

펜 기반 웹 문서 교정을 위한 모호성 문제 해결에 관한 연구

손원성

경인교육대학교 컴퓨터교육과

요약

전자펜을 이용한 문서교정 시스템에서 정확한 교정결과를 보장하기 위해서는 문서 교정자가 드로잉한 교정부호와 문서내용간의 영역 모호성(ambiguity)을 해결하여야 한다. 한편 교정의 대상이 되는 전자문서가 HTML/XML과 같은 경우 교정된 문서구조가 반드시 기 정의된 DTD를 위배하지 않아야 한다. 본 논문에서는 펜 기반의 교정시스템에서 교정부호(마킹)와 대상문서간의 모호성 문제를 최소화하기 위한 기법을 제안한다. 제안 인터페이스에서는 모호성 문제를 최소화하기 위하여 교정부호와 문서간의 컨텍스트(Context)를 반영하였으며 동시에 대상문서의 문서 구조를 유지하기 위한 방법을 제공한다. 그 결과 본 논문에서 제안한 교정 인터페이스는 기존 교정시스템에 비하여 보다 정확한 영역정보를 포함할 수 있으며, 교정부호 입력에 따른 구조문서 변경시에도 원본문서의 DTD에 따르는 문서구조를 유지할 수 있다.

A Study on Ambiguity Resolving for Pen-based Proofreading of Web Documents

Won-Sung Sohn

Gyeongin National University of Education, Dept, of Computer Education

ABSTRACT

To produce accurate editing results, the ambiguity of editing scopes related to marked correction signs should be solved. Proofreading the web document modifies the document structures, and the modified structures should be robustly valid for the defined DTD. This paper presents a pen-based proof-reading interface in the XML document. In the proposed interface, correction signs are free-drawn, and the editing scopes are recognized and revised based on the contexts of the document to minimize the ambiguity of the editing scopes. The proposed interface provides both implicit and explicit modification methods for document structures. As a result, the editing scopes processed in the proposed interface are more accurate, and the document structures are maintained valid for DTD after the editing.

Keywords: Pen-based Proofreading, XML, Annotation, Ambiguity, Context

1. 서론

전자펜을 이용한 마킹기능은 태블릿 PC 및 PDA에서 구동되는 어플리케이션에서 주로 주석, 하이라이팅, 그리고 도형그리기와 같은 부가 기능을 담당하고 있다. 따라서 휴대용 기기를 이용하는 전자책[14] 및 전자교과서 등에서는 전자펜 기반의 마킹기능을 필수로 제공하고 있다[4]. 아래의 (그림 1)은 싱가포르의 휴대용 전자교과서 프로젝트인 EduPDA[16]의 사용화면이며, 이러한 환경에서는 다양한 전자펜의 활용이 가능하다[2, 4, 6, 7, 8, 9].

한편 컴퓨터 상에서 교정부호를 마킹(Marking)하여 전자문서를 수정하는 기법은 앞서 언급한 전자교과서 및 전자책과 같은 환경에서 효과적으로 사용될 수 있다. 최근의 교정시스템은 종이 문서가 아닌 전자문서에 교정부호를 입력하고 그 결과를 즉시 확인할 수 있는 형태의 문서편집 과정을 수행한다[3, 6, 7, 10, 11, 12].



(그림 1) 휴대용 기기에서의 전자교과서 사용 예[16]

이러한 교정시스템은 간결한 교정 절차, 자유로운 교정부호 입력 및 편집, 다중사용자간의 협업, 교정 결과에 대한 재활용 등과 같은 기능[1, 8]을 제공하기 때문에 유비쿼터스 기반의 수업에 유용하게 사용될 수 있다. 특히 전자펜을 이용하여 전자문서 환경에서도 전통적인 종이환경에서 사용하는 마크업 모델[11]을 적용한 교정 인터페이스는 사용성(usability) 측면에서 다양한 사용자 그룹에 적용 가능하다.[11],[12].

펜 기반 교정 모델을 채택한 기존 시스템으로는 MATE[12], PenEdit[12], 그리고 Amaya+PEN[2] 등을 살펴볼 수 있다. MATE에서는 펜을 이용하여 체크처를 드로잉하고 입력된 체크처는 다음 (그림 2)와 같이 별도의 인식과정을 거쳐 적절한 편집동작

으로 변환된다. 따라서 사용자는 편집자의 의도 및 편집결과를 직관적으로 파악할 수가 있다.

ANNOTATION VIEW	EDIT VIEW
<p>This is a sample document with several annotations marked on it. Note that some annotations correspond to specific editing commands.</p> <p>Whereas others are more general comments.</p>	<p>This is a sample document with several annotations marked on it. Note that some annotations correspond to editing commands: Whereas others are more general comments.</p>

(그림 2) MATE 시스템의 화면 예

PenEdit는 다음 (그림 3)과 같이 스크린을 통한 펜기반 입력 및 제스처 인식 기능을 제공하며, 편집자는 자신의 의도를 교정부호와는 별도의 어노테이션 형태로 표현할 수 있는 인터페이스를 제공하여 저자와 편집자간의 용이한 커뮤니케이션을 가능케 한다.

Much of the background knowledge & and guidelines for structure kiosk plan design comes from the field of exhibit plan design (e.g., Klein, 1988; Konikow, [Au: Spell 1] 1984; Miles & Alt, 1988). [Au: Date 1] Indeed, a major principle of exhibit plan design is to create displays that attract attention & and invite participation. However, the many technology components & and the nature of the interaction possible with interactive systems poses many new challenges even for experienced exhibit designers[FN 2].

End Page 1

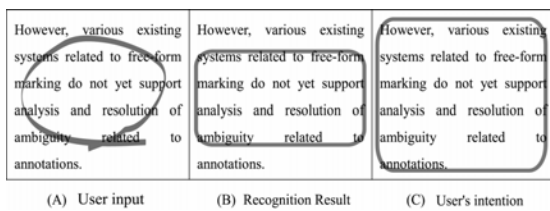
(그림 3) PenEdit 시스템의 예

한편 Aamaya+PEN[3]에서는 앞서 살펴본 교정 인터페이스와는 달리 HTML 문서를 교정 타겟으로 간주한다. Aamaya+PEN은 다음 (그림 4)와 같이 W3C에서 제공하는 HTML 편집기인 Amaya[2]를 기반으로 펜기반의 웹문서 편집이 가능하다. 펜기반의 제스처 인식에서는 교정 환경의 특성을 고려한 기하학적 알고리즘을 사용하였으며 텍스트 정보와 부호간의 관계를 고려한 교정 부호 인식 기법을 적용하였다.



(그림 4) 펜기반 온라인 교정시스템 (Amaya+Pen)

앞서 살펴본 시스템들은 다양한 장점들을 제공하고 있으나 기술적으로 해결되어야 할 문제점들을 포함하고 있다. 전자문서 환경에서 펜 입력기반의 교정 마킹 기능을 제공하는 시스템에서는 사용자가 입력한 자유형(free-form) 마킹과 대상문서의 영역을 판별하는 과정에서 다음 (그림 5)와 같은 영역 인식에 대한 모호성(ambiguity)[5, 10] 문제가 필수적으로 발생하며, 이러한 모호성은 사용자의 의도와 전혀 다른 교정결과를 생성하게 된다.



(그림 5) 영역 모호성 발생의 예

최근의 교정 시스템은 일반 전자문서뿐만 아니라 웹 문서(HTML/XML)를 교정의 대상으로 간주된다[2]. 이 경우 교정 작업은 마크업문서 구조에 대한 변화를 수반하며, 특히 XML과 같은 구조문서 환경에서는 교정에 따라 변경된 문서 구조 또한 정의된 DTD(document type definition)에 따라야 한다. 그러나 현재 대부분의 웹 기반 교정 시스템에서는 교정작업에 따른 문서 구조 변경을 고려하지 않거나 HTML 문서만을 고려한다[2, 11].

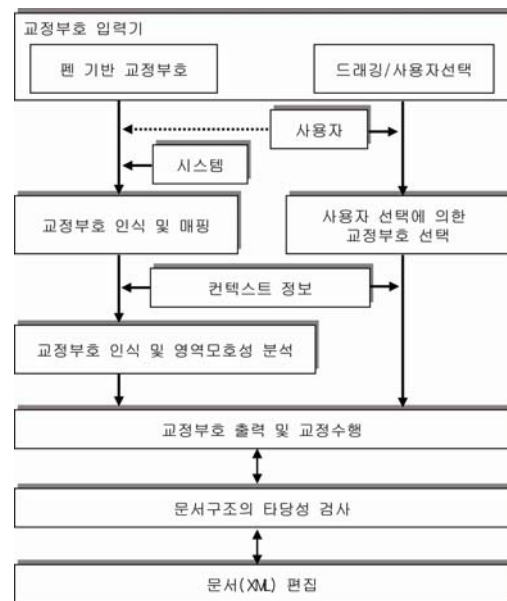
따라서 본 논문에서는 구조문서(XML)를 기반으로 하며 모호성 문제를 해결하기 위한 교정 인터페이스를 설계 및 구현하였다. 본 인터페이스에서는 영역 모호성을 최소화하기 위한 컨텍스트 기반 영역 인식 및 보정 기법을 제공하며 입력된 교정부호를 본 논문에서 정의한 Context-based Proofreading Markup Language(CPML)에 기반한

마크업 언어로 표기한다. 또한 교정부호의 입력에 따른 문서 내용 및 구조의 변화가 발생할 경우 기존 HTML 기반의 교정 시스템과는 달리 시스템 내부에서 자동으로 문서 구조를 유지하기 위한 묵시적인 구조변경 기법과 사용자와의 인터랙션을 통하여 문서 구조를 유지하는 명시적인 문서 구조변경 방법을 제공한다.

그 결과 제안 인터페이스의 사용자들은 보다 정확한 교정 결과를 제공받을 수 있으며 교정 부호 입력에 따른 구조문서 변경에도 원본문서의 DTD에 위배되지 않는 문서 구조를 유지할 수 있다.

2. 컨텍스트 기반 교정 인터페이스

본 장에서는 제안 교정 인터페이스의 전체 구조 및 주요 기능에 대하여 살펴본다. 본 논문의 XML 기반 교정 시스템은 Windows Tablet OS 및 Windows CE 플랫폼을 기반으로 한다. 또한 교정문서는 모두 XML 기반의 전자책 표준[14]을 사용하였다. 제안 시스템의 구성도는 다음 (그림 6)과 같다.



(그림 6) 제안 교정 인터페이스의 구성도

본 인터페이스의 처리단계는 다음과 같다. 사용자로부터 입력된 교정부호는 시스템의 사용자 인터페이스를 통하여 픽셀정보로 처리되며 그 정보를 인식모듈로 전달한다. 인식모듈에서는 먼저 사용자 인터페이스로부터 전달 받은 물리적 교정부호 정보를 인식하여 적절한 교정부호 타입을 선택한다. 또한

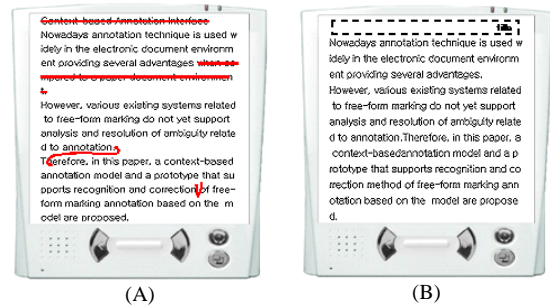
문서처리 모듈에서 추출한 컨텍스트 정보(그림 9 참조)와 3장의 제안 기법을 이용하여 교정 영역에 대한 최종 영역을 판단한다. 인식모듈에서 처리한 최종 교정 타입 및 영역 정보는 XML 형태로 표현되며 이것을 기반으로 문서의 내용을 교정하게 된다. 교정된 결과에 따라 문서구조가 변경되었을 경우 해당 DTD에 타당한지를 검사하도록 한다.

한편 본 논문에서는 XML 형태로 교정부호를 정의하고자 하며 이를 위하여 XML 기반의 Context-based Proofreading Markup Language (CPML)를 정의하였다. CPML에서는 (그림 7)과 같이 총 12개의 교정부호 타입을 정의하였으며 이는 교정 부호 표준인 Chicago Style Proofreader's Marks[15]를 전자문서 환경에 맞게 간소화한 것이다. CPML의 교정 타입은 delete, insert, transpose, closeUp, splitPara, joinPara, moveLeft, moveRight, createLink, deleteLink, replace, note 엘리먼트를 포함한다.

Correction	Notation	Correction	Notation
Delete		CloseUp)
Insert	<	MoveLeft	↶
Transpose	↷	MoveRight	↷
SplitPara	↵	CreateLink	⊕
JoinPara	↵	DeleteLink	⊖
Replace	≡	Note	≈

(그림 7) CPML의 교정부호 (DeleteLink 부호의 경우 링크가 생성된 이후에만 적용된다.)

실제 구현된 본 연구의 교정 시스템은 다음 (그림 8) (A)와 같이 편의성을 고려한 자유형 드로잉방식과 정확성을 고려한 드래깅(dragging) 방식을 포함하는 2가지 교정부호 입력 인터페이스를 제공한다. (그림 8) (A)와 같이 드로잉 방식으로 입력된 교정마크의 타입은 제스처 인식 기법에 의하여 결정된다. 또한 입력된 교정마크의 최종 영역은 모호성 해결을 위한 컨텍스트 기반 인식 및 보정기법에 근거하여 자동으로 보정되어 화면에 출력된다. 출력된 교정 부호는 선택 옵션에 따라 곧바로 문서의 편집에 사용되거나 입력된 부호를 화면에 출력된 채로 남겨두었다가 메뉴에 의한 사용자의 실행 명령으로 일괄적 혹은 선택적으로 문서를 편집하게 된다.



(그림 8) 제안 시스템의 교정부호 입력 및 편집결과 화면

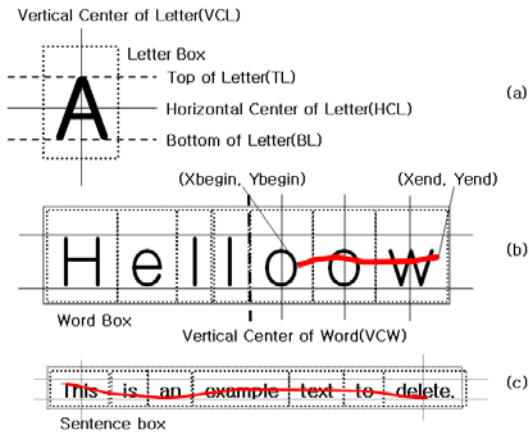
(그림 8) (B)에서는 실제 교정부호 입력 및 교정 결과를 나타내고 있다. (그림 8) (A)에서는 총 4개의 교정부호를 입력한 예를 보여주고 있으며 (그림 8) (B)에서는 각각의 교정 연산이 적용된 결과를 보여주고 있다. 첫번째 삭제 교정부호는 DTD에서 반드시 발생하여야 하는 구조를 포함하기 때문에 삭제 후에도 그 영역을 별도로 출력되었다. 나머지 교정 부호들은 시스템 내부의 인식 모듈을 통하여 영역 보정되어 (그림 8) (B)와 같이 편집된다.

3. 영역 모호성 해결을 위한 교정기법

본 장에서는 교정 부호와 편집 대상 텍스트간에 편집 영역 판별시 발생하는 모호성을 최소화하기 위한 인식 기법에 대하여 설명한다. 제안 기법은 IF-THEN 문법으로 표현되는 규칙 모델을 통하여 처리된다. 본 논문의 규칙 시스템은 총 12개의 교정부호 타입과 편집 텍스트 영역 인식을 위한 조건으로 이루어진 90여개의 규칙으로 정의되어 있다. 본 장에서는 이러한 규칙들 중 텍스트 영역 및 교정부호 인식과정에 대한 설명을 위해 DELETE 연산의 경우를 예를 들도록 한다.

3.1 텍스트 영역 인식

먼저 입력된 펜기반 제스처는 기존 인식 알고리즘[13]을 이용하여 타입이 결정된다. 예를 들어 본 연구에서는 텍스트의 중간을 가로로 긋는 선을 DELETE라고 정의 하였다. 따라서 (그림 9)의 (b), (c)와 같은 경우 교정 부호의 타입은 DELETE가 된다.



(그림 9) 텍스트 영역 기준선 및 DELETE 부호 입력 예

다음으로 교정 부호가 적용될 텍스트 영역을 인식하기 위해서 부호와 텍스트에 몇가지 기준선 및 점들을 정의하였다. 먼저 모델 정의 3과 같이 텍스트는 LETTER, WORD, SENTENCE, PARAGRAPH 단위로 구성되며 각 단위는 BOX로 구분한다. 이 단위들은 (그림 9)와 같이 부호의 시작점(Xbegin, Ybegin)과 끝점(Xend, Yend), LETTER의 중앙수직선(VCL)와 중앙수평선(HCL), LETTER의 상단선(TL)과 하단(BL), 단어의 중앙수직선(VCW)등으로 기준선에 의해 영역이 세분화된다.

(그림 9)의 (b)와 같은 경우 제안규칙에 의해 LETTER 단위의 DELETE 연산으로 판단이 되며 유효한 삭제 대상 텍스트 영역은 마지막의 'ow'가 된다. 먼저 입력된 교정 부호의 타입이 DELETE로 인식되고 Ybegin과 Yend이 TL과 BL 사이에 포함될 경우 해당 텍스트에 대한 삭제연산이다.

삭제 대상 텍스트 영역을 판별하기 위해 먼저 Xbegin과 Xend의 위치가 VCL의 좌우측에 존재하는가의 여부에 따라 그 위치의 LETTER를 편집 대상 영역으로 넣을것인지 결정한다. 위 예에서 5번째 LETTER인 'o'는 교정 부호의 Xbegin이 VCL 좌측에 존재하므로 편집 대상 텍스트 영역은 6번째 'o'가 된다. 마찬가지로 Xend에서의 편집 대상 텍스트는 'w'가 된다.

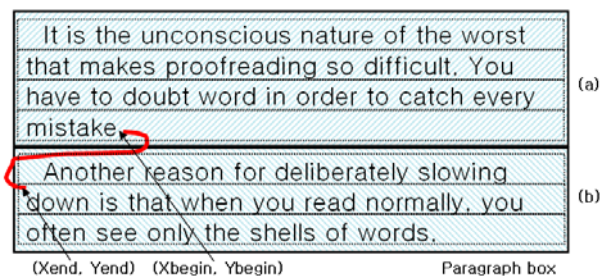
다음으로 Xbegin과 Xend 사이의 LETTER의 개수가 WORD를 구성하고 있는 전체 LETTER의 개수보다 작아야 한다. 왜냐하면 WORD 영역 내에 포함되는 교정 부호는 LETTER 단위의 편집 명령이며 그렇지 않은 것은 WORD 단위 이상의 편집 명령이기 때문이다. 이 예에서는 이 조건을 만족하

여 LETTER 단위의 DELETE 명령을 실행하게 되며 만약 그렇지 않다면 WORD 단위의 연산이 되어 다른 규칙으로 처리가 된다.

(그림 9)의 (c)는 LETTER 단위가 아닌 SENTENCE 단위의 편집임을 직관적으로 알 수 있다. 따라서 (b)의 경우와는 다른 인식 규칙을 적용할 필요가 있다. 이 예에서 마지막 WORD인 'delete'에 DELETE 부호가 완전히 입력되지 않았음을 알 수 있다. 사람은 인지적으로 전후 컨텍스트를 파악하여 교정자가 한 문장을 삭제하라는 것임을 쉽게 알 수 있으나 소프트웨어로 이를 처리할 때는 교정 부호와 텍스트간의 영역 매칭이 명확하지 않아 입력시 모호성이 발생하는 것이다.

본 연구에서는 이러한 경우 모호성을 제거하기 위하여 해당규칙을 적용하여 한 문장을 삭제하도록 정의하였다. 또한 본 연구에서는 동일 타입의 부호 외에도 각 교정 부호의 타입별 특성에 따라 인식 규칙을 달리 하는 것이 바람직하기 때문에 이를 위한 규칙 모델을 정의하였다. 예를 들어 SPLIT/JOIN_PARAGRAPH와 같은 부호는 주로 문단 단위로 연산이 이루어지기 때문에 이에 맞는 문단 단위의 텍스트 영역 인식 규칙을 적용하였다.

(그림 10)은 문단 단위 편집의 예로써, 입력된 교정 부호는 JOIN_PARAGRAPH인데 이것은 (a) 문단과 (b) 문단을 합치는 역할을 한다. 이 경우 교정 부호의 시작위치는 (a) 문단의 마지막 문장의 끝에 위치하고, 부호의 종료 지점은 (b) 문단의 두 번째 문장의 시작 부분에 위치한다. 이 경우 입력의 모호성으로 (a) 문단을 (b) 문단의 어느 문장 앞에 병합하여야 하는지가 불명확하다.



(그림 10) 문단 단위 교정의 예

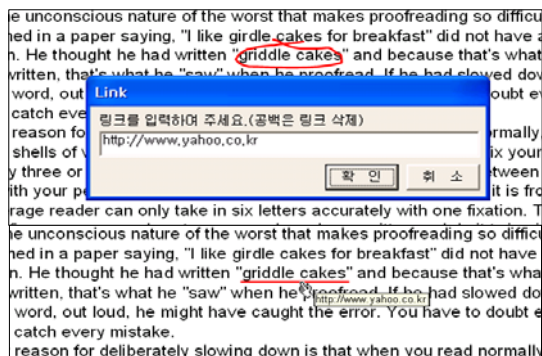
그러나 일반적으로 문단의 병합시에는 인접한 상하 문단을 합치게 되며 문단 병합 교정 부호를 사용하여 한 문단을 다른 문단의 중간에 삽입하는 경우는 극히 드물다. 따라서 (그림 10)와 같은 예에서는 사용자가 (a) 문단과 (b) 문단을 병합시키는 목

적으로 교정 부호를 입력하였다고 판단해야 한다.

이와 같은 사용자의 의도를 파악하기 위해 본 연구에서는 JOIN_PARAGRAPH와 같은 문단 단위 편집 명령에 대한 규칙을 적용한다. (그림 10)과 같은 예제에서는 해당규칙에 의하여 (a)와 (b) 문단을 병합하는 명령을 실행한다.

다음은 온라인 구조 문서 환경을 위해 종이 환경의 기존 교정 부호에 새로이 추가된 링크와 구조정보의 추가 및 삭제 부호에 대하여 알아본다. 먼저 링크의 추가 명령시 편집 영역 인식은 앞서 설명한 DELETE 명령의 경우와 비슷하게 이루어진다. 즉 입력 영역에 따라 글자, 단어 및 문장단위로 나누어져 규칙이 적용된다. 편집 영역이 결정되면 (그림 11)과 같이 링크를 입력할 수 있는 다이얼로그에 임의의 웹 형식의 링크를 입력하면 된다.

링크의 삭제 명령시에는 영역 인식 방법이 조금 틀려진다. 왜냐하면 사용자가 링크의 삭제 부호를 입력할 때는 이미 존재하는 링크 영역을 대상으로 하게 된다. 따라서 링크의 삭제 부호가 입력된 곳에 존재하는 텍스트 링크 영역이 편집 영역이 된다.



(그림 11) 링크의 추가

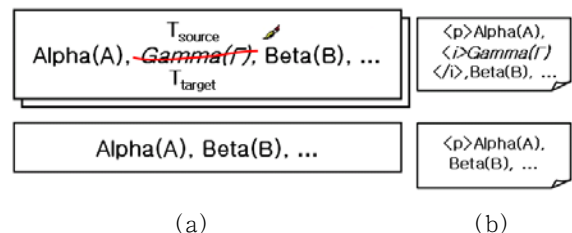
3.2 구조 정보 변경 규칙

HTML이나 XML 형태의 웹 문서의 특징은 일반 텍스트 문서와 달리 구조를 갖는 것이다. 따라서 웹 문서를 편집 할 때는 텍스트의 삭제나 추가등과 같은 연산으로 생기는 구조의 변경을 고려해야 한다. 그러나 펜 기반 교정 시스템에서는 이미 작성된 문서에 대하여 오류가 있는 곳을 펜을 통한 교정 부호로 수정하는 것을 목적으로 하기 때문에 세밀한 구조 정보 편집기능을 지원하기에는 어려움이 따른다.

본 연구에서는 펜 기반 교정 시스템에서 텍스트 편집 중 발생할 수 있는 구조정보의 변경만을 고려하여 이에 대한 처리 규칙을 정의하였다. 즉, 제안

된 구조 정보 편집 규칙모델에서는 입력된 교정 부호와 구조정보와의 관계를 파악하여 구조 정보의 변경이 예측될 시에는 변경 가능한 모든 경우의 수를 산출한다. 즉 제안된 규칙에 따라 구조 문서의 DTD에 위배되지 않는 범위 내에서 규칙 모델에 따라 적합한 구조 변경을 묵시적으로 시도한다. 그러나 경우에 따라서 이것은 사용자가 원하지 않는 결과가 될 수 있으므로 가능한 구조 변경 결과들을 추천하여 사용자가 명시적으로 선택할 수 있는 기법을 제안한다.

먼저 묵시적인 구조 변경의 경우는 사용자의 편집 명령으로 인한 문서의 구조 정보 변경의 결과를 정확히 예측 가능할 때 적용될 수 있다. 즉 편집 명령으로 인하여 나올 수 있는 문서 구조 변경의 경우의 수가 한가지 일 때를 말한다. 예를 들어 교정자가 인라인 태그로 묶여 있는 단어를 포함하는 한 단어를 (그림 12) (a)와 같이 DELETE 명령으로 삭제하였을 때 그 문장이 포함하고 있던 인라인 태그의 구조정보는 (b)와 같이 단어와 함께 제거된다. 단, 구조 정보 편집 결과가 DTD에서 허용하지 않는다면 텍스트 노드 없이 빈 태그만을 남겨놓는다.



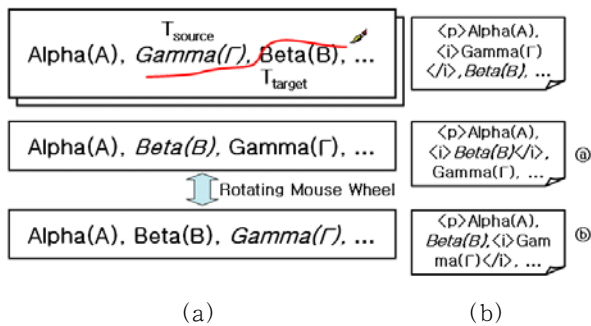
(그림 12) DELETE 부호입력시 구조 정보의 변경

교정 부호의 특성이나 구조정보의 위치에 따라 편집 명령으로 변경된 구조정보의 형태가 다수가 될 수 있다. 이러한 경우에 시스템에서 묵시적으로 판단하여 가장 적절한 결과를 교정자에게 출력하여 줄 수 있지만 경우에 따라서는 교정자가 원하지 않는 결과가 나올 수 있다.

따라서 가능한 결과들을 교정자가 화면으로 확인하면서 원하는 결과를 선택할 수 있도록 하는 것이 바람직하다. 이를 위해 가능한 구조정보 변경 결과를 편집 작업시 일어나는 모든 경우에 따라 정의해 놓아야 하며 임의의 편집 명령으로 특정 결과들이 예측될 때는 이를 교정자에게 제공하여 준다. 본 연구에서는 교정 부호 입력에 따른 구조정보 변경 결과들을 복수의 규칙에 따라 정의하였으며 교정자는 마우스 휠을 회전시켜 시스템에서 추천하는 결과들

을 미리보고 원하는 결과를 선택할 수 있다

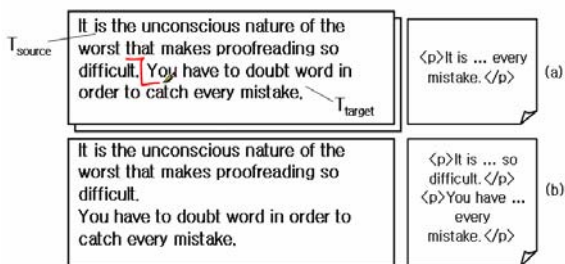
예를 들어 (그림 13)의 (a)와 같은 경우 Tsource와 Ttarget를 교환하는 편집 부호를 입력하는 예인데 Ttarget이 <i> 구조 정보를 가지기 때문에 해당 규칙에 따라서 (b)의 ㉠ 또는 ㉡와 같은 결과를 예측할 수 있다. 교정자는 1차적으로 교정결과가 적용된 문장 위에서 별도의 메뉴를 생성하거나 혹은 마우스 휠을 회전하여 두개의 결과를 화면에서 전환하여 확인하며 원하는 결과를 선택한다.



(그림 13) Transpose 명령에 의한 구조정보 변경

다음은 단락 단위 편집의 예를 살펴보기로 한다. 앞서 JOIN_PARAGRAPH에 대하여 설명했으므로 SPLIT_PARAGRAPH의 경우를 보자.

SPLIT_PARAGRAPH는 단락을 두 개로 나누는 것으로 이에 해당하는 교정부호의 입력 후 화면상으로 단락이 분리되는 결과이외에 내부 문서의 구조 정보 또한 변경이 될 수 있다. 즉 (그림 14)의 (a)와 같이 교정 부호를 입력하였을 때 화면상으로 (b)와 같은 결과를 출력하게 되는 것은 기존 시스템과 동일하다.



(그림 14) 문단 단위 편집시 구조 정보의 변경

여기서 문단 나눔 편집으로 변경될 수 있는 사항을 고려하여야 한다. 본 연구에서 사용된 문서에서는 단락이 <p> 태그로 묶여져 있는데 이 단락이 나누어지게 될 경우 두 개의 <p> 태그를 사용할

것인가를 결정해야 한다. 일반적으로 단락들은 각각 <p> 태그로 표현하기 때문에 하나의 <p> 태그를 두 개로 분리한다. 물론 이와 같은 구조 정보의 변경은 DTD에서 허용되는 범위에서 일어나야 한다.

지금까지 펜기반 편집에 의해 일어날 수 있는 구조 정보 변경의 예와 이를 처리하기 위해 본 연구에서 제안하는 규칙과 기법들에 대하여 알아보았다. 이러한 규칙과 기법들은 웹 문서의 세밀한 구조 정보 편집을 위하여 정의된 것은 아니며 교정 작업도중 발생하는 문서의 구조 변경을 최대한 지원하기 위한 것이다.

따라서 교정자가 문서의 구조에 대해 최대한 신경을 쓰지 않도록 규칙을 정의하여 내부적으로 문서 구조의 변경이 이루어지도록 하였다. 그러나 경우에 따라 시스템에서 판별하기 어려운 경우에는 최소한의 사용자의 피드백을 받아 사용자가 원하는 결과를 적용할 수 있도록 하였다.

6. 실험결과

본 논문에서는 제안 기법에 대한 성능을 평가하기 위하여 경험적(empirical) 사용자 평가를 수행하였다. 본 실험에서는 사용자들이 생성한 교정부호 영역에 대한 인식 정확도를 측정하기 위하여, 제안 기법과 기존 Amaya+PEN[2]에서 사용하는 기법을 적용한 교정 프로토타입을 이용하여 사용자 평가를 수행하였다. 펜 기반의 교정부호 입력에 대한 정확도 측정은 정량적인 측정이 어렵기 때문에, 2가지 프로토타입에서 처리된 교정 결과에 대한 정확도를 사용자들이 측정하도록 하였다.

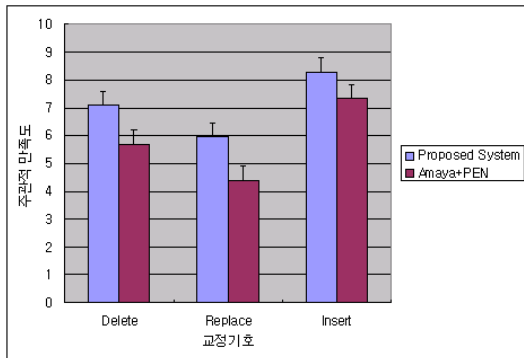
6.1 실험 1

본 실험에서는 본 논문에서 제안한 기법과 기존 시스템에서의 교정 영역에 대한 정확도를 측정하기 위하여 자신이 의도한 교정 영역과, 시스템에서 처리한 영역과의 정확도를 개인으로부터 입력받도록 한다. 이를 위하여 본 실험에서는 먼저 30명의 사용자들에게 각각 삭제(10개), 대체(10개), 삽입(10개)을 위한 교정부호를 드로잉 하도록 한다. 프로토타입 내부에서는 입력된 교정 마킹에 대하여 제안 기법 및 기존기법[3]을 적용한 각각의 영역 판별 결과를 추출한다. 모든 실험이 끝난 후에는 시스템에서 추출한 판별 영역을 사용자에게 보여주고, 자신이 의도한 영역과 얼마나 차이가 나는지를 알아보기 위하여 1(lowest accuracy) 부터 10(highest accuracy)

까지의 스케일(scale)로 구성된 설문지를 이용한 설문 조사를 실시하였다. 본 실험에는 20명의 참여자가 참여하였다. 실험을 위해서는 반복있는 분산분석(ANOVA)를 사용하였다.

다음 (그림 15)는 입력된 교정영역에 대한 사용자들의 주관적 정확도를 나타내고 있다. (그림 15)의 하면 제안 시스템에서의 교정 영역결과에 대하여 대부분의 사용자들은 보다 정확하다고 평가하였으며, 또한 제안 기법과 기존 기법에 따른 영역정확도 간에는 통계적으로 유의한 차이가 있음을 알 수 있었다. ($F(1,114) = 36, P < 0.05$). 또한 각 교정 부호에 대해서도 통계적으로 유의한 차이가 있음을 알 수 있었다. (삭제, $F(1,38) = 12.1, P < 0.05$, 대체, $F(1,38) = 14.7, P < 0.05$, 삽입, $F(1,38) = 9.33, P < 0.05$).

대부분의 사용자들은 제안 시스템에서 처리된 교정영역을 기본 기법보다 정확하다고 판단하였다. 특히 삭제와 대체 교정 연산의 경우 기존 시스템에서 처리된 영역보다 더 높은 정확도를 보였으며, 이는 복수의 단락에서 처리되는 경우가 많기 때문이다. 즉 제안 기법은 복수의 단락에서 수행되는 교정연산 기호에 보다 효과적으로 적용됨을 알 수 있었다.



(그림 15) 실험1에서 교정 영역에 대한 주관적 정확도

6.2 실험 2

본 실험에서는 제안 기법과 사용자들의 인지적 오버헤드(cognitive overhead)간의 영향을 분석하기 위하여, 제안 기법 및 기존 기법에 따른 사용자들의 삭제 횟수 및 총 작성 시간을 분석하였다. 이를 위하여 미리 정해진 총 30개의 교정 부호를 입력하도록 하고, 이에 대한 총 삭제 횟수 및 작성 시간을 추출하였다. 이에 대한 결과는 다음과 같다.

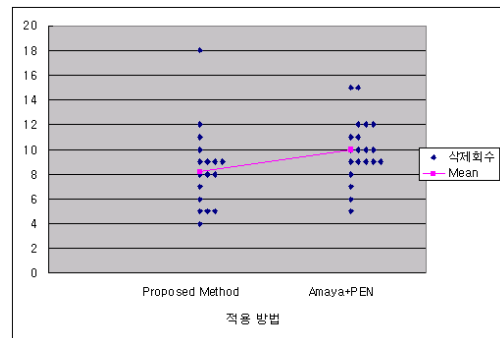
먼저 제안 기법과 기존 시스템과의 인지적 오버헤드를 비교하기 위하여 삭제 횟수를 분석하였다.

그 결과 다음 (그림 16)과 같이 제안 기법이 기존 기법보다 평균적으로 본 논문에서 제안된 기법 삭제 횟수에 있어 통계적으로 유의한 차이가 보이는 것을 알 수 있었다. ($F(1,19) = 8.44, P < 0.05$)

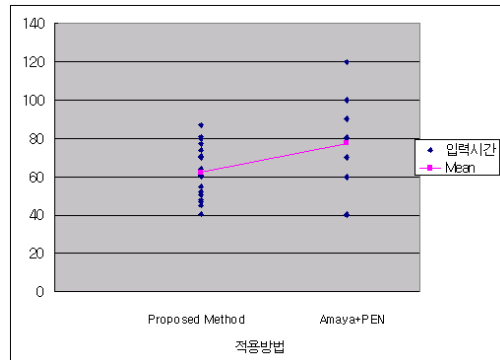
한편 각 시스템에 따른 총 입력 시간은 다음 (그림 17)과 같이 평균적으로는 적은 시간을 포함하고 있으나, 분산분석 결과 입력 기법에 따른 각 시스템 간의 총 입력시간들은 통계적으로 유의한 차이가 보이지 않음을 알 수 있다. ($F(1,19) = 5.82, P > 0.05$)

제안 기법에 따른 인지적 오버헤드 중, 삭제 횟수에 대해서는 제안 기법이 효과적임을 알 수 있었다. 이는 제안 기법이 기존 시스템보다 정확한 교정 영역을 포함하기 때문인 동시에 이에 대한 만족도가 높기 때문이다.

그러나 총 수행시간에 대해서는 유의점을 찾지 못하였는데, 그 이유는 사용자들이 교정 부호에 따라 서로 다른 드로잉 형태를 갖기 때문이었다. 즉 복수라인에 걸쳐있는 교정 부호와 같은 경우는 제안 시스템 및 기존 시스템과는 무관하게 모두 신중히 드로잉하는 경향을 보였기 때문이다. 따라서 이 부분에 대해서는 보다 밀도 있는 연구가 필요할 것으로 예상된다.



(그림 16) 각 시스템에 따른 삭제 횟수의 비교



(그림 17) 각 시스템에 따른 총 수행시간의 비교

7. 결론

본 연구에서는 전통적 교정 모델을 기반으로 하는 전자펜 기반의 교정 기법 및 시스템을 제시하였다. 이를 위하여 먼저 펜 기반 교정 시스템 모델을 통하여 교정 부호와 편집 대상 텍스트의 속성을 정의하고 분류하였다. 정의된 모델을 이용하여 펜 기반의 교정 부호와 편집 대상 텍스트간의 영역 매칭 과정에서 발생하는 모호성을 해결하기 위하여 텍스트의 영역을 다양한 기준선으로 세분화 하고 입력된 교정 부호의 특성 및 입력 영역에 따라 서로 다른 규칙을 적용하여 텍스트를 글자, 단어, 문장, 절 등의 영역으로 인식하도록 하였다. 결과적으로 모호한 교정 부호의 입력을 인지적인 관점에서 해석하여 전통적 교정 모델의 교정 환경에서 보다 정확한 교정 작업 수행을 지원하도록 하였다.

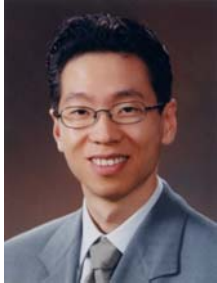
또한 현재 대부분의 교정 작업은 HTML이나 XML과 같은 웹 기반의 구조문서 환경에서 교정이 이루어지므로 제안된 모델을 이용하여 구조 정보 변경 규칙을 적용할 수 있다.

향후 본 연구진은 문서 구조 변경을 위한 규칙 모델 정의와 다중 사용자 환경에서의 협업작업 및 문서 변경 탐지를 통한 문서 버전관리와 관련하여 연구를 수행할 예정이다.

참고문헌

- [1] Ackerman, S.S. & Turechek, W.W., "The risks and rewards of online editing," IEEE Transactions on Professional Communication, Vol. 31-3, pp. 122 - 123, 1988.
- [2] Amaya, <http://www.w3.org/Amaya>, 1997.
- [3] Andre, J. & Helene R., "Paper-less Editing and Proofreading of Electronic Documents," in Proceedings of EuroTex'99, 1999.
- [4] Catherine C. Marshall,(1997). Annotation: From Paper Books to Digital Library, Proc. of the 2nd ACM International Conference on Digital Libraries, ACM, Philadelphia.
- [5] Christine, A. & Randall, D., "Resolving Ambiguities to Create a Natural Sketch Based Interface," in Proceedings of IJCAI-2001, 2001.
- [6] Farkas, D.K. & Poltrock, S.E., "Online editing, mark-up models, and the workplace lives of editors and writers," IEEE Transactions on Professional Communication, Vol. 38-2, pp. 110 - 117, 1995.
- [7] Goldberg, D. & Goodisman, A., "Stylus user interfaces for manipulating text," in Proceedings of the fourth annual ACM symposium on UIST, ACM Press, 1995.
- [8] Hardock, G, Kurtenbach, G. & W. Buxton, "A Marking based Interface for Collaborative Writing," in Proceedings of the sixth annual ACM symposium on User interface software and technology, Atlanta, ACM Press, NY, pp. 259-266, 1993.
- [9] Meyer, A., "Pen Computing: A Technology Overview and a Vision," in Proceedings of ACM SIGCHI Bulletin, ACM Press, pp. 46-90, 1995.
- [10] Miller, R. C. & Myers, B. A., "Multiple Selections in Smart Text Editing," in Proceedings of the 6th International Conference on Intelligent User Interfaces (IUI 2002), San Francisco, pp. 103-110, 2002.
- [11] Ogata, H., Yano, Y. & Wakita, R., "CCML: Exchanging Marked-up Documents in a Networked Writing Classroom," Computer Assisted Language Learning, Vol. 11-2, pp. 201-214, 1998.
- [12] Richey, H. & Lorette, G., "On-line correction of Web pages," Document Analysis and Recognition, ICDAR '99, Proceedings of the Fifth International Conference, pp. 581-584, 1999.
- [13] Rubine, D., "Specifying Gestures by Example," Computer Graphics, Vol. 25-4, pp. 329-337, 1991.
- [14] Sohn, W. S., et al., "Standardization of eBook documents in the Korean Industry," Computer Standards & Interfaces, Vol. 24-1, 45-60, 2002.
- [15] University of Chicago Press, The Chicago Manual of Style, 14th edition, Chicago: University of Chicago Press, 1993.
- [16] EduPAD, http://www.intel.com/cd/corporate/iscsc/apac/eng/tech_innovation/273649.htm #pa d

저자소개



손원성

1998년 동국대학교 컴퓨터공학(학사)

2000년 동국대학교 컴퓨터공학(석사)

2004년 연세대학교 컴퓨터과학(박사)

2004년 4월 ~ 2006년 2월

Carnegie Mellon University, Post Doc.

2006년 3월 ~ 현재 경인교육대학교 컴퓨터교육과 전임강사

연구분야 : 웹 문서처리, HCI, Mobile