unit 블록들이 아이템 블록으로 지정된다.

step 1 : 최하위 블록(Leaf Block)은 자동적으로 Unit 블록으로 지정한다.

step 2 : 후위순회(Postorder) 방식으로 Unit 블록과 Unit 블록을 비교하여 아이템 블록의 규칙(표 3)에 맞으면 하나의 Unit 블록으로 통합한다. 만약 맞지 않으면 각각 다른 아이템이라 생각하여 통합하지 않는다.

step 3 : 만약 블록의 하위노드에 1개의 Unit 블록만 포함하면 자동으로 블록을 Unit 블록으로 지정한다.

step 4: step 2로 순환 반복한다.

Andrew W. Cole의 트리 비교 알고리즘[15]은 트리와 트리가 일치하는 여부를 판별하기 위해 각 노드와 노드사이의 간선(Edge)을 스트링으로 변환한 후 일치하는 간선 수를 계산하였다. 본 논문에서는 Unit A와 B의 유사성을 계산하기 위해 Andrew W. Cole의 트리 비교 알고리즘을 이용하여 전체 간선 수에서 일치하는 간선 수의 비율이 80%를 넘을 경우 유사하다고 간주하였다.

S(Similarity) = C / (A_list.size() + B_list.size()) (식 1)

표 3. 아이템 블록 추출을 위한 규칙

우선권	명칭	조 건
1	Invalid Rule	2개의 Unit 블록 A와 B중 하나가 Invalid Block일 때
2	Low-Text Rule	2개의 Unit 블록 A와 B중 하나가 텍스트만 포함하고 텍스트 크기가 50자 이하일 때
3	Form Rule	2개의 Unit 블록 A와 B 모두 같은 Form Block에 포함되어 있는 경우.
4	Image Rule	2개의 Unit 블록 A와 B중 하나가 Image Block일 때.
5	Head Rule	처음 Unit 블록 A가 Head Block이고 Unit B가 Head Block이 아닐 경우.
6	Same-Pattern Rule	2개의 Unit 블록 A, B가 유사한 패턴을 가지는 경우.

A_list_size : Unit 블록 A의 간선 리스트 개수
B_list_size : Unit 블록 B의 간선 리스트 개수
C : A_list와 B_list에서 일치된 간선 수

아이템 단위 추출 단계후 최상위 Unit 블록이 하나의 아이템을 표현하는 블록으로 선택된다. 아래 그림 6의 굵은 선은 CNN 웹페이지에서 추출된 아이템 블록을 표시한 것이다.

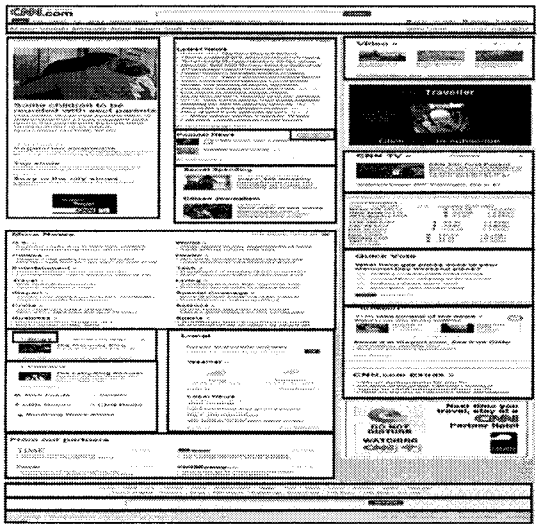


그림 6. CNN의 추출된 아이템 블록들

3.4 비교 및 실험결과

대부분의 알고리즘들이 태그를 이용한 패턴 분석에 기반하여 블록을 추출하는 휴리스틱 알고리즘을 사용하고 있다. 그러나 대부분 표처럼 특수한 구조에 의존하거나 크기와 태그에만 기반하여 블록을 추출할 경우 내용단위인 아이템 단위의 블록을 추출하는데 어려움을 가지고 있다. 표 4는 웹 콘텐츠의 블록 추출 알고리즘과 본 논문에서 제안한 아이템 단위의 블록 추출 알고리즘을 비교한 것이다.

표 5는 본 논문에서 제안하는 아이템 기반 세그먼트 알고리즘의 성능을 평가하기 위하여 각 분야별 사이트에서 선택한 20개의 웹 사이트들로서 각 웹사이트의 인덱스 페이지를 테스트에 이용하였다. 그 이유는 인덱스 페이지는 전체 웹사이트의 내용을 표현하기 위해 다양한 아이템들이 복잡하게 구성되어 있기 때문이다.

아이템 기반 세그먼트 알고리즘 과정에서 추출된 아이템 블록의 개수는 표 6에 나타낸다. 20개의 웹 사이트에 적용한 결과 아이템 블록의 추출 정확률는 84.55%이며 에러률은 15.45%였다. 에러률에서 여러 개의 아이템이 하나의 아이템으로 잘못 묶인 경우는 평균 0.6개이며, 하나의 아이템으로 묶여야 하는 블록이 따로 추출된 경우는 평균 3.55개였다. 여러 개의

표 4. 블록 추출 알고리즘 비교분석

블록 추출 알고리즘	블록 추출 기준	블록 추출 방법
구조-인식 웹 트랜스코딩	• 일반 윤곽 변환 • 선택 제거 변환	• 구조적 태그(예:<UL>, <Table>)에서 반복적 패턴 • 테이블 구조의 셀
시각 기반 페이지 분할	• DOM 트리 이용 • 시각적 인식에 기반한 페이지 분할	• 줄바꿈 태그(예: <p>) • 배경 색상 • 텍스트 크기 • 노드 크기
하이-레벨 콘텐츠 블록 검출	• 하이-레벨 콘텐츠 블록 (Header, Footer, Left SideBar, Right SideBar, Body 분류)	• 공간적 위치에 의해 분류 (예: 페이지 위쪽 N픽셀 범위안에 포함되면 Header 블록) • 명시적 분리대에 의해 블록 분할 (<HR><TABLE><TD><DIV>,이미지바) • 함축적 분리대를 이용하여 블록 분할 (수평 또는 수직 Gap(갭)을 이용)
아이템 블록 검출 (제안 알고리즘)	• 세부내용단위인 아이템에 기반한 블록 추출	• 태그에 기반한 블록 추출 (예: <P><DIV><UL><H1> 등) • 내용 분석을 통한 블록 분류 (Invalid,head,Image,Form,Content Block) • 블록과 블록간의 의미적 관계에 기반한 블록 통합 (예: Head Block과 Content Block간의 관계)

표 5. 분류별 웹 사이트 리스트

번호	웹 페이지 URL	분류
1	http://www.daum.net/index.html	포탈
2	http://kr.yahoo.com/index.html	
3	http://www.naver.com/index.html	
4	http://www.korea.com/index.html	
5	http://www.google.com/index.html	
6	http://www.nate.com/index.html	
7	http://www.cyworld.com/main2/index.asp	
8	http://edition.cnn.com/index.html	뉴스
9	http://www.chosun.com/index.html	
10	http://www.etnews.co.kr/index.html	
11	http://www.times.com/index.html#	
12	http://www.latimes.com	
13	http://www.bbc.co.uk	방송
14	http://www.kbs.co.kr/index.html	
15	http://www.mbc.co.kr/index.html	
16	http://www.interpark.co.kr/malls/index.html	쇼핑
17	http://www.amazon.com/gp/homepage.html	
18	http://www.ebay.com	
19	http://java.sun.com/index.jsp	회사
20	http://www.ibm.com/us	

아이템이 하나로 묶는 것보다 같은 아이템을 다른 아이템으로 인식하는 오류가 더 많게 나왔다. 국내의 데스크탑 기반의 웹 콘텐츠들은 내용을 표현하기 위해 저마다 다양한 형식과 구조를 사용하고 있어 내용을 제대로 분석하지 못해 에러를 발생시키고 있었다. 예로 아이템의 제목을 표현할 때 태그나 스타일시트를 사용하는 외에 이미지를 사용하는 경우도 있었고, <UL> 태그처럼 리스트형식으로 지정하는 경우도 있었다. 아이템의 제목의 위치도 각 내용의 앞에 위치하지 않을 경우 블록 사이의 관계 분석에서 오류를 발생하는 경우도 있었다. Google의 경우는 모바일까지 고려하여 웹 콘텐츠를 만들어 제공하여 성공률이 100%가 나왔고, Korea 그리고 Times처럼 아이템수가 많아도 내용을 표현하기 위한 태그와 구조가 규칙적일 때 아이템 추출의 성공률이 높게 나왔다.

그림 7은 웹 콘텐츠를 종류별로 분류한후 아이템 추출하는데 걸린 평균 정확률과 평균 처리시간을 비교한 그림이다. 포탈과 뉴스가 다른 종류의 웹 콘텐츠보다 처리시간이 높게 측정되었다. 간혹 Google처럼 포탈 분류에 포함되지만 모바일에 맞추어 간단한 웹 콘텐츠를 제공하고 있어 아이템 추출시간이 다른 웹 콘텐츠에 비해 현저히 낮게 측정되는 경우도 있었다.

표 6. 아이템 블록의 추출 결과

번호	웹콘텐츠	N	CN	에러개수		퍼센트(%)	
				MN	LN	에러률	정확률
1	Daum	27	22	0	5	19%	81%
2	Yahoo	31	27	1	3	13%	87%
3	Naver	19	16	1	2	16%	84%
4	Korea	52	52	0	0	0%	100%
5	Google	1	1	0	0	0%	100%
6	Nate	33	22	0	11	33%	67%
7	Cyworld	21	16	1	4	24%	76%
8	CNN	31	27	0	4	13%	87%
9	Chosun	40	27	0	10	25%	75%
10	EtNews	20	18	2	0	10%	90%
11	Times	58	52	0	6	10%	90%
12	LATimes	32	20	2	10	38%	62%
13	BBC	30	24	0	6	20%	80%
14	KBS	34	30	0	4	12%	88%
15	MBC	13	9	2	2	31%	69%
16	Interpark	8	7	0	1	13%	87%
17	Amazon	26	23	0	3	12%	88%
18	ebay	10	9	1	0	10%	90%
19	JavaSun	20	19	1	0	5%	95%
20	IBM	19	18	1	0	5%	95%
평균		26.25	21.95	0.60	3.55	15.45%	84.55%

N : 추출된 아이템 블록의 개수
CN : 정확한 아이템 블록의 개수
MN : 너무 크게 추출된 아이템 블록의 개수
LN : 너무 작게 추출된 아이템 블록의 개수

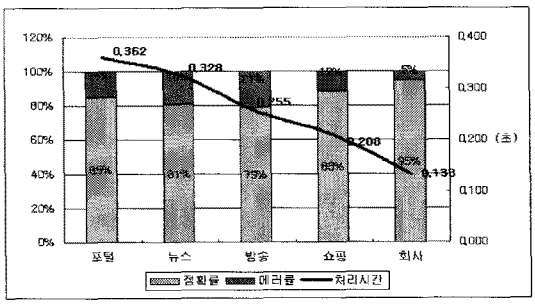


그림 7. 웹 콘텐츠 분류별 정확률과 처리시간

4. 결 론

데스크탑 기반의 웹 콘텐츠를 모바일 디바이스로 표현하기 위한 다양한 연구가 진행되고 있다. 가장 큰 문제중 하나가 데스크탑에 비해 현저히 작은 스크린 사이즈이다. 최근 모바일 디바이스의 스크린사이즈 3인치에 해상도를 800×480까지 최대한 높여 데스크탑과 유사하게 웹 콘텐츠를 보여줄 수 있는 모바일 디바이스가 출시되었다. 그러나 데스크탑 화면이 3인치로 축소되어 표현되는 형태로 가독성과 같은 문제점을 여전히 내포하고 있다. 결국 내용을 보기 위해서 여러 가지 추가기능(확대기능, 세로정렬방식 등)을 부가적으로 지원해야 한다. 사용자는 원하는 내용을 보기 위해 좌우 또는 상하 스크롤링처럼 인터페이스 조작을 거쳐 원하는 아이템을 찾아야 불편함이 발생된다. 사용자가 원하는 아이템을 바로 찾아 표현할 경우 이런 불편함을 감소시켜줄 수 있다. 그러기 위해 웹 콘텐츠는 코드나 크기가 아닌 내용 구성단위인 아이템에 기반하여 세그먼트되어 표현될 수 있어야 한다.

기존의 세그먼트 알고리즘들이 코드나 크기에 기반하여 블록을 추출하므로 다양한 아이템들이 구조적으로 점점 더 복잡하게 표현되고 있는 현대의 웹 콘텐츠에서 세부내용에 기반한 아이템 단위의 블록을 추출하는데 어려움이 있다. 본 논문에서는 이런 문제를 해결하기 위해 웹 콘텐츠 내용을 분석하여 관련없는 주제일 경우 다른 아이템으로, 같은 주제를 표현할 경우 하나의 아이템으로 구분하여 세그먼트하는 아이템 기반 세그먼트 알고리즘을 제안하였다. 이러한 세그먼트 알고리즘은 모바일 브라우징뿐 아니라 데스크탑에서도 검색시 전체 웹 콘텐츠 단위가 아닌 내용기반의 아이템 단위의 검색도 가능하여 사용자의 편리성을 높여 줄 수 있을 것이다.

향후 연구 과제로는 모바일 브라우징을 위해 다양한 아이템으로 표현된 웹 콘텐츠를 사용자가 자주 사용하는 또는 선호하는 아이템을 찾아 먼저 볼 수 있도록 개인화 웹 콘텐츠 생성에 관한 연구를 진행하고자 한다. 개인화 웹 콘텐츠 생성은 사용자가 선호하는 아이템들을 먼저 표현해주어 모바일 브라우징에서 사용자들이 가장 불편해하는 인터페이스 조작과 검색의 불편함을 감소시켜줄 수 있을 것이다.

참 고 문 헌

[1] Muriel Bowie, Adaptation of a Webshop for Mobile Devices, Master Thesis, Computer Science, Fribourg University, October 2005.

[2] 제갈병직, "모바일 풀 브라우저 시장 동향", IITA 주간기술동향 1278호, 한국전자통신연구원, 2006.

[3] V. Roto, "Browsing on Mobile Phones," Nokia Research Center, http://www.research.att.com/~rjana/WF12_Paper1.pdf.

[4] V. Roto, Web Browsing on Mobile Phones-Characteristics of User Experience, Doctoral Dissertation, Dep. Computer Science and Engineering, Helsinki University, 2006.

[5] 김수도, 박만곤, "모바일 환경에서 유형기반 웹 페이지 적응화를 시스템 아키텍쳐," 한국멀티미디어학회, 10권 2호, pp. 108-1111, 2007.

[6] Su-do Kim and Man-Gon Park, "A Study on the M-learning System on CC/PP for Multimedia Messaging Service Adaptation," *Journal of Korea Multimedia Society*, Vol.11, No.6, 2008.

[7] J. Kang and J. Choi, "Detecting Informative Web Page Blocks for Efficient Information Extraction Using Visual Block Segmentation," 2007 International Symposium on Information Technology Convergence, pp. 306-310, 2007.

[8] P. F. Xiang et al., "Effective Page Segmentation Combining Pattern Analysis and Visual Separators for Browsing on Small Screens," Int'l Conf. on Web Intellegence, pp. 831-840, 2006.

[9] C. Wu, G. Zeng, and G. Xu, "A Web Page Segmentation Algorithm for Extracting Product Information," Int'l Conf. on Information Acquisition, pp. 1374-1379, 2006.

[10] Timo Laakko and Tapio Hiltunen, "Adapting Web Content to Mobile User Agents," *IEEE Internet Computing Magazine*, Vol.9, pp. 46-53, Apr. 2005.

[11] Y. Hwang, J. Kim, and E. Seo, "Structure-

Aware Web Transcoding for Mobile Devices," *IEEE Internet Computing Magazine*, Vol.7, pp. 14-21, Oct. 2003.

[12] D. Cai, S. Yu, J. Web, and W. Ma, "VIPS : a Vision-based Page Segmentation Algorithm," Microsoft Technical Report, MSR-TR-2003-79, Microsoft Research, Nov. 2003.

[13] Y. Chen, X. Xie, W. Ma, and H. Zhang, "Adapting Web Pages for Small-Screen Devices," *IEEE Internet Computing Magazine*, Vol.9, pp. 50-56, Feb. 2005.

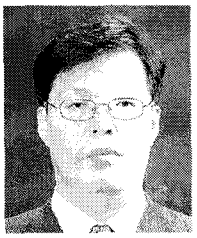
[14] Y. Chen, W. Y. Ma, and H. J. Zhang, "Detecting Web Page Structure for Adaptive Viewing on Small Form Factor Devices," Proc. 12th Int'l World Wide Web Conf., May 2003.

[15] Andrew W. Cole, Proposed Component Tools for Corpus Map Utilityto Characterize Large File Systems, Master Thesis, Dep. Computer Information Systems, Pennsylvania University, 2002.

김 수 도

- 1995년 2월 부경대학교 전자계산학과 (이학사)
- 2001년 2월 부경대학교 전산교육전공 (교육학석사)
- 2008년 8월 부경대학교 정보시스템과정 (이학박사)
- 2008년 9월 ~ 현재 부경대학교 누리사업단 계약교수

관심분야 : 모바일 웹, 적응화 기법, m-learning, u-learning



박 태 진

- 1988년 2월 동의대학교 물리학과 (이학사)
- 1995년 2월 부경대학교 전산정보학과 (이학석사)
- 2008년 8월 부경대학교 전자계산학과 (이학박사)
- 2000년 ~ 2003년 거제대학 조선정보기술계열 초빙전임강사
- 2005년 ~ 현재 마산대학 조선메카트로닉스학과 강의교수

관심분야 : 영상처리, 신호처리, 임베디드 시스템

박 만 곤

- 경북대학교 수학교육 (이학사)
- 경북대학교 수학교육 (교육학석사)
- 경북대학교 전산통계학 (이학박사)
- Philippine Women's University (국제행정학석사)
- University of Rizal System, Philippines (명예 기술학박사)
- Dept. of Electrical & Computer Engineering, University of Kansas (Post Doc.)
- 1981년 ~ 현재 부경대학교 전자컴퓨터정보통신공학부 교수
- 2008년 현재 한국멀티미디어학회(KMMS) 회장
- 2002년 ~ 2007년 정부간 국제기구 CPSC (콜롬보플랜 기술교육대학) 총재 (Director General and CEO)
- 2004년 ~ 2007년 Asia Pacific Accreditation and Certification Commission 아태지역 인증 및 검증위원회 위원장
- 2005년 ~ 2007년 유네스코 (UNESCO-UNEVOC) 자문위원, 아시아개발은행 자문관

관심분야 : 소프트웨어신뢰성공학, 비즈니스 프로세스 재공학 (BPR), 소프트웨어 공학 및 재공학, 멀티미디어정보처리기술, 정보시스템성능평가, ICT-based HRD System