

# 효과적인 e-러닝 시스템 구축을 위한 과제물 표절 검사

Reports Plagiarism Inspection for Efficient Implementing e-learning System

조동욱, 홍윤선, 조선옥  
충북과학대학

Cho Dong-Uk, Hong Yoon-Sun, Cho Sun-Ok  
Chungbuk Provincial University of Science & Technology

## 요약

최근 e-러닝시스템에 대한 사회적 관심이 증대되고 있다. 그러나 효율적인 e-러닝시스템이 되기 위해서는 온라인상에서 제출된 과제물에 대한 표절 검사가 가장 중요한 사항이 된다. 본 논문에서는 효율적인 e-러닝시스템 구축을 위해 과제물의 표절 검사를 행하기 위한 방법론과 관련 소프트웨어 툴에 대한 비교, 분석을 행하고자 한다.

## Abstract

Recently, social interest are increasing to e-learning system. For realizing efficient e-learning system, reports plagiarism inspection is the most important topic. This paper describes the methods of reports plagiarism inspection and analyzing the S/W tools to implement e-learning system.

## 1. 서론

현재 국내의 경우 불법소프트웨어의 복제율이 50%를 넘어서고 있어 이에 대한 디지털 재산권의 보호가 시급한 실정이다[1][2].

이의 중요성을 인지한 각 나라에서도 이를 위해 예방 정책(preventative control)과 억제 정책(deterrent control)을 행하고 있는데 가장 좋은 방법은 예방 정책과 억제 정책이 적절히 조화를 이루는 가운데 예방 정책으로 나아가는 즉, 소프트웨어프로그램에 대한 감정이 효과적이며 강력히 수행되어야만 한다. 다시 말해 불법 소프트웨어 복제에 대한 형사 처벌의 강화보다는 불법 복제 프로그램의 감정을 효과적으로 수행 할 수 있는 기관의 설립과 운영 그리고 표절을 효과적으로 검출할 수 있는 감정 도구의 개발과 활용 그리고 기술적 감정 기법의 개발[3]-[6]이 문제 해결의 핵심이라고 할 수 있다. 이를 위해 많은 연구들이 행해져 왔다[7]-[8]. 그러나 수많은 연구들이 실제 소송중인 소프트웨어 프로그램에 대한 표절 검출보다는 학교에서 학생들이 제출한 프로그램에 대한 표절 검출 등에 한해 사용 가능한 것이 절대적이어서 이를 e-learning 시스템에 있어 과제물의 표절을 위한 검출 방법과 툴 이용에 적용하면 대

단히 큰 효과를 거두리라 여겨진다. 실제로 현재 표절 감정에 관해 가장 방대한 양의 연구와 활동을 한 단체는 영국의 JISC이다. 이 단체는 국내로 예를 들자면 고등교육연구위원회와 같은 조직이다. 따라서 본 고에서는 효과적인 e-learning 시스템의 구축을 위해 현재 존재하고 있는 소프트웨어 프로그램의 표절 감정 방법과 이의 툴에 대한 분석을 통해 실제적으로 학교 현장에서 효과적으로 쓰일 수 있는 프로그램 과제물의 표절 검출 방법에 대해 다루고자 한다. 아울러 프로그램 뿐 아니라 다큐먼트 파일에 대해서도 표절을 검출할 수 있는 방법론의 소개와 이와 관련된 검출 툴에 대해 살펴봄으로써 e-learning 시스템에 있어 가장 큰 문제점이었던 과제물에 대한 표절 행위를 효과적으로 검출하여 실제 현장에서 유익하게 쓰일 수 있는 방법에 대해 소개하고자 한다. 이를 위해 우선적으로 프로그램의 표절을 감정하는 방법론에 대해 살펴볼 것이다. 차후 이와 관련된 각종 툴에 대한 소개와 이의 비교, 분석을 행하고자 한다. 또한 다큐먼트 파일에 대해 표절하는 방법론에 대한 기술과 이와 관련된 개발툴에 대한 비교, 분석을 통해 효과적으로 과제물 전체에 대한 표절 행위 검출과 운영방법에 대해 다루고자 한다.

## 2. 소프트웨어 프로그램 과제의 표절 감정 방법론

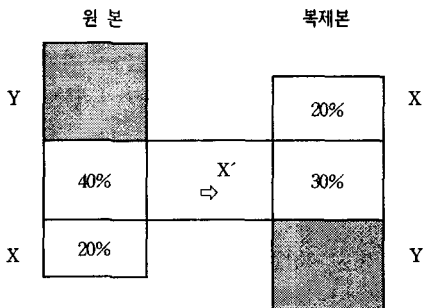
소프트웨어 프로그램의 표절을 감정하기 위해서는 A, B 두 프로그램의 복제도를 계산해야 한다. 이는 우선 원본 기준방식과 복제본 기준방식 그리고 기준 영역 선정방식과 복제 조사 방법에 따라 구분되어질 수 있으며 이는 표절감정을 위한 대상 소프트웨어 프로그램에 따라 적절히 선택해야 할 것이라 여겨진다.

### 2.1 원본 기준방식과 복제본 기준방식

원본 기준방식은 원본프로그램의 얼마나 많은 부분이 도용되었는지를 기준으로 하는 방식이며 이에 비해 복제본 기준방식은 복제본의 얼마나 많은 부분이 복제한 것인지를 기준으로 한다. 이때 원본 기준 방식은 원본이 복제본보다 추가된 기능이 많을수록 복제도가 작아지며 복제본 기준 방식은 독자적으로 추가한 부분이 많을수록 복제도가 작아진다.

### 2.2 기준영역 선정방식

이는 크게 전체 영역 기준방식과 공통영역 기준방식으로 나뉜다. 이중 전체영역 기준방식은 원본 소스 프로그램 전체와 복제본 소스 프로그램 전체가 감정대상이며 독자적으로 새로 추가된 부분이 많을수록 복제도는 작아진다. 공통영역 기준방식은 원본 소스프로그램과 복제본 소스프로그램 중 서로 공통된 부분만이 감정대상이며 추가된 부분에 거의 상관없이 복제도를 계산할 수 있다. 아래 그림1에 이의 예를 나타내었다.



- 원본 기준 계산 방식  
 ⇨전체 영역 기준 방식 : 복제도 =  $(X'/(X+Y)) = 40/100 = 40\%$   
 ⇨공통 영역 기준 방식 : 복제도 =  $(X'(X)) = 40/60 = 67\%$
- 복제본 기준 계산 방식

- ⇨전체 영역 기준 방식 : 복제도 =  $(X'/(X+Y)) = 30/100 = 30\%$
- ⇨공통 영역 기준 방식 : 복제도 =  $(X'(X)) = 30/50 = 60\%$

#### ▶▶ 그림1. 복제도 계산의 예

### 2.3 복제 조사 방법

이는 크게 전수 조사 방식과 표본 조사 방식으로 나뉜다. 전수 조사 방식은 감정할 과제물의 소스프로그램 전체를 감정하여 복제도를 계산하는 가장 일반적인 방식으로써 단순한 알고리즘이지만 수많은 서버 프로그램으로 이루어진 프로그램을 감정할 때는 너무나 많은 시간이 소요된다는 문제가 있다.

이에 비해 표본 조사 방식은 감정할 전체 과제물 소스 프로그램 중에서 표본 소스 프로그램을 추출하여 복제도를 계산한 후 그 복제도를 토대로 그 전체 과제물 소스 프로그램의 복제도를 추정하는 방식으로 전체 과제 소스프로그램을 대표하는 표본을 제대로 추출하지 못하면 잘못된 복제도가 계산되는 문제가 존재한다. 또한 최종 프로그램의 복제도는 weighted sum 형태가 될 수 있다.

## 3. 소프트웨어 프로그램 과제의 표절을 감정기 위한 틀 분석

본 장에서는 소프트웨어 프로그램의 표절 감정을 가장 많이 현재까지 해온 단체인 영국의 JISC에 대한 소개와 프로그램 과제의 표절 등을 감정하기 위한 방법론에 대한 소개 등을 행하고자 한다.

### 3.1.1 국외의 대표적 활동 단체 및 기관

국외에서 대표적으로 프로그램의 표절을 검출하기 위한 다각도의 노력을 경주해온 기관은 영국의 JISC (Joint Information Systems Committee)이다.

JISC는 영국, 스코틀랜드, 웨일즈와 북아일랜드에서 고등 교육의 질적 향상을 위한 재단 법인 FE and HE (Further Education and Higher Education) 산하의 전략적 자문 위원회로서 프로그램 표절에 대한 검출 및 방지에 대한 사업을 행하고 있다[8]. 이 기관은 CAI에 관한 사업을 주로 하고 있으며 보다 효과적인 고등교육 기관에 대한 다양한 사업을 전개하고 있다. 이중에 우리

에게 주로 관계되어 있는 분야가 바로 프로그램 과제에 대한 표절 검출이며 표1.에 JISC의 표절 검출 및 방지 사업에 대한 주요 내용을 나타내었다.

**[표1] 전자 표절 검출 및 방지를 위한 JISC 프로젝트의 주된 사업내용**

A technical review of free-text plagiarism detection S/W
Source code plagiarism detection S/W
A small size free-text detection S/W
A good practice guide to plagiarism prevention

### 3.1.2 국내의 경우

국내의 경우 KAIST 이광근 교수팀이 개발한 프로그램 표절 검출 툴인 clonechecker와 부산대 조환규 교수팀이 개발한 LOFC가 주된 프로그램 표절 검출 툴이다. 또한 교수클럽에서 자연어로 된 전자 파일에 대한 표절을 검출하기 위한 툴을 개발하였으며 현재 서비스를 시행중에 있다. 아래 표2.에 이에 대한 대략적인 요약표를 나타내었다.

**[표2] 국내에서 개발된 과제물에 대한 표절 감정 툴에 대한 소개**

기관	관련사이트 및 기타
프로그램심의 조정위원회	<a href="http://www.pdmc.ac.kr">http://www.pdmc.ac.kr</a>
한국소프트웨어 감정 평가학회	2002년12월5일에 창립되어 주로 민사상 문제가 제기된 프로그램에 대해 감정을 수행하고 있음
부산대학교	<a href="http://jade.cs.pusan.ac.kr/~hgcho">http://jade.cs.pusan.ac.kr/~hgcho</a>
한국과학기술원	<a href="http://ropas.kaist.ac.kr/n/clonechecker">http://ropas.kaist.ac.kr/n/clonechecker</a>
교수클럽	<a href="http://www.gyosuclub.com">http://www.gyosuclub.com</a>

## 3.2 프로그램 표절 검출 소프트웨어 툴의 비교

현재까지 일반에게 알려진 감정도구, 특히 소프트웨어 표절 검출 도구는 제공하는 형태에 따라 크게 세 가지로 나눌 수 있다.

### 3.2.1 제공되는 형태에 따른 분류

(1) 상업용 소프트웨어 : 유료로 판매되고 있으며 이를 사용하기 위해서는 라이선스를 받아야 하는 소프

- 트웨어로 McCabe와 Windiff가 여기에 해당된다.
- (2) 인터넷 서비스 : 인터넷을 이용하여 표절 검출을 무료로 해주는 서비스를 하는 소프트웨어로 Moss와 JPlag이 이 부류에 해당한다.
- (3) 공개소프트웨어 : 인터넷을 이용하여 소스코드 또는 실행파일을 다운로드한 후 자신의 컴퓨터에 설치하여 사용하는 소프트웨어로 이 부류에는 SIM, YAP, CloneChecker가 있다.

### 3.2.2 표절검출 방법론에 대한 고찰

프로그램 표절 검출에 쓰일 수 있는 알고리즘은 크게 Attribute Counting 방법과 Structure Metric 방법으로 나뉘어진다. 이 중 Attribute Counting 방법은 사용된 단어 등의 유사성이나 빈도 등을 검사하는 방법이고 Structure Metric 방법은 구조적인 방법을 사용하여 정확한 정합이 아닌 토큰 스트링(token string)의 유사성을 계산하는 방법이다. Structure Metric 방법이 Attribute Counting 방법보다 효율적인 방법으로 평가되고 있으며 Plague, SIM, YAP 등의 표절 검출 툴에 사용되고 있다.

### 3.2.3 표절 검출 툴에 대한 비교, 분석

(1) Windiff

Microsoft Visual Studio 안에서 제공하는 도구로서 디렉토리, 파일 비교가 가능하며 윈도우 환경에서 바로 사용이 가능하다. 사용자 인터페이스가 GUI를 제공하여 편리성을 제공해주며 소스코드 비교가 가능하다.

(2) McCabe

McCabe는 모듈의 사이즈와 구조를 결정하는 하나의 모듈을 통해서 독립적인 실행 경로의 수를 계산함으로써 소프트웨어 품질을 측정한다. McCabe는 표절 검출을 위한 도구가 아니므로 정확하게 이 목적으로 사용할 수 있는 도구를 제공하지 않으며, 상업적 소프트웨어로서 고가로 널리 유통되지는 않는다.

(3) Moss

이 도구의 목적은 C, C++, Java, Pascal, Ada, ML,

Lisp 또는 Scheme으로 작성된 소스코드의 유사성을 측정하는 것이다. 현재까지 Moss의 주요 적용분야는 프로그래밍 수업에서 표절을 검출하는 것이며, 이 목적에 있어서는 매우 효과적이었다.

이 도구는 인터넷 서비스 형태로 제공되며 인터넷을 통해서 소스 파일 묶음이 주어지면 유사한 소스코드를 가진 프로그램의 쌍들을 나열한 HTML 페이지를 출력한다. 또한, 프로그램 안의 개별적인 경로들을 하이라이트로 표시하여 쉽게 추적, 비교할 수 있게 해준다. 그리고, 공유하는 것이 당연한 소스 코드는 유사성 검사에서 제외시켜준다. Window나 UNIX 시스템에서 모두 사용할 수 있다.

#### (4) JPlag

이 도구는 C, C++, Java, Scheme 그리고 자연어로 작성된 여러 세트의 프로그램 소스 코드의 유사성을 검출한다. 사용방법은 문장 및 프로그램 구조를 함께 비교하여 준다. 이렇게 함으로써 표절을 위장하는 여러 가지 방식에 대처할 수 있게 해준다. 표3.은 Moss와 JPlag에서 검출 가능한 언어별로 비교한 것이다.

[표 3] Moss와 JPlag의 검출 언어 비교

	JAVA	C++	C	Pascal	Ada	ML	Lisp	Scheme
MOSS	0	0	0	0	0	0	0	0
JPlag	0	0	0					0

#### (5) SIM

SIM은 소프트웨어 프로젝트에서의 표절과 대형 소프트웨어 프로젝트에서 복제가 의심되는 코드 조각을 검출할 수 있다. SIM의 출력은 히스토그램 EH는 의심되는 프로그램 제출물의 목록을 작성하기 위한 셸 스크립트로 처리할 수 있으며, 토큰을 기본 단위로 하여, 연속되는 여러 개의 토큰으로 구성되는 런을 비교 단위로 하여 두 파일을 비교한다. SIM의 기본적인 알고리즘은 표 4.와 같다.

[표 4] SIM의 기본 알고리즘

1. 프로그램 파일을 읽어들인다.
2. 전향 참조표를 작성한다. 최소 크기를 만족하는 각 텍스트 서브스트링을 자신의 오른쪽에 있는 다른 모든 서브스트링과 비교한다. 이 결과가 전향 참조 표인데, 동일한 위치로 해석되는 다음 서브스트링의 인덱스를 저장한다. 유사한 다음 서브스트링이 발견되지 않았으면 인덱스는 0이다.
3. 의미있는 런들의 집합을 결정한다.
4. 의미있는 런들의 줄 번호를 결정한다.
5. 런들의 내용을 순서대로 출력한다.

SIM은 메뉴 방식이 아닌 명령줄 방식으로 사용하도록 되어 있어 몇 가지 명령어를 암기하거나 찾아봐야 하는 단점이 있다.

#### (5) YAP

YAP의 목적은 이전의 속성 계수 방법이나 구조 기반 방법들보다 성공적이었던 Plague를 기반으로 하되, Plague의 문제점을 극복할 수 있게 하는 것이었다. 모든 YAP 시스템들은 같은 방식으로 작동하며 두 단계로 구성된다.

1단계 : 각 제출물에 대해서 하나의 토큰 파일을 생성한다.

2단계 : 토큰 파일의 쌍들을 비교한다.

YAP에 의해서 검출될 수 있는 복제 행위에는 표 5.와 같은 것들이 있다. 또한 표 6.은 YAP를 이용한 정밀도 수준을 나타낸다.

[표 5] YAP에 의한 검출 가능한 복제

- 주석이나 출력 포맷 변경
- 식별자 변경
- 연산수의 변경
- 데이터 타입 변경
- 식을 동등한 다른 것으로 변경

[표 6] YAP 1, 2, 3의 정밀도 수준

	Q/Q1	Q/Q2	Q1/Q2
YAP1	73	55	65
YAP2	98	56	62
YAP33	97	69	69
YAP32	100	75	75

(6) CloneChecker

이 도구는 두 프로그램을 비교해서 유사도를 0~1사이의 실수값으로 알려준다. 단순히 두 프로그램의 텍스트만을 비교하는 것과 달리, 프로그램에서 사용된 이름 등을 생략하고 구문 구조만을 비교함으로써 변수 이름 바꾸기, 함수 순서 바꾸기 등에 영향을 받지 않는다. C, Java, S초든, nML로 짜여진 프로그램 등에 대하여 사용할 수 있다. CloneChecker의 유사성 검사 방법은 요약, 유사성 비교, 그룹짓기의 세 과정을 거친다. 두 번째 단계인 유사성은 표 7.에서와 같이 구해진다.

[표 7] CloneChecker에서 유사성의 정의

$$\text{similarity} = \frac{S_a + S_b}{T_a + T_b}$$

Sa = number of sub-tree of a that appears identically in b  
 Sb = number of sub-tree of b that appears identically in a  
 Ta = number of total nodes(sub-trees) in tree a  
 Tb = number of total nodes(sub-trees) in tree b

세 번째 단계에서는 유사성에 따라 프로그램을 기본적으로 표 8.과 같이 그룹 짓는다. 그룹 안의 임의의 두 프로그램의 유사성은 0과 1사이의 실수인 전역유사성보다 큰 값을 갖게 된다.

[표 8] 유사성에 의한 그룹 짓기

정의 1. 전역유사성 g에 의해 정의된 그룹 G는  
 $\forall a \in G, \forall b \in G, \text{similarity}(a,b) \geq g$   
 정의 2. 지역유사성 1에 의해 정의된 그룹 G는  
 $\forall a \in G, \exists b \in G, \text{similarity}(a,b) \geq 1$

3.2.2 개발 툴에 대한 방향 제시

최근의 소프트웨어개발 방향은 비주얼 환경에서 객체 지향 기법의 언어를 사용하고 있으며 Web과 GUI를 활용하고 있다. 이를 위해서는 개인용 컴퓨터의 윈도우 환경에서 사용할 수 있는 감정도구가 필요하며, 파일구조와 소스코드에 대한 많은 정보를 제공해 주어야 할 것이다.

[표 9] 표절 검출 감정 도구의 개발

기존도구	개발방향
-UNIX, 도스환경 -명령어 형태 -구조적 언어기반 -스트링 매칭 -단일 파일 비교 -사용에 폐쇄적 -문법적, 토큰 형태 분석	-개인용 컴퓨터, 윈도우 환경 -GUI 형태, Web 제공 -비주얼 언어기반 -여러 파일 비교기능 -공개소프트, 참조파일 등 -논리형 흐름 분석 -컨텐츠 감정기법 -언어의 변형에 대처 (C언어->C++, JAVA 등)

4. 다큐먼트 과제의 표절을 검출키 위한 방법론과 관련 툴의 분석

다큐먼트형태의 전자파일은 아래 표 10.과 같으며 표절을 검출키 위해 통상 통계적 기법을 사용하며 주요 표절 검출 방법은 아래 표 11.과 같다.

[표 10] 자연어 표절의 형태

Copying directly form the source
Rewording a sentence (paraphrasing) <ul style="list-style-type: none"> <li>Using appropriate synonyms</li> <li>Changing the sentence type</li> <li>Changing the order of a sentence</li> <li>Reducing a clause to a phrase</li> <li>Changing the part-of-speech</li> <li>Making abstract ideas more concrete</li> </ul>
text summarization <ul style="list-style-type: none"> <li>Sentence reduction</li> <li>Sentence combination</li> <li>Syntactic transformation</li> <li>Lexical paraphrasing</li> <li>Generalisation / specification</li> <li>Sentence reordering</li> </ul>

통상 통계적 방법을 이용한 클러스터링 방법, 지식 기반 방법을 이용한 클러스터링 방법 그리고 위의 두 방법을 결합한 하이브리드 방법 등이 있다.

4.1 국외의 경우

다큐먼트 자연어 표절 검출을 위한 S/W 툴에 대해 아래 표 12.에 나타내었다.

[표 12] 자연어 표절 검출 소프트웨어

시스템	개발 회사명	웹 사이트
Findsame	Digital Integrity	http://www.findsame.com
EVE2	CaNexus	http://www.CaNexus.com
Turnitin	iParadigms	http://www.turnitin.com
CopyCatch	CFL S/W Developments	http://www.CopyCatch.freeerve.co.uk
WordCHECK	WordCHECK Systems	http://www.WordCHECKsystems.com

[표 13] 다큐먼트 표절 검출의 주요 요소

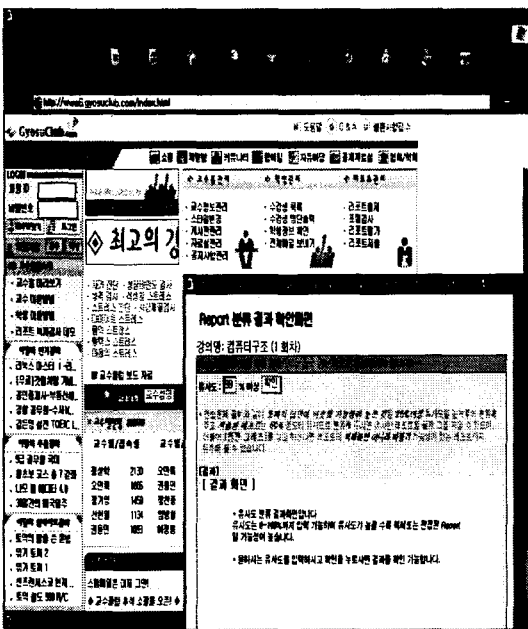
- the average length of sentences (words)
- the average length of paragraphs (sentences)
- the use of passive voice (expressed as a percentage)
- the number of preposition as a percentage of the total number of words
- the frequency of "function words" used in each text

## 4.2 국내의 경우

국내의 경우 (주) 교수클럽이 학생들의 레포트 표절 검출을 위해 개발 하였다. 아래 표 13.에 이의 웹사이트 소개를 그리고 그림 2에 하나의 예를 나타내었다.

[표 13] 자연어 표절 검출 소프트웨어

시스템	개발회사명	웹 사이트
교수클럽	교수클럽	http://www.gyosuclub.com



▶▶ 그림 2. 레포트 표절 검출의 예

## 5. 프로그램에 대한 재산 가치 평가

현재까지는 학교 현장에서 그리고 가장 큰 대상은 e-learning에서 과제물의 효과적인 표절 감정을 위한 여러 가지 방법론과 관련 툴을 비교, 분석하였다. 그러나 표절 감정은 학교 현장이 아닌 일반 업체들 사이에서도 회사의 사운이 걸린 일로 상당한 문제를 야기할 수 있다. 이의 경우 감정을 수행하고 그 결과 불법 복제로 판정되었을 시 이에 대한 민사상의 재산 보상이 이루어져야 한다. 이를 위해 소프트웨어의 가치 평가가 이루어져야 하는데 최종적으로 이에 대해 살펴보고자 한다. 또한 이는 소프트웨어 프로그램을 담보로 대출시도 고려해야 하는 방법들이다.

### 5.1 비용 접근법(Cost Approach)

이 방법은 동일한 가치를 얻기 위해 사용된 물적, 인적 자원을 금액으로 산출한 것으로서 산정 방식은 활동 기준 회계법, 재생산 또는 대체 비용법 등이 있다. 그러나 이는 향후 기대 수익 예측, 미래 경제에 대한 정보와 기대 수익의 획득에 수반되는 위험 등을 고려하지 못하는 문제점이 있다.

#### 5.1.1 활동 기준 회계법(Activity based costing)

지적 재산의 대상을 만드는 활동에 소모된 사람, 기계/장비, 재료 등을 통하여 원가 대상별, 활동별 가치를 산출하는 방식이다.

#### 5.1.2 재작성(Reproduction) 비용 계산법

유사한 자산을 다시 작성하는 데 필요한 노력과 비용을 직접 계산하는 방법이다.

### 5.2 시장 사례 접근법(Market approach)

이 방법은 시장에서 이루어지는 유사 거래 사례를 참조하여 평가하는 방법으로써 활발한 공개 시장이 존재하고 비교 가능한 자산이 거래되고 있을 시 가능한 방법이다.

### 5.3 수익 접근법(Income approach)

이 방법은 유사 자산 거래 사례가 없어 시장 접근법을 사용할 수 없고 대체 원가 산정 자료가 불충분하여 비용 접근법을 사용할 수 없을 경우에 적용할 수 있다. 평가 대상 자산으로 인한 미래 기대 수익 예측, 현재 가치화하는 것으로 현금 흐름 할인법이 있다. 단점으로는 예측의 어려움, 지적 재산의 기여도 산정 어려움, 미래의 불확실성과 위험요소의 반영이 어렵다는 문제가 있다.

#### 5.3.1 현금 흐름 할인법(Discounted cash flow)

지적 자산이 장래에 산출할 수 있는 회전 자금(현금 흐름)에 할인율을 적용하여 현재가치로 평가하는 방법으로 현재 가치는 아래 수식과 같이 하여 구한다.

$$Pv = \sum_{t=0}^n \frac{CFt}{(1+r)^t}$$

여기서 Pv : 현재가치, n : 자산의 수명

CFt : t 기간의 현금흐름, r : 현금 흐름의 위험을 반영한 할인율이다.

### 5.4 옵션 가치 평가법

옵션이라 함은 특정한 조건하에서만 가치를 갖는 자산을 말하며 콜 옵션은 살 수 있는 권리, 풋 옵션은 팔 수 있는 권리를 뜻한다. Black-sholes의 OPM(Option Pricing Model)을 이용한다.

### 5.5 기술 요소법(Technology factor method)

이 방법은 일정한 자격을 갖춘 평가자와 전문가가 패널을 통해 평가하는 방법으로써 특허 판매 및 관리에 실제 응용된다. 주된 평가 항목으로는 기술의 사업/제품에 공헌도, 기술의 효용속성, 기술의 경쟁력 등이 되며 일관된 측정 방법을 사용할 수 있다는 장점이 있다.

## 6. 결론

본 고에서는 e-learning을 위해 각종 전자 파일로 된 문서와 프로그램에 대한 과제물 소프트웨어 프로그램의

표절 검출 방법과 각종 틀에 대한 비교, 분석을 행하였다. 이를 통해 향후 과제물 소프트웨어 프로그램의 표절을 검출하기 위한 틀이 가져야 할 성질과 개발 방향에 대한 한 방향을 제시하였다. 아울러 소프트웨어 프로그램의 자산 가치 평가에 대해서도 다름으로서 향후 확장된 표절 감시기 재산 가치를 평가할 수 있는 방법들을 소개하였다. 차후로 실제 현장인 e-learning 시스템과 민사 사건에 효과적으로 활용될 수 있는 소프트웨어 틀을 개발하여 이의 분야에 도움을 주도록 하기 위한 연구가 지속적으로 수행되어야 하리라 여겨진다.

#### ■ 참고문헌 ■

- [1] 조동욱의, "프로그램 감정기법 개발에 관한 연구", 한국 s/w 감정연구회 수주 프로젝트(프로그램심의조정위원회 최종보고서), 2002년 2월.
- [2] 조동욱의, "컴퓨터 s/w 감정관련 국내·외 동향조사 및 분석", 프로그램심의 조정위원회 최종보고서, 2002년 11월.
- [3] 이규대, "디지털 콘텐츠의 특징 및 감정 활용 방법", 한국 s/w 감정연구회 학술대회 논문집, 2002년 12월.
- [4] 김도완, "소프트웨어 가치 평가 방법", 한국 s/w 감정연구회 학술대회 논문집, 2002년 12월.
- [5] 안개일, "병원 전자 차트 프로그램에 대한 감정", 한국 s/w 감정연구회 학술대회 논문집, 2002년 12월.
- [6] 박성균, "계측기의 복제도에 대한 감정", 한국 s/w 감정연구회 학술대회 논문집, 2002년 12월.
- [7] 각국의 소프트웨어 불법 복제 방지 대책에 관한 연구, 프로그램 심의 조정위원회, 2002년 2월.
- [8] <http://www.jisc.ac.uk/jciel/plagiarism>