

XML 문서에서 어노테이션의 위치재생성 기법 (Annotation Repositioning Methods in XML Documents)

손원성[†] 김재경^{**} 고명철^{***} 임순범^{****} 최윤철^{*****}
(Won-Sung Sohn) (Jae-Kyung Kim) (Myeong-Cheol Ko) (Soon-Bum Lim) (Yoon-Chul Choy)

요약 어노테이션 시스템에서 원본문서가 갱신되었을 경우 어노테이션이 항상 적절한 위치를 유지하기 위해서는 로버스트(robust)¹⁾한 위치재생성(repositioning) 기능이 필요하다. XML 문서환경에서 어노테이션에 대한 위치재생성을 위해서는 텍스트 정보뿐만 아니라 구조문서 특성을 포함할 수 있어야 한다. 이를 위하여 본 논문에서는 XML 기반의 원본문서 및 어노테이션 정보를 논리구조트리(logical structure tree)로 표현하고, 각 트리간의 대응관계를 분석하여 복수의 후보 앵커들을 생성한다. 또한 복수의 후보 앵커들 중 최적의 후보 앵커를 선택하기 위하여 논리구조트리 앵커 노드의 문자열(textual data) 및 레이블 정보에 기반한 단계별 앵커링 기준을 제시한다. 그 결과 본 논문에서는 구조문서 환경에서 다양한 형태의 컨텍스트 갱신이 발생하였을 경우에도 로버스트한 위치재생성 이 가능하다.

키워드 : 어노테이션, 어노테이션 위치재생성, XML, 컨텍스트, 사용자 인터페이스

Abstract A robust repositioning method is required for annotations to always maintain proper positions when original documents were modified. Robust anchoring in the XML document provides better anchoring results when it includes features of structured documents as well as annotated texts. This paper proposes robust annotation anchoring method in XML document. To do this, this work presents annotation information as logical structure trees, and creates candidate anchors by analyzing matching relations between the annotation and document trees. To select the appropriate candidate anchor among many candidate anchors, this work presents several anchoring criteria based on the textual and label context of anchor nodes in the logical structure trees. As a result, robust anchoring is realized even after various modifications of contexts in the structured document.

Key words : Annotation, Robust Annotation Repositioning, XML, Context, User Interface

1. 서론

어노테이션은 문서환경에서 문서의 주제 및 내용에 관한 해설, 설명, 그리고 강조를 목적으로 추가되는 문장 또는 텍스트를 의미한다[1,2]. 또한 웹 및 전자문서 환경에서의 어노테이션은 종이문서 환경에 비하여 어노테이션에 대한 자유로운 생성, 편집 및 삭제 그리고 분

산 다중사용자간의 어노테이션 공유 등과 같은 다양한 장점을 제공할 수 있기 때문에 여러 분야에서 활발히 응용되고 있다[1,3-7].

일반적으로 전자문서 환경의 어노테이션 시스템에서는 어노테이션을 외부링크 형태로 생성하며, 그 결과 생성된 어노테이션은 원본문서와는 별도로 시스템의 내부 및 외부에 저장된다[1]. 이러한 특성상 어노테이션이 연결되어있는 대상(target)문서의 내용이 삭제되거나 변경될 경우, 어노테이션은 앵커(anchor)의 대상을 잃기 때문에 의미적으로 동떨어진 형태의 고아(orphan data)가 된다[3,8]. 위의 문제를 해결하기 위해서는 원본문서 변경이 발생하더라도 어노테이션의 위치를 항상 재생성할 수 있는 기능이 필요하며[3,9], 실제로도 어노테이션 시스템의 매우 중요한 요소로 간주되고 있다[4].

한편 대부분의 어노테이션의 위치재생성 기법에서는 고유(unique)아이디, 서브스트링(substring), 주변정보(surrounding text) 등과 같은 컨텍스트[6,7,10]를 기본

· 이 논문은 2004년도 한국학술진흥재단의 지원에 의하여 연구되었음 (M01-2004-000-10162-0)

† 비 회 원 : Carnegie Mellon University
sohnws@cmu.edu

** 학 생 회 원 : 연세대학교 컴퓨터과학과
ki187cm@rainbow.yonsei.ac.kr

*** 비 회 원 : 건국대학교 컴퓨터응용과학부 교수
cheol@kku.ac.kr

**** 종 신 회 원 : 숙명여자대학교 멀티미디어과학과 교수
sblim@sookmyung.ac.kr

***** 종 신 회 원 : 연세대학교 컴퓨터과학과 교수
ycchoy@rainbow.yonsei.ac.kr

논문접수 : 2003년 9월 29일

심사완료 : 2005년 5월 20일

적으로 고려한다. 그러나 대부분의 어노테이션의 위치재생성(앵커링) 관련 연구에서는 원본문서 텍스트의 이동만을 고려하기 때문에 컨텍스트가 갱신되었을 경우 해당 어노테이션을 무조건 고아처리한다. 한편 어노테이션이 가장 빈번히 사용되는 사이버클래스, 이-러닝(e-Learning), 전자책 등의 환경에서는 기본적으로 XML을 원본문서로 사용한다. 따라서 XML 기반의 어노테이션 환경에서도 문서 변경에 대한 정확한 위치정보를 제공할 수 있어야 하나 대부분의 관련 연구에서는 일반 텍스트 문서만을 대상으로 한다[9]. 또한 구조 문서환경을 고려하는 기존 연구[9,11]에서는 단순히 어노테이션 및 원본문서간의 패스(path)비교를 통하여 문서의 갱신 여부만을 판별하기 때문에, 구조간의 삭제, 이동, 갱신 등이 발생하였을 경우 어노테이션의 위치를 적절히 재생성하기가 어렵다.

본 논문에서는 XML기반 어노테이션 시스템에서 다양한 문서갱신에 대해서도 위치재생성이 가능한 기법을 제안한다. 제안된 기법에서는 XML 원본 문서와 어노테이션 정보를 논리구조트리(logical structure tree)로 표현한다. 또한 위치를 재생성하기 위하여 각 트리간의 대응관계를 생성하고 이 과정에서 생성되는 복수의 후보앵커들 중 최적의 후보 앵커를 선택하기 위한 단계별 앵커링 기준을 제시한다. 제안된 앵커링 기준에서는 논리구조트리 앵커 노드의 문자열(textual data) 및 레이블 정보에 기반한 후보 앵커 생성 및 병합 등의 기능들을 포함한다. 마지막으로, 생성된 후보앵커들은 제안 시스템의 사용자 인터페이스를 통하여 최종 선택 및 추천, 그리고 고아처리 된다. 제안 인터페이스에서는 복수의

후보 앵커중 가장 적절한 1차 앵커를 사용자에게 제시하고 만일 사용자가 새로운 앵커를 요구할 경우 단계별로 적절한 후보 앵커를 추천할 수 있는 사용자 인터랙션을 제공한다.

그 결과 본 논문에서는 구조문서 컨텍스트의 삭제, 이동, 그리고 갱신이 발생하였을 경우에도 어노테이션의 위치를 적절히 재생성할 수 있다.

2. 관련연구

어노테이션의 위치재생성과 관련하여 대부분의 기존 연구에서는 원본문서 및 앵커 텍스트와 관련된 컨텍스트를 이용하며, 특히 원본문서의 고유아이디, 서브스트링 등의 컨텍스트 정보들을 포함한다. 본 절에서는 이와 관련된 기존 연구에 대하여 상세히 설명하며 이를 다음 표 1과 같이 정리하였다.

먼저 서브스트링과 관련하여 Annotator[1], ComMentor[7]에서는 어노테이션의 대상인 앵커 텍스트 중 고유한 서브스트링 및 아이디를 저장하고, 어노테이션 출력시 이러한 컨텍스트의 유무에 따라 원본 문서의 갱신 여부를 판단한다. 그러나 위와 같은 시스템에서는 원본 문서와 생성된 어노테이션간의 갱신 여부는 판별 가능하지만, 고유 컨텍스트 탐지에 실패하였을 경우 어노테이션은 무조건 고아처리(orphaning) 혹은 삭제된다.

어노테이션 위치재생성을 위한 가장 대표적인 기법은 그림 1과 같이 중복 컨텍스트(redundant context)를 이용하여 어노테이션의 위치를 탐지하는 것이다. 웹 기반의 어노테이션 시스템인 Webwise[6], CritLink[12]에서는 어노테이션 정보에 선택된 앵커 텍스트의 전체 내용

표 1 어노테이션 위치 재생성 기법 비교

시스템	컨텍스트	특징	대상문서
ComMentor[7]	고유(unique) 서브스트링	고유 서브스트링이 포함된 앵커 정보를 파트리샤 트리에 저장하여 갱신된 어노테이션 위치를 추출. 갱신위치 실패시 무조건 하단에 고아처리	텍스트
Annotator[1]	고유 서브스트링	atom으로 표현되는 고유 서브스트링을 이용하며 이를 hash 기반의 데이터베이스에 저장하여 원본문서의 갱신 여부를 판별. 갱신위치 실패시 고아처리.	텍스트
CritLink[12]	주변정보 (surrounding text)	어노테이션 내용과 주변정보 일정 부분을 동시에 저장하여 최종 위치를 판단. 갱신위치 실패시 고아처리.	텍스트
Webwise[6]	주변정보	ID, span, 주변정보, 시작지점등의 정보를 함께 저장하여 최종위치를 판별. 갱신위치 실패시 하단에 고아처리 표기.	텍스트
Robust Locations[9]	HTML 패스(ID) 및 주변정보	Tree Walk라는 트리 탐색기법 및 매칭을 이용하여 갱신 위치를 판별. 갱신위치 실패시 사용자가 위치 지정가능.	텍스트/HTML
Keyword robust anchoring[8]	키워드	앵커내부의 고유한 키워드를 추출하고 이를 이용하여 갱신된 위치를 판별. 스코어를 이용한 위치재생성 가능.	텍스트
Annotea[13]	HTML 패스(ID) 정보	Xpointer를 이용하여 구조정보에 기반한 앵커패스(ID) 정보를 생성 및 저장. 갱신위치 실패시 삭제처리.	텍스트/HTML
제안기법	구조정보	XML 문서상에서 주변정보, 패스(id), 레이블(의미정보) 등을 이용하여 어노테이션의 삭제, 이동, 갱신에도 로버스트한 위치판별 가능.	텍스트/XML /HTML

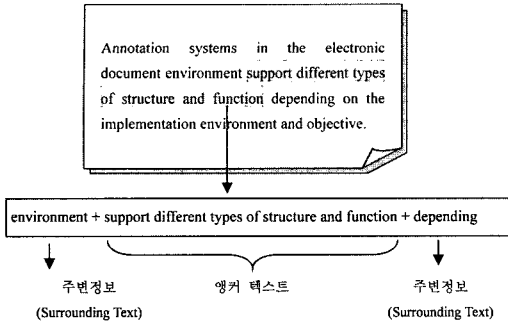


그림 1 어노테이션 위치재생성을 위한 컨텍스트의 예

과 주변정보(surrounding text)의 일정 부분을 동시에 저장한다. 다음으로 어노테이션 출력시 원본 문서와 앵커 텍스트를 비교하여 문서의 갱신 여부를 판별하고 동시에 주변정보 비교를 통하여 최종 위치를 판단한다. 위와 같은 시스템들에서는 어노테이션내부의 중복 정보를 이용하여 원본 문서 텍스트의 삭제 및 이동이 발생하였을 경우 적절한 앵커링 결과를 보장할 수 있지만 갱신이 발생하는 경우에는 무조건 어노테이션을 고아처리하거나 삭제한다. 이러한 단점을 개선하기 위해서는 텍스트 변경에 대한 위치 정보 재생성시 발생하는 복수의 앵커 후보들 중 최적의 앵커를 선택하기 위한 별도의 기법이 요구된다.

Robust Locations[9,11]에서는 앞서 설명한 중복정보 및 트리 탐색기법을 이용한다. Robust Locations에서는 가장 먼저 원본 문서와 앵커 간의 고유 아이디를 확인하여 갱신여부를 판별하고, 원본문서에서 갱신이 발생하였다면 트리 탐색을 통하여 적절한 노드를 선택한다. 그러나 Robust Locations의 기법은 HTML문서와 어노테이션간의 단순한 패스 매칭만을 수행한다. 따라서 XML과 같은 명확한 구조문서 환경에 적용하기 위해서는 구조정보의 삭제, 이동 및 갱신이 발생하는 경우에도 적절한 위치를 보장할 수 있는 새로운 기법이 요구된다.

한편 Keyword robust anchoring[8]에서는 기존 기법과는 달리 키워드 기반의 앵커링 기법을 제공한다. Keyword robust anchoring에서는 어노테이션 내부의 고유한 키워드를 추출하고 앵커 위치를 탐지한다. Keyword robust anchoring에서는 원본 문서의 갱신되었을 경우 저장된 어노테이션 정보와 원본 문서 내부의 키워드 스패(keyword span)를 추출하고, 추출된 앵커들이 복수로 존재할 경우 키워드 스패를 기준으로 최종 앵커 위치를 선택한다. 그러나 Keyword robust anchoring 기법은 일반 텍스트 문서만을 대상으로 하며, 방대한 규모의 웹 환경에서도 항상 원본 문서에 대한 키워드 추출 과정을 반드시 필요로 한다.

Annotea[13]에서는 Xpointer를 이용하여 구조정보에 기반한 앵커 패스 정보를 생성한다. 그러나 Annotea에서는 원본 문서의 갱신이 발생할 경우 앵커 노드의 아이디를 검색하여 해당 아이디가 존재하지 않는 경우 무조건 앵커 노드의 부모노드를 새로운 어노테이션 위치로 지정하는 가장 기초적인 방법만을 포함하고 있다.

3. 어노테이션 위치재생성 기법

본 연구에서는 XML 환경에서 적절한 위치를 보장하기 위하여 XML 문서 뿐만 아니라 하이퍼텍스트 환경에도 적용가능한 다음 그림 2와 같은 어노테이션 처리 기법을 제안한다.

제안된 기법에서는 문서구조의 변경 유무에 따라 먼저 원본문서 순회를 통하여 생성된 어노테이션 정보와 문서 구조간의 일치 여부를 확인하고, 대상 문서의 갱신 정도를 판별한다. 그 결과 원본 문서와 어노테이션간의 구조가 동일하면서 텍스트의 내용만 갱신될 경우 그림 2의 왼쪽부분처럼 노드간의 일대일 레이블 비교가 가능하며 동시에 LCSR(Longest common subsequence rate)를 이용한 텍스트 비교과정을 수행한다. 또한 노드간의 구조가 변경될 경우는 그림 2의 오른쪽 부분과 같이 LCSR을 이용한 갱신 정도에 따라 각기 적합한 노드간의 대응관계를 생성하며, 이때에 발생하는 1-1, 1-N, N-1, N-N 대응관계의 단계에 따라 적절한 후보 앵커들을 생성한다. 이에 대한 자세한 설명은 다음 절에서 설명하도록 한다.

한편 본 연구에서는 구조문서 환경에서의 앵커링을 위하여 기존 구조문서 환경에서의 변화탐지 기법 중 특히 패스 매칭(path matching) 기법[14]을 응용한다. 그러나 어노테이션 환경의 매칭과정에서는 하나의 앵커 레이블에 서로 유사한 복수의 텍스트 노드들이 빈번히 발생하며, 이러한 복수의 노드들 중 하나의 완전한 앵커

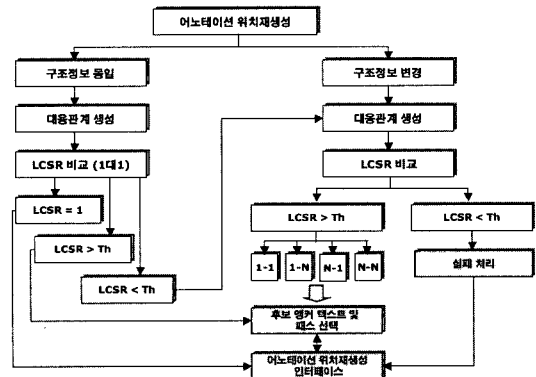


그림 2 제안 논문의 어노테이션 위치재생성 과정

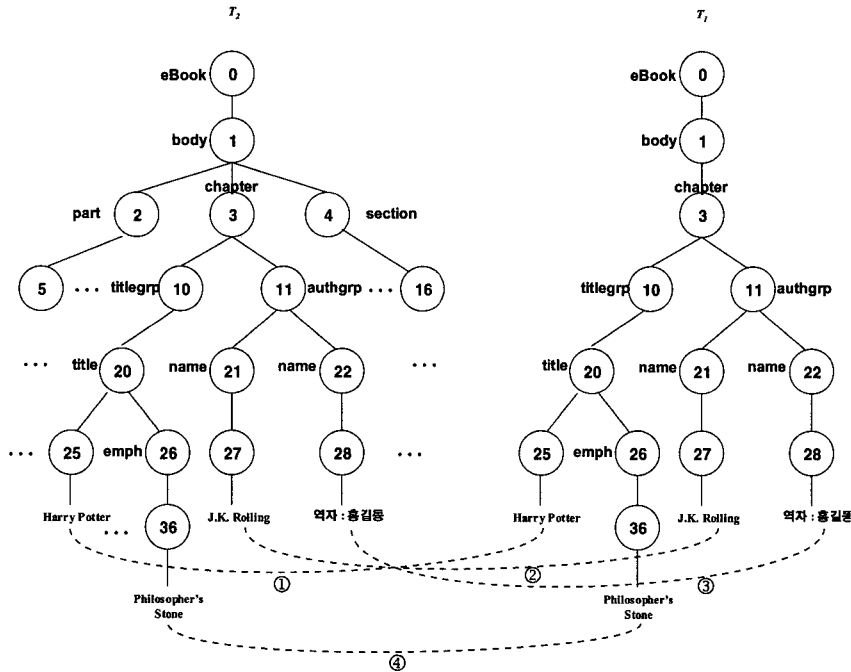


그림 3 원본문서와 입력된 어노테이션 정보에 대한 논리구조트리 표현 예

노드를 선택하는 것은 불가능하다. 따라서 본 논문에서는 논리구조트리의 패스 및 어노테이션의 정보 간의 복합적인 변화를 고려하며, 동시에 일-대-다, 다-대-다와 같은 대응관계를 단계별로 생성하여 어노테이션환경에 적합한 결과를 얻고자 하였다. 제안 기법의 상세설명은 다음절에서 설명하도록 한다.

3.1 논리구조트리 생성

본 시스템에서는 XML 기반의 전자책 표준[2]을 원본문서로 사용하며, 이러한 원본문서에 어노테이션의 위치 정보를 부여하기 위하여 다음 그림 3과 같은 논리구조트리를 생성한다. 논리구조트리의 각 노드 레이블은 어노테이션 컨텍스트의 논리정보(path)의 기본요소에 해당되며, 리프노드의 PCDATA는 앵커텍스트의 대상인 앵커노드가 된다. 또한 논리구조트리에서는 각 노드 레이블 및 깊이(depth) 별 순번(sequential number), 단말노드(anchor node)까지의 경로(path), 그리고 단말노드 텍스트의 오프셋(offset)을 이용하여 원본문서에 대한 컨텍스트 정보를 표현한다.

앞서 설명한 바와 같이 논리구조트리 형태의 원본문서에 대한 어노테이션 입력은 XML 정보로 표현되며 이는 그림 3의 문서 T_1 와 같은 정보로 표현가능하다. 따라서 어노테이션의 위치 재생성을 위해서는 어노테이션 정보 T_1 의 앵커 노드 (25, 27, 28, 36) 와 원본 문서 T_2 의 리프 노드들간 갱신 여부 판별 및 위치재생성 과

정을 필요로 한다. 이러한 내용은 다음절에서 자세히 설명하도록 한다.

3.2 구조 정보가 동일한 문서의 어노테이션 위치재생성

제안기법에서는 그림 2와 같이 원본문서 구조의 갱신 여부에 따라 상이한 위치재생성 과정을 수행한다. 이를 위하여 먼저 원본 문서의 논리구조트리를 순회하면서 어노테이션 앵커의 패스 및 오프셋, 그리고 텍스트 정보들이 원본문서에 어떠한 형태로 존재하는지를 확인한다. 만일 어노테이션 패스정보가 원본 문서에 동일한 형태로 존재한다면 어노테이션 텍스트간의 갱신 여부만을 판별한다.

한편 본 연구에서는 어노테이션의 텍스트와 원본문서의 텍스트 노드간의 갱신 여부를 판별하기 위하여 LCS(Longest common subsequence)에 기반한 다음의 수식 (Longest common subsequence Rate)[15]을 이용한다.

$$LCSR(x, y) = \frac{2 \times |lcs(x, y)|}{|x| + |y|} \quad (1)$$

본 논문에서는 위와 같이 어노테이션 패스와 원본문서간의 구조가 갱신되지 않은 경우는 다음 위치재생성 기준 1 및 2를 적용하여 적절한 앵커를 추출하며 자세

1) $|lcs(x, y)|$, $|x|$, 그리고 $|y|$ 는 각각 문자열 x 와 y 간의 LCS의 길이, x 의 길이, 그리고 y 의 길이를 나타낸다.

한 내용은 다음과 같다.

위치재생성 기준 1 : 어노테이션 정보인 T_1 과, 어노테이션의 대상 문서인 T_2 에 대하여 각각 어노테이션의 앵커 텍스트 노드 $T_1 = \{x_i\}$, $1 \leq i \leq s$,와 원본문서의 텍스트 노드 $T_2 = \{y_i\}$, $1 \leq i \leq a$,가 존재한다고 하자. 또한 각각의 텍스트 노드들은 문자열 $x_i = \{T_1strin\}$, $1 \leq n \leq t$ 및 $y_i = \{T_2strin\}$, $1 \leq n \leq \beta$ 을 포함하고 각각의 문자열들은 캐릭터 $T_1str_{ik} = \{a_{ik}\}$, $1 \leq n \leq k$, 및 $T_2str_{ik} = \{b_{ir}\}$, $1 \leq n \leq \gamma$,를 포함한다. 이때에 어노테이션 정보 T_1 의 모든 노드들이 레이블 및 순서가 동일한 형태로 T_2 에 존재하고, 동시에 서로 부모노드 레이블이 동일한 T_1 의 앵커 텍스트 노드 x_{i_0}, \dots, x_{i_p} ,와 T_2 의 텍스트 노드 y_{i_q}, \dots, y_{i_r} ,가 존재한다면 대응관계 $[x_{i_0}, y_{i_q}], \dots, [x_{i_p}, y_{i_r}]$ 를 생성한다.

기준 1을 만족한다면 원본 문서의 패스는 변경되지 않았다고 가정할 수 있기 때문에 각 앵커 텍스트간의 일-대-일 대응관계 생성이 가능하다. 예를 들어 그림 4와 같이 원본문서의 구조가 변경되지 않은 경우에는 원본문서와 어노테이션 패스간의 일-대-일 대응관계인 대응관계 [25, 25], ①, 및 [27, 27], ②, [28, 28], ③, 그리고 [36, 36], ④의 생성이 가능하다.

기준 1에 의해서 대응관계가 생성되었다면, 각 앵커 텍스트간의 유사도를 측정하고 그 결과에 따라 어노테이션 위치를 설정한다. 이에 대한 내용은 다음 기준 2에

서 설명하고 있다.

위치재생성 기준 2 : 기준 1에 의하여 생성된 대응관계간 노드들의 문자열 유사도가 1인 경우는 원본 문서의 텍스트 노드 y_{i_q}, \dots, y_{i_r} ,를 앵커 영역으로 지정하며, 만일 $LCSR$ 이 1 이하이면서 일정 임계값 이상일 경우, 텍스트 노드 y_{i_q}, \dots, y_{i_r} ,를 후보 앵커로 간주한다. 한편 노드들의 $LCSR$ 이 일정 임계값 이하인 경우는 구 버전의 앵커 정보가 삭제되거나 구조가 변경된 경우임으로 다시 기준 3~6 들을 적용한다. 만일 기준 3~6을 적용하여 조건을 만족하는 T_2 의 텍스트 노드가 존재하지 않는다면 T_2 의 텍스트 노드 y_{i_q} 를 후보 앵커로 지정한다.

위치재생성 기준 1, 2를 적용한 결과 패스가 변경되지 않고, 앵커 텍스트가 동일 혹은 갱신되었을 경우 기존 영역을 그대로 사용하거나, 새로운 후보 앵커 영역을 추출할 수 있다. 예를 들어 다음 그림 4의 대응관계 [25, 25], ①,은 $LCSR$ 1의 관계이기 때문에 바로 어노테이션 앵커로 지정 가능하며, 대응관계 [27, 27], ②,와 [28, 28], ③,은 $LCSR$ 이 1이하이면서 일정 임계값 이상을 포함하므로 T_2 의 노드 [27],과 [28]을 후보 앵커영역으로 지정한다. 한편 대응관계 [36, 36], ④,은 유사도가 임계값의 기준을 벗어나므로 다음에서 설명할 새로운 앵커 기준을 적용하도록 한다. 만일 대응관계 [36, 36], ④,가 기준을 만족하지 않는다면 이는 어노테이션의 정보가 완전히 수정된 경우이므로, <emph> 엘리먼트의 텍

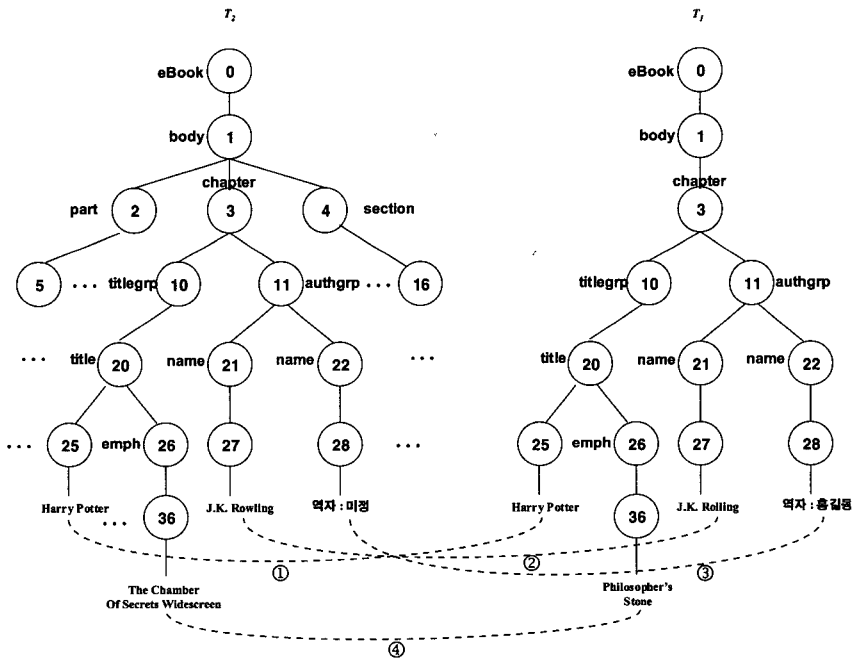


그림 4 패스정보가 동일한 앵커 노드간의 위치재생성의 예

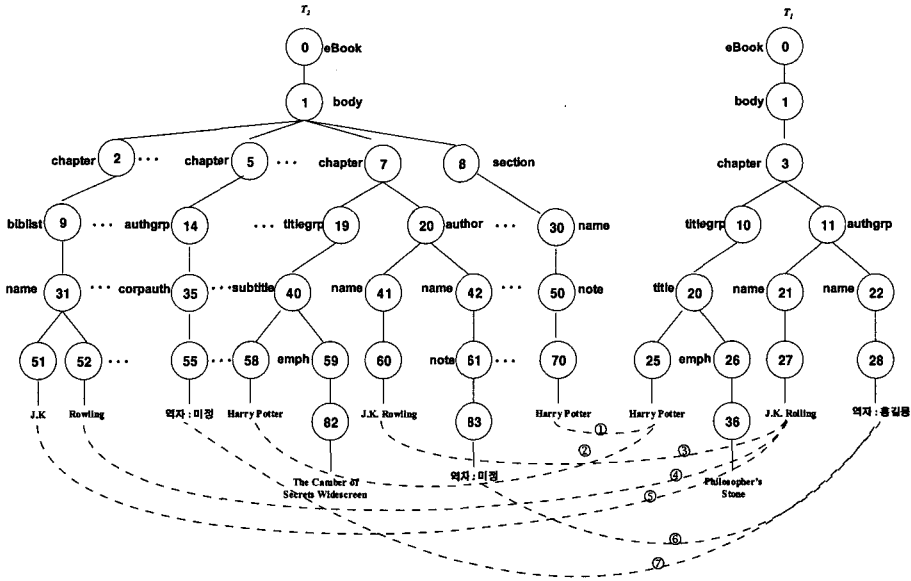


그림 5 문자열 유사도에 근거한 후보 앵커 생성의 예

스트 노드를 후보 앵커 지점으로 선택한다. 어노테이션 위치재생성 기준으로 선택된 후보 앵커들은 제안 인터페이스의 사용자 인터랙션을 통하여 최종 어노테이션 앵커로 선택된다.

3.3 구조 정보가 상이한 문서간의 어노테이션 위치재 생성

원본문서와 어노테이션간의 패스 및 앵커 텍스트의 변화가 발생하였을 경우는 앞서 설명한 3.2의 기준과 같이 앵커 텍스트간의 비교뿐만 아니라 원본문서와 어노테이션 앵커간의 연결성을 극대화하기 위하여 구조간의 유사성등을 고려한다. 또한 이러한 정보에 의하여 추출된 복수의 앵커들은 어노테이션 특성 상 앵커 텍스트 및 패스에 대한 일-대-일, 일-대-다, 다-대-다, 다-대-일과 같은 복수의 후보들을 포함한다. 따라서 본 연구에서는 이러한 복수의 후보들에 대한 적절한 선택을 위하여 단계별 경로 매칭 및, 복수 후보간의 병합, 링크 등의 기법들을 적용한다. 상세한 설명은 다음과 같다.

위치재생성 기준 3 : DOM 트리 순회를 통하여 어노테이션 정보인 T_1 의 패스 정보가, 어노테이션의 대상 문서인 T_2 의 노드 레이블 및 형제 순서 등과 일치하지 않는 경우가 발생한다고 하자. 이때에 T_1 의 어노테이션 앵커 텍스트 노드인 x_{io}, \dots, x_{ip} 와 T_2 의 모든 텍스트 노드간의 LCSR을 추출하여 일정 임계값 이상이 되는 T_2 의 텍스트 노드를 선택하고 이와 관련된 대응관계 $[x_{io}, y_{iq}], \dots, [x_{io}, y_{ir}], \dots, [x_{ip}, y_{iu}], \dots, [x_{ip}, y_{iv}]$ 등을 생성한다.

원본문서에서 구조 및 텍스트 노드의 내용이 갱신되

었다면 먼저 기준 3을 이용하여 LCSR이 일정 이상되는 대응관계들을 생성한다. 그 예로 다음 그림 5처럼 노드 25에 대해서는 대응관계 [25, 70], ①, 및 [25, 58], ②를 생성하며, 같은 기준에 근거하여 대응관계 ③, ④, ⑤, ⑥, ⑦을 생성할 수 있다.

한편 생성된 텍스트 노드간의 대응관계들은 그림 8의 경우처럼 복수의 앵커 노드 대 복수의 텍스트 노드들로 대응될 수 있다. 제안기법에서는 이러한 경우들이 발생한다면 단순히 LCSR에 근거하여 하나의 대응관계를 선택하기 보다는, 노드 레이블간의 LCSR을 비교하여 의미적으로 연관되는 새로운 대응관계를 생성한다. 이러한 내용은 다음 기준 4와 같다.

위치재생성 기준 4 : 어노테이션의 앵커 노드에 대한 원본문서의 LCSR이 일정 이상이 되는 대응관계들이 복수로 존재한다면, 대응하는 각각의 텍스트 노드의 경로²⁾ 간 레이블 LCSR을 비교하여 일정값 이상의 레이블 유사도를 포함하는 노드들의 대응관계 $[x_{io}, y_{iq}], \dots, [x_{io}, y_{ir}], \dots, [x_{ip}, y_{iu}], \dots, [x_{ip}, y_{iv}]$ 를 생성한다.

위치재생성 기준 4에 의하여 문자열 유사도에 근거한 복수의 대응관계들 중 레이블 유사도가 일정이상인 대응관계들은 새로운 후보 앵커로 지정된다. 예를 들어 다음 그림 6에서 검은색 라인의 대응관계 [25, 58], ②, [27, 60], ③, [27, 52], ④, [27, 51], ⑥, 그리고 [28, 83], ⑦은 경로간 레이블 LCSR이 일정 이상이 되는 경우들

2) 경로(x)란 노드 x의 부모 노드로부터 뿌리노드까지의 노드의 순차적인 집합을 의미한다.

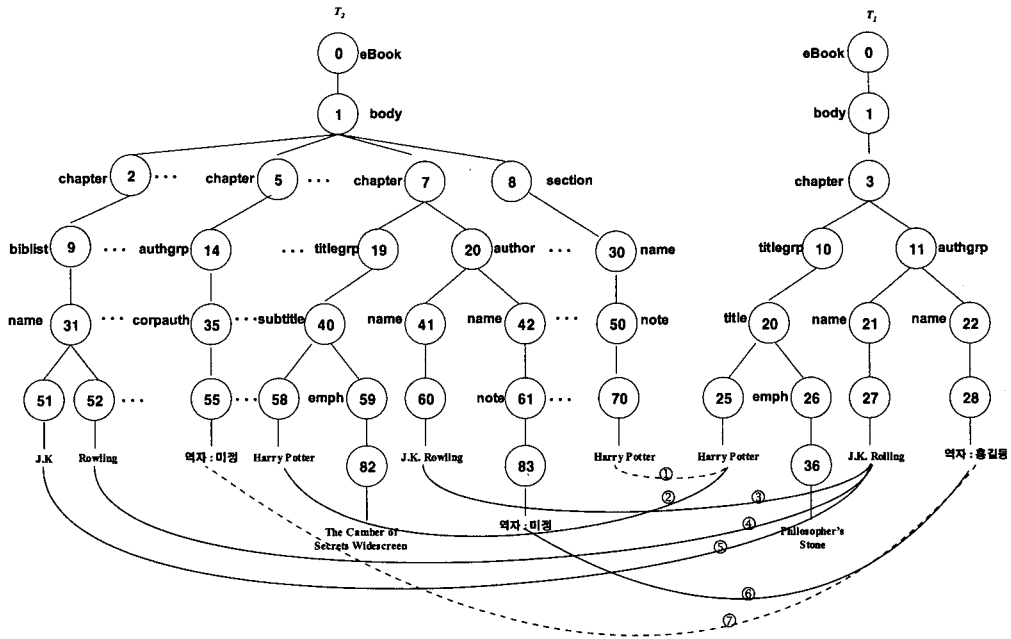


그림 6 레이블 유사도에 의한 후보 앵커 생성의 예

이며, 이들을 새로운 대응관계로 지정한다. 이와 같이 기준 3을 만족하는 동시에 기준 4를 만족하는 복수의 후보 앵커들이 존재한다면 제안 기법에서는 이들을 보다 높은 우선 순위의 후보 앵커로 간주하며, 차후 인터페이스에서 앵커 추천 인터랙션의 기준으로 사용한다.

기준 4에 의하여 생성된 대응관계들 중 동일한 레이블 유사도를 포함하는 대응관계가 발생한다면 유사도 정보 및 레이블간의 인접관계등을 이용하여 대응관계간의 병합가능 여부를 판단하고 이를 통하여 대응관계간의 경우의 수를 보다 줄일 수 있도록 한다. 이러한 내용은 다음 기준 5와 같다.

위치재생성 기준 5 : 원본 문서의 텍스트 노드 y_{i1}, \dots, y_{ir} 에서 생성 기준 4를 적용한 결과 경로간 레이블 유사도가 일정 이상이 되는 노드들이 복수로 존재한다면 이들간의 병합 적합성을 판별한다. 병합 적합성을 판별하기 위하여 먼저, 텍스트 노드가 서로 동일한 부모 노드를 갖는 형제 관계이거나 또는 텍스트 노드가 일련의 순서를 갖는다면, 해당 노드를 병합하여 후보 앵커 영역으로 간주한다.

기준 4에 의하여 생성된 복수의 대응관계 중 기준 5의 병합 기준을 만족하는 관계들은 하나의 앵커영역으로 병합하고 이것을 새로운 후보 앵커로 선택한다. 예를 들어 다음 그림 7에서 대응관계 [27, 51]과 [27, 52]는 서로 동일한 부모 노드를 갖는 형제 관계이며, 대응관계 [27, 60]과 [28, 83]의 텍스트 노드들은 원본문서에서 일

련의 순서를 갖기 때문에 각각 하나의 앵커 영역으로 병합하여 이를 참조하도록 한다.

만일 기준 5에 의하여 병합된 노드 이외에 또 다른 대응관계가 존재한다면 제안기법에서는 이것을 새로운 후보 앵커로 선택하고 어노테이션의 앵커노드에서 이것을 참조하도록 하며 그 내용은 다음 기준 6과 같다.

위치재생성 기준 6 : 병합 적합성을 포함하지 않는 복수의 텍스트 노드 y_{i1}, \dots, y_{ir} 들은 물리적으로 근접하거나 유사하지 않지만 서로 최소한의 의미적인 연관성을 갖을 수 있으므로 이를 링크로 연결된 형태의 후보 앵커 영역으로 간주한다.

예를 들어 다음 그림 8에서 병합된 대응관계 [27, 60], [27, 51]를 후보 앵커로 선택하고 동시에 병합할 수 없는 대응관계 [27, 70]가 존재한다면 제안 기법에서는 이러한 노드들을 어노테이션 앵커에서 참조할 수 있도록 한다. 이러한 대응관계들은 인터페이스에서 1차적으로 선택된 앵커 노드이외에 추가로 사용자에게 추천할 경우 사용된다.

한편 제안기법에서는 보다 적절한 의미있는 노드를 제공하기 위하여 경로간 레이블 LCSR의 임계값을 만족하지는 않지만 텍스트(textual data) LCSR이 어느정도 유사한 노드들에 대해서도 후보 앵커로서의 자격을 부여한다. 이를 위해서는 다음 기준을 적용한다.

위치재생성 기준 7 : 만일 T2의 텍스트 노드에서 기준 3을 만족하지만 기준 4를 만족하지 않는 대응관계들

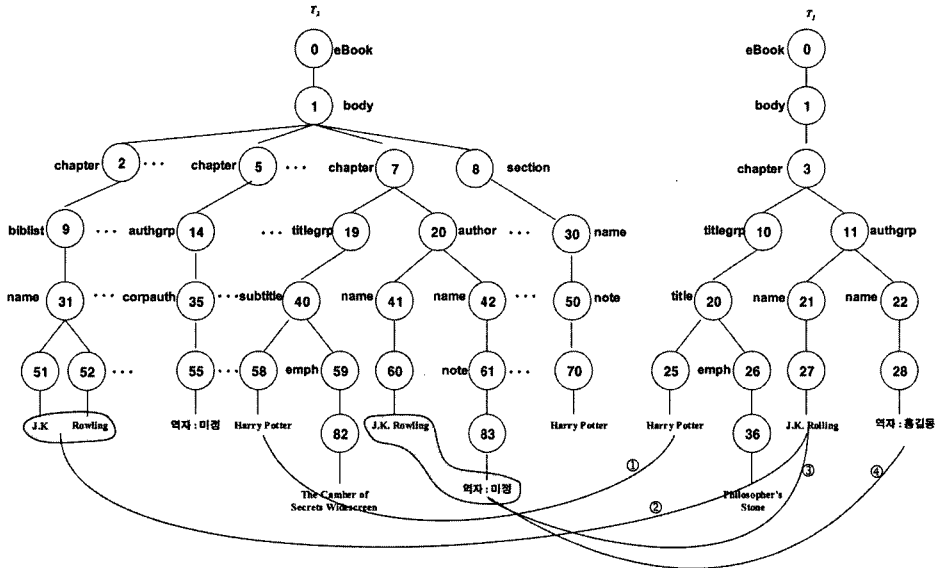


그림 7 노드 병합에 의한 후보 앵커 생성의 예

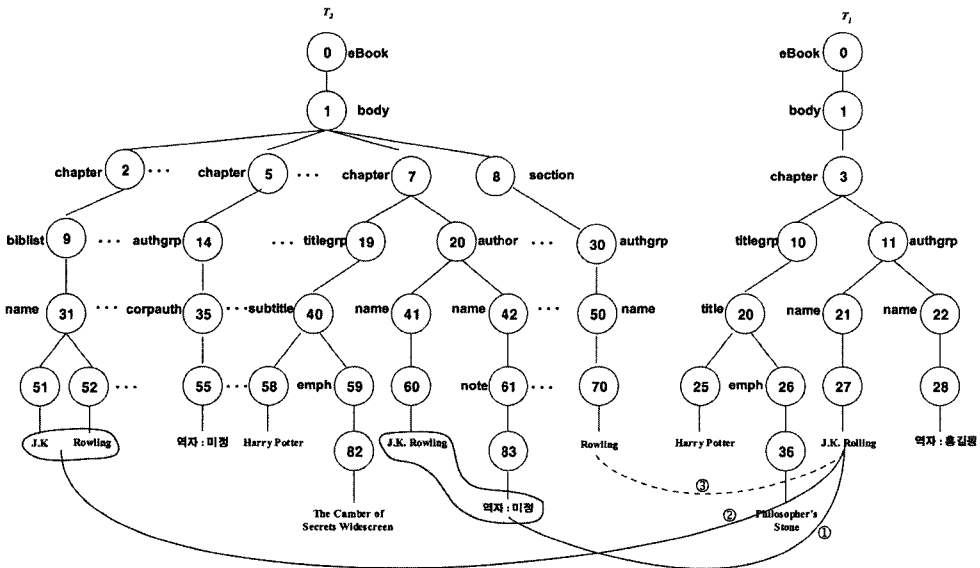


그림 8 기준 6에 의하여 생성된 후보 앵커의 예

이 존재한다면 이러한 대응관계들 중 서로 부모노드가 동일한 노드들을 우선적으로 후보 앵커로 간주하고 나머지 대응관계들에 대해서는 텍스트 LCSR을 기준으로 후보 앵커로서의 자격을 부여한다.

다음 그림 9에서 T_2 의 앵커 노드 27에 대한 대응관계 중 앵커노드간 LCSR이 일정이상이면서 레이블 LCSR이 일정 이하인 대응관계는 [27, 55], ①, [27, 60], ②, [27, 82], ③, 그리고 [27, 83], ④와 같다. 이러한 복수의 대응

관계중에서 먼저 부모노드가 일치하는 대응관계 ①을 가장 우선적으로 후보 앵커로 선택하며, 계속하여 LCSR을 기준으로 대응관계 ②, ③, ④를 차례로 선택한다.

한편 지금까지 살펴본 기준들을 만족하지 않는 텍스트 노드들이 원본문서에 존재한다면 이는 원본문서의 구조 및 앵커 텍스트가 심하게 변경되었거나 삭제된 경우를 의미한다. 따라서 제안 기법에서는 이러한 노드들에 대하여 인터페이스를 통하여 고아처리한다.

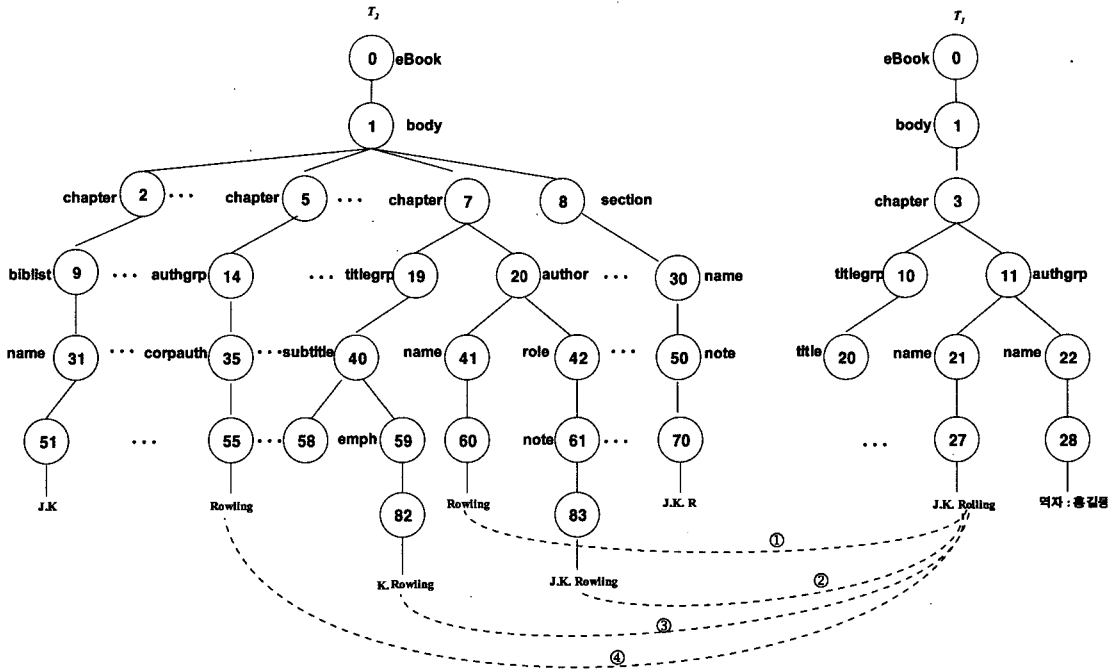


그림 9 기준 7에 의한 후보 앵커 생성의 예

4. 시스템 구현

본 절에서는 제안 기법 및 인터페이스를 포함한 어노테이션 시스템 구현 결과를 설명한다. 본 시스템은 XML 기반의 전자책표준[2]을 사용하였으며, Window XP 및 Window CE 환경에서 구동된다. 본 내용에서는 Window CE 환경에서 실행되는 어노테이션 시스템을 설명하며 전체 과정은 다음 그림 10과 같다.

제안 시스템에서는 그림 10과 같이 XML 기반의 원본문서에 대한 논리구조트리(logical structure tree) 생성과 논리구조 트리에 대한 어노테이션 입력, 생성된 어노테이션 정보에 대한 구조화, 그리고 입력된 어노테이션과 갱신된 원본문서간의 위치재생성 과정을 수행한다.

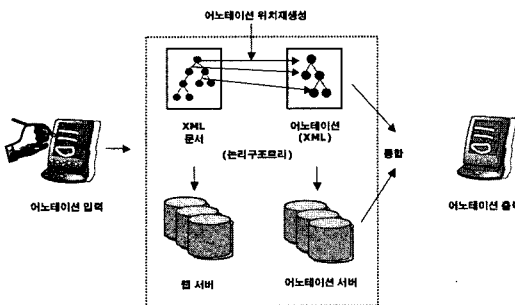


그림 10 제안 어노테이션 시스템의 처리과정

한편 다음 그림 11(a)에서는 제안 시스템의 어노테이션 브라우저에서 입력된 어노테이션 정보들을 나타내고 있으며 총 4개의 하이라이팅과 1개의 노트를 삽입하였다. 그림 11(b)에서는 변경된 그림(a)의 대상문서에 대하여 제안 기법 및 인터페이스를 적용하여 위치재생성 기능을 제공하고 있는 내용을 보여주고 있다. 그림 11(b)에서는 먼저 그림 11(a)의 어노테이션 ①,②,⑤에 대한 1차 위치정보를 박스(rectangle) 형태의 어노테이션으로 출력하며 특히 LCSR에 의하여 추출된 공통된 앵커 텍스트를 하이라이트로 포함한다. 이는 사용자에게 어노테이션의 위치선택의 기준을 제공하기 위함이다.

또한 그림 11(a)의 ③ 어노테이션의 경우는 그림 11(b)에서 앵커의 구조는 동일하지만 앵커 텍스트의 내용이 심하게 변경되었기 때문에 제안 인터페이스에 근거하여 앵커노드의 엘리먼트 영역을 1차 앵커로 출력하고 다음 그림 12(a)와 같이 별도의 아이콘을 통하여 기존 어노테이션의 내용을 제공한다. 그림 11(a)의 ④는 앵커의 구조 및 내용이 동일하므로 별도의 위치재생성 기법이 필요하지 않다.

한편 그림 11의 어노테이션 ②는 구조와 앵커 텍스트의 내용이 동시에 변경되어 구조 레이블 및 문자 유사도에 근거하여 가장 우선순위가 높은 후보 어노테이션들을 시스템에서 제공한 경우이지만, 사용자가 이러한 1

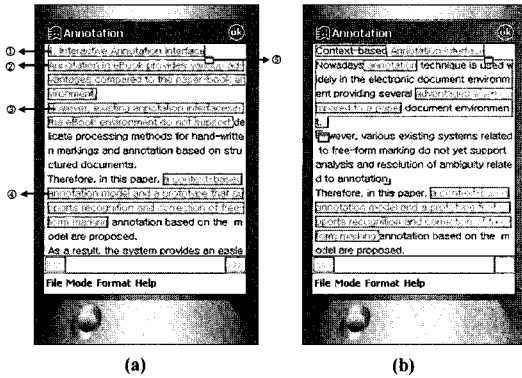


그림 11 어노테이션을 입력한 원본문서(A)와 갱신된 원본문서에 대한 위치재생성 결과(B)

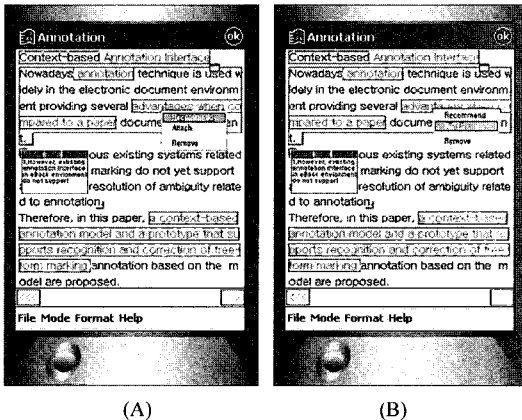


그림 12 제안 인터페이스에서의 후보 앵커 추천(a) 및 'attach'(b) 과정

차 앵커에 만족하지 않는다면 그림 12의 (a)와 같은 추천 기능을 이용하여 새로운 앵커를 선택하거나, 그림 12의 (b)처럼 사용자가 직접 'attach' 메뉴를 이용하여 최종 앵커 영역을 선택할 수 있도록 한다.

5. 실험결과

본 논문에서는 제안 기법에 대한 성능을 평가하기 위하여 총 3가지의 경험적 평가방법(empirical user test)을 수행하였다. 실험 1에서는 제안 시스템과 기존 시스템간의 위치 재생성 정확도를 비교하기 위한 사용자 평가를 수행하였다. 특히 제안 시스템과 기존 시스템 중 구조정보를 이용하는 Robust Location[9], 그리고 주변 정보(surrounding text)만을 사용하는 WebVise[6]의 기법을 적용한 프로토타입을 이용하였으며, 어노테이션에 대한 재위치 결과의 정확도 측정은 컴퓨터상에서 정량적인 측정이 어렵기 때문에, 3가지 프로토타입이 제시

한 앵커링 정확도를 사용자들이 측정하도록 하였다. 또한 실험 2에서는 제안 시스템의 단계별 위치재생성 기준이 사용자들이 판별한 정확도에 어떠한 영향을 미치는지를 확인해본다.

5.1 실험 1

본 실험에서는 XML 문서환경에서 제안 기법과 기존 기법 중 구조정보를 이용하는 Robust Location의 기법과 구조 정보를 이용하지는 않지만 주변정보 및 옵션정보등을 이용하는 WebVise의 기법을 적용한 프로토타입을 이용하였다. 사용자들은 3가지 프로토타입을 이용하여 XML 기반의 전자책 문서에 대하여 총 50개의 어노테이션을 생성하고 다시 원본문서를 변경시켜 최종적으로 구조와 앵커가 변경된 어노테이션20개, 앵커의 내용만 변경된 어노테이션 20개, 그리고 전혀 수정되지 않은 어노테이션 10개를 발생시켰다. 이러한 갱신내용에 대하여 각 프로토타입에서는 각기 다른 기법에 따라 적절한 앵커링을 생성하고 이를 사용자에게 출력하였다.

또한 본 실험에서는 사용자들이 판단한 각 프로토타입의 위치재생성 정확도를 평가하기 위하여 1(최저 정확도) 부터 10(최고 정확도)을 포함하는 설문지를 이용한 설문 조사를 실시하였다. 본 실험에는 10명의 참여자가 (남자 7명, 여자 3명, 대학원생)가 참여 하였으며, 실험을 위해서는 Compaq PDA와 Windows CE 환경에서 구현된 프로토타입을 사용하였다. 설문 항목은 크게 3가지 질문을 포함한다. 먼저 사용자가 생성한 총 50개 어노테이션의 위치와 문서 변경후 새롭게 변경된 어노테이션 위치에 대하여 어느정도 만족하는지를 파악하기 위하여 정확도를 선택하는 항목을 포함한다. 또한 자신이 어노테이션 텍스트에서 가장 중요하게 생각하는 키워드 및 기타 정보를 표기하도록 하였으며, 마지막으로 정확도 7이하의 경우 사용자에게 직접 그 이유와 자신의 의견을 작성하도록 하였다.

실험자들의 PDA 사용에 대한 전문성이 실험 결과에 영향을 미칠 수 있으므로, 본 실험에서는 PDA에 대한 사용 경험이 있는 실험자들만을 고려하였으며, 동시에 프로토타입에 대한 간략한 인터페이스 설명과 각 10분간의 사용시간을 제공하였다. 그러나 PDA 사용경험 및 인터페이스의 친숙도에 따라 실험 결과가 상이할 수도 있으며 데스크탑 환경에서 본 프로토타입을 평가할 경우 새로운 결과가 나올 수 있기 때문에 향후 이에 대한 새로운 평가가 필요할 것으로 예상된다.

각 프로토타입에 따른 사용자들의 정확도를 분석하기 위하여 본 실험에서는 일원 분산분석 (Single-Factor ANOVA)을 사용하였으며 그 결과는 다음 그림 13과 같다. 그림 13은 50개의 어노테이션 위치를 재생성한 것에 대한 사용자 평가의 평균을 나타내고 있으며 대체적

으로 제안 기법이 기존 기법보다 높은 정확도를 유지할 수 있었다. 또한 실험결과는 전체적으로 유의함($F(2,57) = 8.98, P < 0.05$) 이 있었음을 확인할 수 있었다.

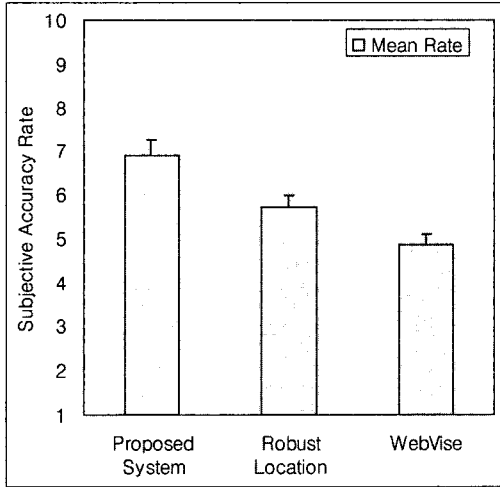


그림 13 각 프로토타입의 위치재생성 결과에 대한 사용자 정확도(1 = 최저, 10 = 최고)

실험결과 구조문서 환경에서 본 논문에서 제안한 시스템 결과가 사용자들에게 가장 높은 정확도를 얻을 수 있었으며, 특히 Robust Location 기법의 구조 정보를 이용한 Tree Walk 기법보다 제안 기법의 단계별 후보 앵커 생성기법이 사용자의 정확도 판단에 효과적으로 영향을 미쳤음을 알 수 있었다. 한편 구조 정보를 고려하지 않고 주변정보(surrounding text) 만을 사용하는 기법은 제안 기법과 비교하여 다소 낮은 사용자 만족도를 나타내었다. 이는 구조문서환경에서는 문자열간의 유사도뿐만 아니라 엘리먼트의 의미적 관계, 즉 패스 경로와 같은 정보가 앵커링 결과에 영향을 미치고 있다고 판단된다.

5.2 실험 2

본 실험에서는 제안 기법에서 사용하는 단계별 위치재생성 기준이 사용자들의 정확도 판단에 어떠한 영향을 미치는지를 평가해보았다. 이를 위하여 제안 기법 중 문자열 유사도만을 사용하는 앵커링 기준 3을 적용한 경우(10개의 어노테이션)와 문자열 유사도 및 구조간의 패스 유사도를 포함하는 앵커링 기준 3과 4를 포함하는 경우(10개의 어노테이션), 그리고 위의 앵커링 기준 및 기타 병합 기준 등을 모두 포함하는 경우(10개의 어노테이션)를 포함하는 각각의 프로토타입을 사용하였으며 각 프로토타입의 앵커링 결과에 대한 정확도를 사용자 평가를 통하여 분석해보았다. 본 실험에서는 총 10명의 참여자가 (남자 7명, 여자 3명, 대학원생)가 참가하였

으며, 실험 1과 마찬가지로 1부터 10까지의 평가기준을 포함하는 설문 조사를 실시하였다.

각 프로토타입에 따른 사용자들의 정확도를 분석하기 위하여 본 실험에서는 일원 분산분석 (Single-Factor ANOVA)을 사용하였으며 그 결과는 다음 그림 14와 같다. 그림 14는 각각의 위치재생성 기준을 적용한 것에 대한 사용자 평가의 평균을 나타내고 있으며 전체적으로 유의함 ($F(2,57) = 7.40, P < 0.05$) 이 있었음을 확인할 수 있었다.

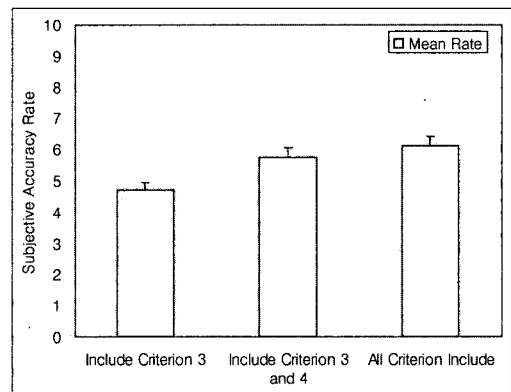


그림 14 서로 다른 기준적용에 따른 위치재생성 결과의 사용자 정확도(1 = 최저, 10 = 최고)

본 실험결과 제안 기법의 모든 앵커링 기준을 적용한 경우에 사용자들이 가장 높은 만족도를 보였으며 다음으로 패스 유사도, 그리고 문자열 유사도만을 적용한 순으로 만족도를 나타내었다. 이는 구조문서 환경에서는 구조 정보간의 의미적 관계가 적절한 어노테이션 지점을 판단하는 중요한 요소가 될 수 있다는 것을 의미한다. 또한 제안 기법의 병합 기준은 복수의 후보 앵커들을 최소화 함으로써 사용자들이 적절한 어노테이션 위치를 판단하는데에 영향을 미치는 것으로 판단된다. 결과적으로 구조문서 환경에서의 위치재생성은 앵커 텍스트 및 노드간의 의미적 정보에 의해 많은 영향을 받는다는 것을 알 수 있다.

6. 결론 및 향후연구

본 연구에서는 XML 문서 환경에서 문서 변경에 대한 어노테이션 위치재생성을 위한 처리 기법 및 시스템을 제시하였다. 제안된 기법에서는 XML 원본 문서와 어노테이션 정보를 논리구조트리로 표현한다. 또한 적절한 위치정보를 탐지하기 위하여 트리간의 대응관계를 생성하고 이 과정에서 생성되는 복수의 후보 앵커들 중 최적의 후보 앵커를 선택하기 위한 단계별 앵커링 기준

을 제시하였다. 제안 앵커링 기준에서는 구조문서의 앵커 텍스트 정보에 대한 유사도를 이용하여 1차적인 앵커링 지점을 찾고자 하였으며, 동시에 구조정보에 대한 의미적인 정보를 추출하고자 경로에 대한 유사도를 기준으로 복수의 후보 앵커를 생성하였다. 또한 복수의 후보 앵커들의 영역을 보다 효과적으로 판단하기 위하여 병합기준을 제공하였다. 마지막으로, 생성된 후보앵커들은 제안 논문의 사용자 인터랙션을 통하여 최종 선택 및 추천, 그리고 고아처리 된다.

그 결과 본 논문에서는 구조문서에서 컨텍스트의 삭제, 이동, 그리고 갱신이 발생하였을 경우에도 적절한 어노테이션의 위치를 재설정할 수 있다. 제안된 인터페이스에서는 인지적 특성에 기반한 어노테이션 앵커 선택 및 추천, 그리고 고아처리 등의 기능을 제공하여 문서변경시 위치재생성에 대한 보다 심도 있는 처리가 가능하다.

한편 본 연구는 웹 뿐만 아니라 XML 및 SGML 과 같은 구조문서를 사용하는 어노테이션 시스템에 적용 가능하다. 따라서 원본문서의 갱신이 빈번하게 발생하는 웹 어노테이션 환경 및 사이버 클래스, e-Learning 등에 효과적으로 적용 가능하다. 향후 본 연구진은 XML 문서를 기반으로 하며 어노테이션 기능을 포함하는 전산 관련 사이버 클래스 콘텐츠를 구축하고 실제 강의 환경에서 본 연구결과에 대한 사용성을 평가할 예정이다.

참 고 문 헌

- [1] OVSIANNIKOV, I. A., ARBIB, M. A. & MCH-EILL, T. H., "Annotation Technology," International Journal of Human-Computer Studies, 50 (4), 329-362, 1999.
- [2] SOHN, W. S., et al., "Standardization of eBook documents in the Korean Industry," Computer Standards & Interfaces, 24(1), 45-60, 2002.
- [3] BRUSH, A.J., DAVID, B., ANOOP, G. & CADIZ, J.J., "Robust Annotation Positioning in Digital Documents," Proceedings of CHI'01, Seattle, March 31, ACM Press, NY, 285-292, 2001.
- [4] CADIZ, J., GUPTA, A., & GRUDIN, J., "Using Web Annotations for Asynchronous Collaboration Around Documents," Proceedings of CSCW '00, Philadelphia, ACM Press, NY, 309-318, 2000.
- [5] E-Quill, E-Quill Corp. Visual Teaming, <http://www.e-quill.com>, 2001.
- [6] GRØNBÆK, K., SLOTH, L. & P. ØRBÆK, "Web-vise: Browser and Proxy Support for Open Hypermedia Structuring Mechanisms on the WWW," Proceedings of the Eighth World Wide Web Conference, Toronto, Canada, 1999.
- [7] ROSCHEISEN, M., MOGENSEN, C. & WINOGRAD, T., "Shared Web Annotations as a Platform for third-party Value-added Information Providers: architecture, protocols, and usage examples," Technical Report, CSDTR/DLTR, Stanford University, 1995.
- [8] BERNHEIM, A.J., BRUSH & BARGERON, D., "Robustly Anchoring Annotations Using Keywords," Technical Report, MSR-TR-2001-107, Microsoft Research, 2001.
- [9] PHELPS, T. A. & WILENSKY, R., "Robust Intra-document Locations," Proceedings of the 9th WWW Conference, Amsterdam, 2000.
- [10] DAVIS, H. C., "Referential Integrity of Links in Open Hypermedia Systems," Proceedings of ACM Hypertext '98, Pittsburgh, ACM Press, NY, 207-216, 2000.
- [11] PHELPS, T. A. & WILENSKY, R., "Multivalent Documents," Communications of the ACM, 43 (6), 83-90, 2000.
- [12] YEE, K.-P., The CritLink Mediator. <http://crit.org/critlink.html>, 1997.
- [13] KAHAN, J., KOIVUNEN, M. R., HOMMEAU, E. & SWICK, R. (2001). Annotea: An Open RDF Infrastructure for Shared Web Annotations. Proceedings of the WWW10 International Conference. Hong Kong.
- [14] CHANG, G. J. S., PATEL, G., RELIHAN, L. & WANG, J. T. J., "A Graphical Environment for Change Detection in Structured Documents," Proceedings of Twenty-First Annual Int'l Computer Software and Applications Conference (COMPSAC'97), Los Alamitos, CA, 536-541, 1997.
- [15] 이경호, 조성배, 최윤철, "경로매칭 알고리즘을 이용한 구조화된 문서의 변화탐지", 정보과학회논문지, 28 (4) 2001.



손 원 성

1998년 동국대학교 컴퓨터공학(학사)
2000년 동국대학교 컴퓨터공학(석사)
2004년 연세대학교 컴퓨터과학과(박사)
2004년~현재 Carnegie Mellon University, Computational Design Lab, Associate Researcher. 관심분야는 웹 문서처리, Annotation, 스마트 홈, 유비쿼터스, XML



김 재 경

2000년 단국대학교 전산통계(학사). 2002년 연세대학교 컴퓨터과학(석사). 2003년 현재 연세대학교 컴퓨터과학과 박사과정
관심분야는 Animated Character, 웹 Annotation 생성 및 응용



고 명 철

1995년 제주대학교 정보공학과(학사)
 1997년 연세대학교 컴퓨터과학과(석사)
 2003년 연세대학교 컴퓨터과학과(박사)
 2004년~현재 건국대학교 컴퓨터응용과
 학부 조교수. 2003년~2004년 (주)GIS뱅크
 연구개발실장. 관심분야는 3D 가상환

경, 3D 입력시스템, Mobile/Ubiquitous 그래픽스, GIS



임 순 범

1982년 서울대학교 계산통계학과(학사)
 1983년 한국과학기술원 전산학과(석사)
 1992년 한국과학기술원 전산학과(박사)
 1989년~1992년 (주)휴먼컴퓨터 창업/연
 구소장. 1992년~1997년 (주)삼보컴퓨터
 프린터개발부 부장. 1997년~2001년 건

국대학교 컴퓨터과학과 교수. 2001년~현재 숙명여자대학교
 멀티미디어과학과 교수. 관심분야는 컴퓨터 그래픽스, 모바
 일 멀티미디어 응용, 전자출판(폰트, XML, 전자책,
 e-Learning)



최 윤 철

1973년 서울대학교(학사). 1975년 Univ.
 of Pittsburgh(석사). 1976년 Univ. of
 California, Berkeley(석사). 1979년 Univ.
 of California, Berkeley(박사). 1979년~
 1982년 Lockheed 사 및 Rockwell
 Internatonal 사 연구원. 1990년~1991년

University of Massachusetts 교환교수. 2002년~2003년
 일본 게이오대학 교환교수. 1984년~현재 연세대학교 컴퓨
 터과학과 교수. 관심분야는 멀티미디어와 웹, 멀티미디어 문
 서처리, 3D 사용자 인터페이스, 아바타 인터페이스, 컴퓨터
 그래픽스, eLearning 및 Cyber Class