

History를 검색하기 위한 Learning UCC 메타데이터 Reference 모델 설계

*송주홍 **이은미 ***문남미

*, ***호서대학교 벤처전문대학원 **명지전문대학

juhongeda@naver.com

Learning UCC Metadata Reference Model Design for History Search

*Song, Ju-Hong ** Lee, Eun Mee ***Moon, Nam-Mee

*, ***HOSEO GRADUATE SCHOOL OF VENTURE

**MYONGJI COLLEGE

요약

Web 2.0 시대의 대표적인 콘텐츠인 UCC와 Learning 콘텐츠를 접목시킨다면, 효과적인 지식공유와 습득을 유도할 수 있을 것이다. 두 콘텐츠를 접목시켰을 때 일어날 수 있는 전문성과 신뢰성, 저작권법 관련 등의 문제점을 효과적으로 해결하기 위한 방안으로 History 검색을 제안한다. 또 그에 따른 메타데이터 설계와 데이터 검색방법을 제안한다. 효율적인 UCC기반 Learning history 검색을 위해서 reference 요소관리가 가능하도록 DB를 설계하였다. 저작도구를 이용하여 제작된 Learning UCC는 reference 요소를 통해 소스 콘텐츠에 관한 정보를 가지고 올 수 있으며, reference DB에 저장된 정보에 따라, 사용자에게 reference에 관한 정보들을 효과적으로 제공할 수 있다. 저작도구를 사용해야 한다는 제한성과 검색 효율성을 보다 개선한다면 Learning UCC를 관리하기 위한 효과적인 시스템이 될 것이다.

1. 서론

인터넷 사용자가 급증하고, web 2.0 시대가 도래함에 따라, 수많은 User Creative Content(이하 UCC)들이 웹을 통해 공유되고, 배포되고 있다. web 2.0의 대표적인 콘텐츠라 할 수 있는 UCC의 특성상, 누구나 쉽게 접근이 가능하고, 제작이 가능하다. 이러한 UCC 특징에 learning을 접목시키게 되면 쉽게 자신의 지식을 남과 공유하고, 필요지식을 습득할 수 있게 될 수 있을 것이다.

우리는 이미 text 기반의 지식공유기반 서비스를 사용하고 있다. 이러한 지식공유가 text뿐만 아니라, media를 통해서 제공하게 되면 좀 더 효과적인 서비스가 될 수 있을 것이다. 하지만 여기에는 콘텐츠의 전문성과 신뢰성, 근래 크게 대두 되고 있는 저작권법 등 몇 가지 문제점이 존재하게 된다.

이러한 문제점을 효과적으로 해결하고자, Learning UCC의 메타데이터 구조를 제안하고, 그 제안을 바탕으로 메타데이터 reference 모델설계와 콘텐츠간의 연관관계를 활용한 효과적인 데이터 검색방법을 제시해보고자 한다.

2. 관련연구

가. 디지털 콘텐츠의 메타데이터 관련 연구

UCC 와 learning 콘텐츠는 디지털콘텐츠라는 공통점을 바탕으로 두 개의 성격이 다른 콘텐츠를 하나로 묶어 주고 그와 관련된 기반요소들을 하나의 메타데이터로 기술, 관리 할 수가 있다. 이렇게 성격이 다른

콘텐츠들을 통합적으로 관리 해주기 위한 통합 메타데이터 기반의 연구들이 지속적으로 진행되고 있다[1][2].

Universal Content Identifier(이하 UCI)의 경우 콘텐츠 관리를 위한 메타데이터 설계 및 시스템 개발 관련 연구들을 진행하고 있으며, Service Oriented interactive Video Application(이하 SoIVA)의 경우 관리에서 한발 더 앞서, 콘텐츠들의 저작권 관련 표준 연구가 진행되고 있는 실정이다.

(1) Universal Content Identifier (UCI)

한국콘텐츠진흥원에 따르면 UCI는 식별 가능한 자원의 효율적인 유통과 활용을 위하여 개별 자원에 유일한 코드를 부여하고 이를 관리해주는 체계 또는 상이한 식별체계 간의 연계표준을 말한다[3].

UCI의 콘텐츠 관리는 UCI요소와 identifier를 통해 이루어진다.

또 contribution, contributor, contributorRole 요소를 통해, 콘텐츠에 영향을 미친 기여, 기여자, 역할을 확인 할 수 있다.

그 밖의 요소로 title, type, moderm, format을 가지고 있다.

(표 1) UCI의 메타데이터 요소[3]

용어명		정의	유니코드	인코딩 스킴	
영문	한글				
1	UCI	UCI	한국정보사회진흥원에서 부여하는 고유한 자원 식별자	1,1	구문구조
2	identifier	기본식별자	UCI 이외 자원에 부여된 기존의 식별자	0,n	ISBN외 디스
3	title	제목	발표된 자원의 이름	1,n	
4	type	유형	자원의 주된 유형	1,1	uciType
5	mode	표현형태	디지털자원의 주된 길리적 혹은 지리적 표현 형태	0,1	uciMode
6	format	파일형식	디지털자원의 데이터 표현형식	0,1	uciFormat
7	contribution	기여	자원의 내용에 주된 책임을 가진 주제의 역할의 의미를 포괄하는 요소 * 기여자와 기여지역의 상위요소 container	1,n	
8	contributor	기여자	자원의 내용에 주된 책임을 가진 자의 이름 * contribution의 하위 요소	1,1	
9	contributorRole	기여지역	자원의 내용에 주된 책임을 가진 자의 역할 * contribution의 하위 요소	1,1	uciRole

1) 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No. 2009-0080970).

(2) Service Oriented interactive Video Application (SoiVA)

SOiVA는 서비스 지향 아키텍처(Service Oriented Architecture : SOA)를 기반으로 양방향 동영상(interactive Video : iVideo)을 기존의 웹을 대체하는 동영상 기반의 웹을 실현하는 기술로서 iVideo 전송 및 SOiVA서비스 정의와 메시징 기술, SOiVA플랫폼 기술, SOiVA 연동 기술 등을 대상으로 하고 있다[4][5].

SOiVA의 메타데이터는 효과적인 콘텐츠 관리를 위해, Identifier를 유일한 전역 식별자를 두고 있으며, Right 요소에 저작권을 기술해 준다. 그 밖에 Title, Description, Genre, Keyword, Language, CreationDate, ProgramRef, FileFormat, Creator, MediaDuration, Relation을 메타데이터로 가지고 있다.

(표 2) SOiVA의 메타데이터 요소 [5]

	용어명	정의
1	Title	iVideo의 제목
2	Identifier	iVideo의 유일한 전역 식별자
3	Description	iVideo에 대한 부가적 설명
4	Genre	iVideo의 장르
5	Keyword	iVideo를 나타내는 특정단어
6	Language	iVideo의 주사용 언어
7	CreationDate	iVideo의 생성일자
8	ProgramRef	iVideo의 인스턴스를 가리키는 참조 식별자
9	FileFormat	iVideo를 이루는 파일 포맷
10	Creator	iVideo 제작자
11	MediaDuration	iVideo 유효기간
12	Relation	iVideo와 관련된 기타 정보
13	Right	저작권 기술

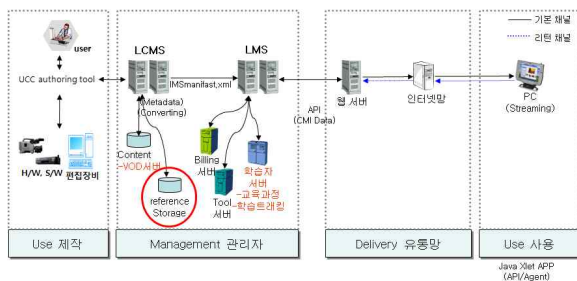
나. UCC의 형태 분석

UCC는 만들어 지는 형태에 따라 UGC, UMC, URC 세가지 분류로 세분화 된다[6]. 이것을 표 3과 같이 refernce 소스의 존재 유무에 따라 순수창작물(이하 UGC)과, 소스콘텐츠를 이용한 변형물(이하 UMC,URC)로 나뉘어 보았다.

(표 3) UCC의 형태 분석 ([6]의 재수정)

분류	세분화	약어	예	유형	
형태	Generated	UGC	A	A가 고유한 창작에 의한 콘텐츠	순수창작물
	Modified	UMC	A+ a =A'	소스콘텐츠 A에 사용자의 아이디어 a를 덧붙인 A'는 A와 제작의도가 동일함	
	Recreated	URC	A+B=C	서로 다른 콘텐츠 AB를 조합하여 새로운 콘텐츠 C가 생성되었지만 제작의도는 A와 다른 고유한 아이디어로 이루어짐	

3. Learning UCC 실험 구성도



(그림 1) 실험 구성도

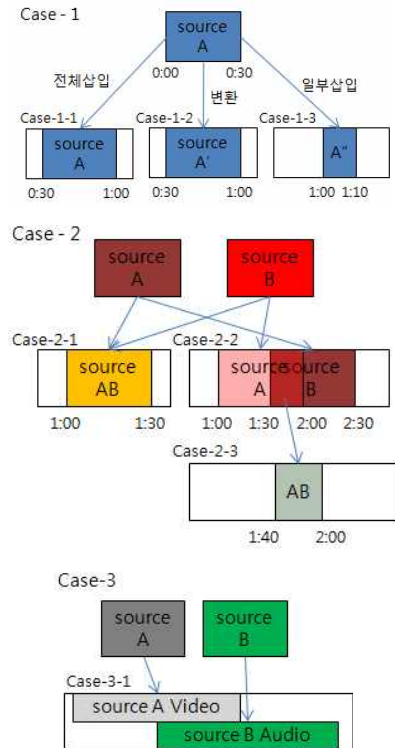
Learning UCC(이하 L_UCC)의 가장 큰 특징은 UCC에 교육을 접목시켰다는데 있다. 그렇기 때문에 일반 UCC들과는 틀리게 Learning Content Management System(이하 LCMS)에 의해서 콘텐츠가 관리되게 된다. LCMS가 reference Storage와 통신함으로써, 콘텐츠간의 연관관계를 밝혀 줄 수 있고, History 검색이 가능하게 된다. 유저는 인터넷망을 통해서 L_UCC서비스에 접속하게 되며, 제작자는 저작도구의 idnetifier check process를 통해, 저작권 여부를 확인 할 수 있다.

4. L_UCC의 메타데이터 reference 모델 설계

UCC의 증가와 함께, UCC의 메타데이터 관리와, UCC 저작권 보호에 대한 방법과 제안들도 크게 대두되고 있다. 따라서 L_UCC 또한 메타데이터 설계를 하는데 있어서, 기존의 디지털콘텐츠 메타데이터와, UCC의 형태, 저작권 보호에 관한 부분도 고려해 보아야 한다.

가. UCC 콘텐츠 저작 Case 분석

표3의 UCC 형태 분석에 따른 소스변형물의 유형에 따라 다음과 같은 case로 나뉘어보았다. (그림 2)의 case 1은 하나의 소스를 이용하여 영상이 만들어지는 경우이다. (그림 2)의 case 2는 두 개 이상의 소스를 이용하여 영상이 만들어지는 경우이다. 2개 이상의 소스를 합성해 새로운 영상을 만들기도 하고, 합성한 특정 부분만이 만들어 지기도 한다, (그림 2)의 case3은 소스의 특정부분-UCC의 Audio,Video 등-만을 가지고 왔을 경우이다.



(그림 2) UMC,URC의 case

크게 나는 3개의 case들을 연결한다면, 무수히 많은 변형케이스들이 존재하게 된다. 즉 원본 소스의 출처를 찾기 위해선 임의의 케이스에 관한 많은 변수들을 따져야만 한다.

이러한 변수들을 따져 최상위 원본소스를 찾는 것은 발생가능한 모든 경우의 수를 거쳐야 하는 비효율성 때문에, 상위 소스를 메타데이터에 명시해 놓음으로, 상위 소스를 찾아 올라 갈수 있도록 하는 것이다.

나. 메타데이터 설계

SOiVA와 UCI의 metadata를 근간으로 하여 메타데이터에 reference 요소를 추가하였다. reference 요소에는 L_UCC가 reference한 콘텐츠 소스에 관한 정보가 들어가게 된다.

(표 4) L_UCC의 메타데이터

용어명	정의
1 Title	L_UCC의 제목
2 Identifier	L_UCC의 유일한 전역 식별자
3 Description	L_UCC에 대한 부가 설명
4 Genre	L_UCC이 속하는 장르
5 Keyword	L_UCC를 나타내는 특정 단어
6 Language	L_UCC의 주 사용 언어
7 CreationDate	L_UCC의 생성일자
8 ProgramRef	L_UCC의 인스턴스를 가리키는 참조 식별자
9 FileFormat	L_UCC를 이루는 파일 포맷
10 Creator	L_UCC제작자
11 MediaDuration	L_UCC유효 기간, 유효 범위
12 Relation	L_UCC와 관련된 기타 정보 (URL, URI 등)
13 Rights	L_UCC저작권
14 Reference	L_UCC에 기여 및 공헌한 참조 아이템/아이템 제작자

reference 요소는 Children요소로 identifier, startTime, duration을 가지고 있다.

identifier에는 reference 한 소스의 identifier를 명시해준다. startTime은 제작물에서 reference한 소스가 시작 되는 시간을 명시해준다. duration은 제작물에서 refernece한 소스의 진행시간을 명시해준다.

(표 5) L_UCC의 reference 요소 구분

Diagram	
Children	<identifier><startTime><duration>
Source	<pre> <xs:element name="reference"> <xs:complexType> <xs:sequence minOccurs="0"> <xs:element name="identifier"/> <xs:element name="startTime"/> <xs:element name="duration"/> </xs:sequence> </xs:complexType> </xs:element> </pre>

reference요소의 활용방법은 다음과 같다.

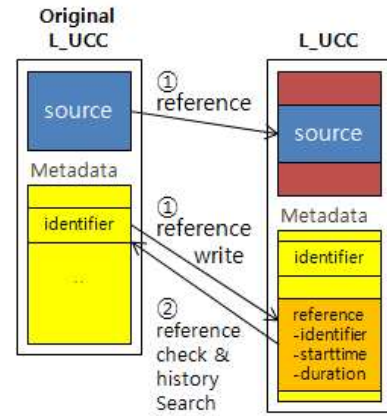
- ① Original L_UCC로부터 특정 소스부분을 refernece 하는 경우, Original L_UCC의 identifier를 L_UCC의 reference 부분에 기록해지게 된다.
- ② L_UCC의 reference들은, reference check & History 검색을 하는데 활용 하게 된다.

```

if( L_UCC.reference.identifier == Original L_UCC.identifier )
  L_UCC ≅ Original L_UCC Source

```

identifier의 reference를 이용해 refernce한 소스를 알 수 있고, startTime과 duration을 이용해, 제작물이 어떠한 타임에, 어떠한 소스가 사용되었는지 확인 할 수가 있다.



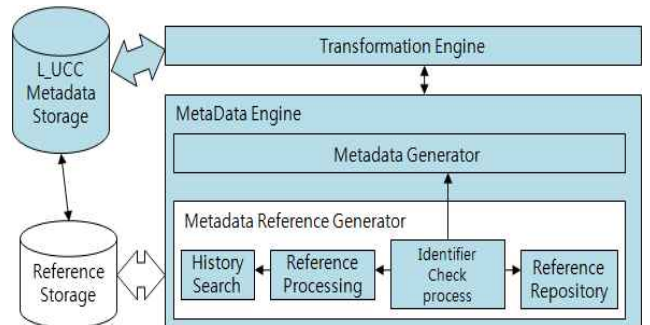
(그림 3) reference 요소를 이용한 Metadata의 원리

여기에는 전제조건들이 따르게 된다.

공통된 부여규칙에 의해 L_UCC들의 identifier가 부여되어야 한다. 또 부여된 identifier들이 관리 될 수 있는 시스템 또한 필요하다.

5. History 검색 시스템 구조 및 기능 설계

History 검색을 위해 LCMS 시스템 구조에 reference Storage를 추가하였고, MetaData Engine에 Metadata Reference Generator를 추가하였다. reference Storage에 콘텐츠간의 reference관계가 저장되게 되며, 이 data와 Metadata Reference Generator(이하 MRG)를 이용해 History 검색을 하게 된다.



(그림 4) history 검색을 위한 시스템 구조

가. Metadata Reference Generator

reference 메타데이터는 MRG를 거쳐, 등록되고 가공된다.

MRG는 Identifier Check Process(이하 ICP), Reference Porcessing(이하 RP), HistorySearch(이하 HS), Reference Repository(이하 RR)로 이루어져 있다.

ICP는 효과적인 콘텐츠 관리와 History검색을 위해, Identifier의 형식을 체크한다. Identifier는 System에서 format형식에 따라 자동으로 부여하도록 한다.

RP는 Reference Storage에 등록된 데이터를 가공하는데 쓰이는 process이다. 여기서 가공된 정보는 History 검색에 쓰일 뿐만 아니라, 콘텐츠의 reference 횟수와 reference 형태를 보여주는데도 활용되어진다.

HS는 가공된 데이터를 history화 하여 보여주는 프로세스이다. 유저는 HS를 통해, 콘텐츠의 출처를 확인 할 수가 있다. 콘텐츠의 출처를 확인함으로써, 콘텐츠의 신뢰도를 높이는데 기여하고, 저작권 분란을 피할 수가 있다.

RS는 L_UCC 메타데이터의 reference요소를 refernce Storage에 등록

하는 프로세스이다. 등록된 메타데이터들은 RP와 HS과정을 거쳐 가공된 데이터를 만드는데 활용되게 된다.

MRG의 processing 과정은 (표 6)과 같다. L_UCC 메타데이터의 Identifier 형식을 체크해서, 형식이 일치한다면, RP 과정을 거쳐, HS에 활용된다. 또는 Reference 정보를 Reference Storage에 등록한다. 등록된 데이터는 추후 RP과정에 쓰이게 된다. 불 일치 한다면, 새로운 Identifier를 부여한 후 콘텐츠 메타데이터를 등록하여 준다.

(표 6) Metadata Reference Generator Process

```
Identifier Check Process{
  if( L_UCC.identifier.format == source.identifier.format )
  { case 0:
    Reference Processing;
    History Search;
  case 1:
    Reference Repository; }
  else
  { Identifier Check Process.Creater Identifier;
    Metadata Generator; }
}
```

나. Reference Storage

referene한 콘텐츠를 찾기 위해, 콘텐츠 메타데이터의 reference요소를 매번 확인 하는 것은, 다수의 콘텐츠 메타데이터에 접근하는 경우, 과부하를 가지고 올 수 있기 때문에 효율적인 방법이라 할 수 없다. 그래서 시스템상에 Reference Storage를 구축하고 reference 데이터를 따로 관리하고, 활용 할 수 있도록 하였다.

reference Storage에는 콘텐츠간의 연관관계가 저장되게 된다 (표 7)은 ReferenceDB의 구성이다. Sid는 DB에서 관리되는 primary key이다. Parent UCC ID에는 reference한 소스의 identifier가 들어간다. UCC ID에는 만들어진 UCC의 identifier가 들어간다. Modifier에는 reference의 형태를 넣어주게 된다.

Parent UCC ID와 UCC ID는 matadata Storage의 UCCidentfier 연관 관계를 이루어 줌으로써, 콘텐츠에 관한 정보를 가져 올수 있도록 한다.

(표 7) Reference DB의 구성

Column Name	DataType
Sid	INTEGER
Parent UCC ID	VARCHAR(10)
UCC ID	VARCHAR(10)
Modifier	VARCHAR(5)

6. 결론

지금까지 History 검색을 위한 L_UCC의 메타데이터 및 DB 설계와 History 시스템 구조에 대해서 정의, 설계 하였다.

History 검색을 통해서 소스 콘텐츠를 찾아 올라감으로써, UCC의 출처가 명확해지고, 콘텐츠의 신뢰도를 높이는데 기여할 수 있을 것이다. 또 소스콘텐츠의 소유권을 명확히 명시해 주게 됨으로써, 저작권 분란 또한 피할 수 있을 것이라 생각한다.

그러나 특정 저작도구를 사용해야 한다는 제한사항과, ID를 관리하는 특정 시스템을 통해야만 한다는 점이 본 연구의 한계사항이다. 또 여러

개의 복수 소스 콘텐츠를 가졌을 때의 과부하 역시 해결해야 할 연구 과제라 할 수 있다. Modifier의 효율적인 활용을 위한 정확한 기준과 분류에 관한 구체적인 논의도 이루어져야 할 것이다.

History 검색의 실제 활용성과 효율성을 위해선 연구의 한계사항과, 연구과제 관한 해결방안, Modifier에 관한 논의가 지속적으로 이루어져야 할 것이다.

7. 참고문헌

- [1] "웹 콘텐츠 저작권 보호를 제공하는 통합 메타데이터 기반의 다중 CMS 시스템", 조영복, 한국통신학회논문지, 2008
- [2] "미디어 정보의 통합을 위한 메타데이터", 송철환, 한국정보과학회 가을 학술발표논문집, 2004
- [3] UCI-차세대 디지털 콘텐츠 유통의 인프라(url : <http://www.uci.or.kr>)
- [4] "정보통신 중점기술 표준화로드맵 Ver.2008", TTA, 한국정보통신 기술협회, 2008
- [5] "SOiVA 양방향 동영상 분류 체계 및 시연 시스템 개발", 한국전자통신연구원 보고서, 2008
- [6]"You! UCC", 마케팅사관학교&김영한, 랜덤하우스, 2007.2 p.30
- [7] "IPTV 기반 Interactive T-Learning 표준화방안 연구", (주)씨티유니온, 한국전자거래진흥원, 2009.2