Journal of the Korea Academia-Industrial cooperation Society
Vol. 23, No. 8 pp. 664-672, 2022

https://doi.org/10.5762/KAIS.2022.23.8.664
ISSN 1975-4701 / eISSN 2288-4688

히스토그램 매칭 방식을 이용한 효율적인 전자문서 유사도 분석 방안에 관한 연구

김영식[1], 백종경[2], 박재표[3*]
[1]숭실대학교 정보과학대학원 정보보안학과, [2]㈜인피니솔루션 연구소, [3]숭실대학교 정보과학대학원

A Study on the Efficient Electronic Document Similarity Analysis using Histogram Matching Method

Yeong-Sik Kim[1], Jong-Kyung Baek[2], Jae-Pyo Park[3*]
[1]Division of Information Security, Graduate School of Information Science, Soongsil University
[2]R&D Center, Infinisolution Co., Ltd.
[3]Graduate School of Information Science, Soongsil University

요 약 지식재산권의 보호가 중요시하게 여겨지는 현대 사회에서 전자문서 유사도 분석은 필수이지만, 여러 상용 소프트웨어는 전자문서 유사도 분석을 인위적으로 회피하면 검출률이 매우 낮아진다. 본 논문에서는 상용 소프트웨어의 기존 분석 방법을 개선하기 위해서 컴퓨터 비전을 활용해 전자문서를 분석한다. 제안 시스템은 전자문서를 문장 단위로 구분하고, 각 문장을 바이너리 이미지로 인식하며, 이렇게 만들어진 이미지를 히스토그램으로 비교 분석할 수 있다. 시료는 총 세 가지 유형으로 분류하였으며, 전자문서 971개 안에 문장 37,960개를 비교 데이터로 활용한다. 세 가지 유형은 어절의 순서만 변형한 유형, 단어의 뜻은 같으나 음절 단위로 단어를 변형한 유형, 비슷한 의미의 단어로 대체한 유형으로 분류한다. 전체적인 평균 수치를 분석하면 제안 시스템을 활용하여 유사도 검사를 했을 때 검출률이 상용 소프트웨어보다 18.3 % 높은 수치를 보였다. 제안 시스템은 추후 전자문서 유사도 분석뿐만 아니라 전자문서의 색인화 및 분류, 서식 인식 등 전자문서가 활용되는 모든 분야에서 활용할 수 있다.

Abstract In today's world, where protection of intellectual property rights is important, electronic document similarity analysis is essential. On the other hand, some commercial software has a very high detection rate if electronic document similarity analysis can be artificially avoided. This paper uses computer vision to analyze electronic documents and improve the existing analysis method of commercial software mentioned above. The proposal system divides electronic documents into sentence units and recognizes the delimited sentences as binary images. Each sentence is converted into an image, and the corresponding image can be compared and analyzed with a histogram. The samples were classified into three types, and 37,960 sentences were used as comparative data in 971 electronic documents. The three types were classified into types. In the first type, only the order of words was transformed. The second type was where words were transformed into syllable units. In the last type, words were replaced with words of similar meanings. Analyzing the overall average value, when the similarity test was performed using the proposed system, the value was 18.3% higher than that of commercial software.

Keywords : Histogram Matching, Binary Imaging, Similarity Analysis, Documents Plagiarism, Paraphrasing

*Corresponding Author : Jae-Pyo Park(Soongsil Univ.)
email: pjerry@ssu.ac.kr
Received May 30, 2022 Revised August 1, 2022
Accepted August 3, 2022 Published August 31, 2022

1. 서론

최근 비대면 서비스의 확장으로 종이로 된 문서보다 전자문서의 활용도가 급격한 속도로 증가하고 있다. 정부에서는 이러한 변화를 인지하여 '전자문서 및 전자거래 기본법'을 개정했다. 이를 통해 전자문서도 종이로 된 문서처럼 법적 효력을 가질 수 있게 되었다. 이렇게 전자문서의 비중이 중요해진 만큼 시중에 전자문서와 관련된 많은 서비스가 나타났다. 그중에서도 특히 사회에서 중요하게 생각하고 있는 서비스는 지식재산의 보호를 위한 전자문서 유사도 분석 서비스이다. 현재 상용 소프트웨어의 전자문서 유사도 분석 서비스는 단순히 주제가 비슷하여 유사한 단어가 자주 쓰이는 것과 타인의 결과물을 무단으로 발췌한 행위를 구분하는 것이 어렵다. 그래서 본 논문에서는 이러한 타인의 지식재산을 무단으로 발췌한 행위를 총 세 가지 유형으로 분류하여 효율적인 전자문서 유사도 분석 시스템을 제안한다. 이 시스템은 최근에 활발히 연구되고 있는 컴퓨터 비전을 활용하여, 전자문서를 바이너리로 인식해서 히스토그램으로 분석하는 시스템이다[1].

본 논문의 2장에서는 전자문서를 바이너리로 인식하여 이미지로 변환하는 방법과 히스토그램 제작 방법을 설명하고, 3장에서는 히스토그램 분석 결과를 활용하는 방안을 제시한다. 4장에서는 검증을 위한 구현 및 성능평가를 하고, 5장에서는 결론 및 향후 연구 방향을 제안한다.

2. 관련 연구

2.1 악성코드의 이미지화 방법

개인별 PC 사용이 일상화된 시대에서 악성코드를 미리 탐지하는 것은 매우 중요한 과제이다. 악성코드는 새로운 형태가 만들어지는 것보다 기존 악성코드를 기반으로 한 변종이 상대적으로 많다[2]. 그래서 악성코드를 이미지화하여 인공신경망으로 딥러닝 기술을 활용하면, 악성코드를 빠르게 식별해낼 수 있다[2]. 기존 악성코드를 직접 분석하여 찾아내는 방법보다 패턴 매칭을 활용하여 탐지하는 것이 악성코드 변종 찾기도 쉽고 탐색 속도도 매우 빠르다[3]. 본 논문은 해당 기술을 활용하여 연구하고자 하는 전자문서의 유사도 분석 분야에 적용하였다.

2.2 OpenCV를 활용한 히스토그램 분석

OpenCV는 두 히스토그램 간에 유사도를 측정할 수 있는 함수를 제공하고 있다. 해당 함수에는 총 네 가지의 분석 방법이 있는데, 상관관계(correlation) 분석, Bhatacharyya Distance 분석, 카이제곱(chi-square) 분석, 교차(intersection) 분석이 있다. 본 논문에서는 상관관계 분석과 Bhattacharyya Distance 분석, 교차 분석 등이다.

상관관계 분석은 두 변수 사이의 선형적 관계를 분석하기 위해 상관계수를 구하는 방법으로써, 공분산을 표준편차의 곱으로 나누어 계산한다. Bhattacharyya Distance 분석은 히스토그램의 값들을 전체 히스토그램으로 나누어 합이 1이 되도록 만드는 분포 유사도 분석 방법이고, 교차 분석은 두 히스토그램의 겹치는 영역 값을 전부 더하여 분석하는 방법이다.

각 분석 방법에 따라 완벽하게 일치하는 이미지, 절반 정도 일치하는 이미지와 완벽하게 일치하지 않는 이미지를 비교한 결과를 Table 1에 나타내었다.

Table 1. Histogram Similarity

Hist.	Correlation	Chi-Square	Intersection	Bhattacharyya distance
Exact Match	1.0	0.0	-	0.0
Half Match	0.7	-	-	0.55
Mismatch	0	-	0.0	1.0

각 값은 상댓값이므로 표준 단위가 없다. Exact Match는 완벽하게 일치하는 이미지를 비교한 값이고, Mismatch는 완전 다른 두 이미지를 비교한 값이다. 카이제곱 분석값과 교차 분석값은 값이 존재하지 않는 경우가 있는데, 이는 이미지의 크기에 따라 값의 변동 폭이 매우 크기 때문이다. 본 논문에서 사용하는 문장의 이미지는 크기가 다양하게 존재하므로, 카이제곱 분석과 교차 분석은 히스토그램 분석에서 유효성 검증을 위한 방안을 제시한다[4].

2.3 상용 소프트웨어 및 기존 연구 유사도 분석의 문제점

전자문서 유사도 분석 상용 소프트웨어는 '카피킬러', 'TurnItIn' 그리고 'WCopyFind' 등이 있다. 카피킬러는 분석하고자 하는 문서를 1문장 혹은 6어절 단위로 나누고, 이를 '카피킬러' 내부에서 가지고 있는 빅데이터를 활용하여 검사 대상 문서와 비교한다. 교육부 지침상, 1문장에서 6어절 이상 같은 어절이 있는 경우

의심 문장으로 분류하게 되어 있다[5]. ‘TurnItin’은 총 단어 수가 ‘TurnItIn’ 데이터베이스에 존재하는 단어 중 일치하는 단어 개수로 유사도를 측정한다.

상용 소프트웨어의 목적은 수많은 데이터베이스 안에 문장과 빠른 비교를 하는 것이 목적이다. 그래서 단순히 어절 단위로 같은지 비교하므로 음절 단위에 작은 변조에도 유사도가 낮은 문장이라고 분석한다.

기존 연구는 형태소 분석을 이용한 유사도 분석의 한계점을 극복하는 연구가 대다수이다[6]. 한 연구는 문장, 어절, 단어 기반의 유사도 분석을 필요에 따라 분석한다. 이 연구에서 제안하는 시스템은 일차적으로 단어 기반으로 유사도를 측정하고 기준보다 높은 유사도를 가진 문서에 대해 표절 의심 문서로 분류한다[7]. 그 다음 기준보다 낮은 유사도를 보인 문장에 대해서는 추가로 문장 유사도 분석과 어절 유사도 분석을 하는 방법이다[7].

다른 연구로 전자문서가 구조적 특징을 가질 수 있도록 XML 스키마를 활용한 연구가 있다. 이 연구는 전자문서를 XML 문서로 변환하고, AST 활용하여 분석하였다[8]. 해당 연구에서 제안하는 시스템으로 분석하면 분석 속도가 빠르고 문장의 길이 차이가 나는 문서도 효율적으로 분석할 수 있다.

기존 연구의 제안 시스템들은 형태소 분석을 이용한 유사도 판별보다 좋은 결과를 보인다. 하지만 인위적으로 의미가 비슷한 단어로 대체하여 문장을 변조할 때는 유사도가 높은 문장으로 검출하지 못한다.

3. 이미지 히스토그램 분석을 활용한 전자문서 간 유사도 비교 방안

이미지 히스토그램을 분석하는 기술은 단순히 한 문장 안에 여섯 개의 어절이 같은지 판단하는 것이 아니라, 6어절 이상인 문장을 대상으로, 이미지 히스토그램을 만들어 비교 분석하는 기술이다.

이는 형태소를 파악하여 파생 단어들을 전부 비교하고, 오탈자 검사도 진행하면서 검사하는 기술보다 매우 경제적이다. 또, 상용 소프트웨어처럼 빅데이터를 기반으로 한 단순 비교보다 더 많은 범위의 유사한 문장을 검출할 수 있다. 제안 시스템의 구성도를 보면 Fig. 1과 같다.

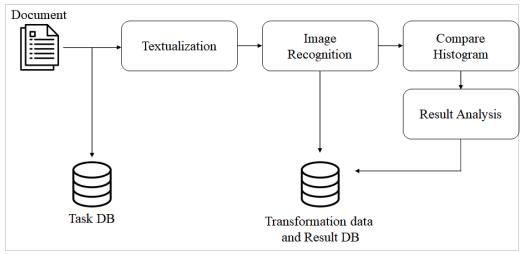


Fig. 1. Image Histogram Comparison System

분석 대상 문서를 로컬에서 제안 시스템으로 폴더 단위 등록이 가능하다. 제안 시스템은 전자문서를 등록하면, 일차적으로 분석 대상이라는 플래그 값과 함께 그 전자문서를 데이터베이스에 저장한다. 데이터베이스에 저장된 문서 중 분석 대상 문서들은 텍스트화를 거쳐 문장 단위로 나누어 이미지로 변환하고, 문장과 변환된 이미지를 연결하여 다시 데이터베이스에 저장한다. 이후, 제안 시스템은 이미지 데이터를 분석하기 위해 스레드 풀을 생성하고 데이터베이스에서 가져온 이미지 데이터를 멀티 스레드로 분배한다. 각 스레드는 문장 이미지를 히스토그램으로 변환하여 비교한다. 비교한 수치를 가지고 분석한 결과를 데이터베이스에 저장한다.

3.1 문서의 문장 단위 데이터 변환

전자문서를 공백과 개행 문자, 문장을 끝마치는 마침표 등 여러 기준을 이용하여 문장 단위로 나눌 수 있다. 마이크로소프트 오피스 문서의 경우 공개 소스 라이브러리를 활용하여 UTF-8 인코딩인 텍스트 파일로 변환한다. 한글 문서의 경우 자동화를 사용하여 텍스트 파일로 변환한다.

3.2 문장 바이너리 이미지 인식

히스토그램 매칭 분석을 활용하기 위해서는 문장을 이미지 정보로 변환해야 한다. 그래서 UTF-8 인코딩 텍스트 파일로 변환된 전자문서를 문장 단위로 나누고, 각 문장 데이터를 바이트 단위로 이미지에 삽입한다. 1 바이트에 해당하는 값을 픽셀 한 칸에 1:1로 대응하여 벡터 이미지를 생성한 결과가 Table 2와 같다.

다섯 가지의 예시 문장을 바이너리 이미지로 변환하였다. 이미지는 전부 한 줄 이미지인데 길이를 보면 문장에 따라 이미지의 길이가 다른 것을 볼 수 있다. 이는 문장의 길이를 반영하여 이미지 데이터를 생성하기 때문이다. 그래서 반대로 이미지의 길이를 보면 문장의 길이를

알 수 있고, 이미지의 길이는 유사도를 비교할 때 하나의 요소로 작용한다.

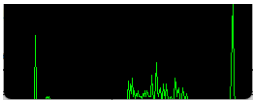
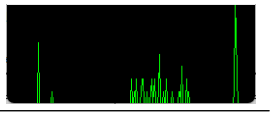
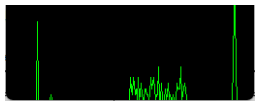
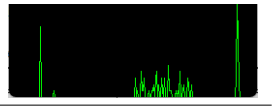
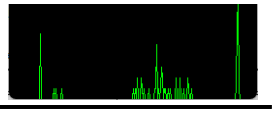
Table 2. Changing from Sentence to Binary Image

Sentences	Binary Images
Due to the increase in age, income activities are reduced, or the sudden death of the head of the household, who is responsible for the family's livelihood, or disease, the standard of living of the family or the family left behind deteriorates rapidly.	
They say that measures against this are not easy.	
And even if you have been without major diseases or injuries, it is natural that you are worried about your retirement life when your age increases.	
It can be said that Korea has a very low pension age compared to other countries.	
Currently, the age of supply and demand in Korea is 60 years old.	

3.3 히스토그램 매칭 비교 방안

문장 간 유사도를 분석하기 위해서는 이미지 데이터를 히스토그램으로 변환하는 과정이 필요하다. Table 2에서 다섯 가지의 예시 문장을 이미지로 변환하였다. 이러한 이미지 데이터를 활용하여 히스토그램으로 변환시킨 데이터는 Table 3과 같다.

Table 3. Creating Histograms with Binary Images

Sentences	Histograms
Due to the increase in age, income activities are reduced, or the sudden death of the head of the household, who is responsible for the family's livelihood, or disease, the standard of living of the family or the family left behind deteriorates rapidly.	
They say that measures against this are not easy.	
And even if you have been without major diseases or injuries, it is natural that you are worried about your retirement life when your age increases.	
It can be said that Korea has a very low pension age compared to other countries.	
Currently, the age of supply and demand in Korea is 60 years old.	

본 연구에서는 상관관계 분석과 Bhattacharyya Distance 분석을 적극적으로 활용하였다.

상관관계 분석은 선형적 유사 관계 분석 방법이고, Bhattacharyya Distance 분석은 이미지의 크기가 서로 다른 두 히스토그램의 분포 유사도 분석 방법이다. 따라서 이 두 방법을 적절하게 활용하기 위해서는 가중 평균이 필요하다. 그래서 교차 분석값을 활용하여 가중치를 구한 식은 Eq. (1)과 같다[9].

$$g = \sum_{I} \frac{\min(H_1(I), H_2(I))}{H_1(I)} \tag{1}$$

이미지 히스토그램 분석 기법을 이용하면 전체적으로 유사도가 높다. 그래서 분산도를 높이기 위해서 가중치와 더불어 이미지의 크기를 고려한 연산식을 구성한다[10].

가중치와 이미지의 크기를 상관관계 식과 Bhattacharyya Distance 식에 대입하면 Eq. (2)와 같다.

$$\begin{aligned} r = & \sqrt{1 - \frac{1}{\sqrt{H_1 H_2 N^2}} \sum_{I} \sqrt{H_1(I) \times H_2(I)}} \times g \\ & + \frac{\sum_{I} (H_1(I) - \overline{H_1})(H_2(I) - \overline{H_2})}{\sqrt{\sum_{I} (H_I(I) - \overline{H_1})^2 \sum_{I} (H_2(I) - \overline{H_2})^2}} \times (1 - g) \\ & - \frac{\left| \sum_{I} H_1(I) - \sum_{I} H_2(I) \right|}{\max(\sum_{I} H_1(I), \sum_{I} H_2(I))} \times 100 \end{aligned} \tag{2}$$

근호가 포함된 수식은 Bhattacharyya Distance 수식이다. 이 수식을 이용하여 크기가 서로 다른 두 이미지 히스토그램을 확률 분포와 같은 형태로 분석한다. Bhattacharyya Distance 수식의 하단부는 상관관계

식이다. 상관관계 분석을 통해 두 히스토그램의 선형적 유사 관계를 비교한다. 이때, 교차 분석값이 크면 분포적 유사도로 분석해야 한다는 것이고 그럴수록 Bhattacharyya Distance 분석 가중치를 높게 한다.

모든 문장에 대해 높은 유사도를 가진 문장과 낮은 유사도를 가진 문장 간의 유사도 차이가 작다. 이를 보완하기 위해서 두 히스토그램 차이를 히스토그램으로 나누어 계산하고 가장 하단에 있는 식과 같이 이미지 크기 차이를 식에 반영한다.

4. 비교분석 및 성능평가

4.1 시험 환경 및 방법

4.1.1 시험 환경

제안한 기술을 구현한 환경과 제작된 프로세스를 실행한 환경은 Table 4와 같다.

Table 4. Proposed System Test Environment

Environment		Explain
Software	IDE	Visual Studio 2022
	OS	Windows 11 Pro
	DBMS	SQLite
	Number of Thread	20
Hardware	CPU	Intel Core(TM) i5-9300H
	RAM	24 GB
	SSD	256 GB

제안 시스템은 Visual Studio 2022를 통해 C++ 바이너리 파일로 제작한다. 데이터베이스는 서버 구축 없이 디스크 파일의 액세스 권한만 있으면 사용할 수 있는 'SQLite'를 사용하여 분석할 문서와 분석 결과를 저장한다. 스레드 개수는 사용자의 환경에 따라 변경할 수 있다. 현재 4코어 CPU에서 스레드의 개수가 20개로 설정되어 있다. 이는 각 스레드에서 분석한 뒤 결과를 데이터베이스에 저장하기 위해 다중의 스레드가 접근할 때 데이터베이스 잠금이 일어난다. 그래서 스레드의 대기시간을 고려하여 설정한 개수이다.

분석할 수 있는 문서들의 형식과 실제 분석한 시료의 양을 나타낸 표는 Table 5와 같다.

Table 5. Information of Samples

Division		Explain
Supported File Format		docx, doc, xlsx, xls, ppt, pptx, pdf, hwp, html, txt, 7z, zip
Number of Comparable Sample Documents		971 EA
Sample	Type 1	1,598 EA
	Type 2	2,010 EA
	Type 3	802 EA
Total Sentences		37,960 EA

모든 표본은 실제 학생들이 과제물을 제출한 것을 이용하여 구성한다. 전자문서 971개, 전자문서 안에 문장 37,960개이고, 문장들을 타입별로 분류하면 위 Table 5와 같이 분류할 수 있다. Sample Type 1은 음절은 전혀 변하지 않고 어절의 순서만 바뀐 경우이다. 이 유형은 음절의 변화가 없으므로 비슷한 단어를 찾으면 검출할 수 있다. Sample Type 2는 전체적인 의미는 변하지 않으나 음절 단위로 변조가 일어난 문장이다. 이 유형은 음절의 변조가 있으므로 같은 단어를 찾을 때 검출되지 않을 수 있다. Sample Type 3은 같은 의미가 아닌 비슷한 의미의 다른 단어로 대체되는 유형이다. 가장 검출하기 어렵고, 오탐률이 높은 유형이다.

4.1.2 시험 방법

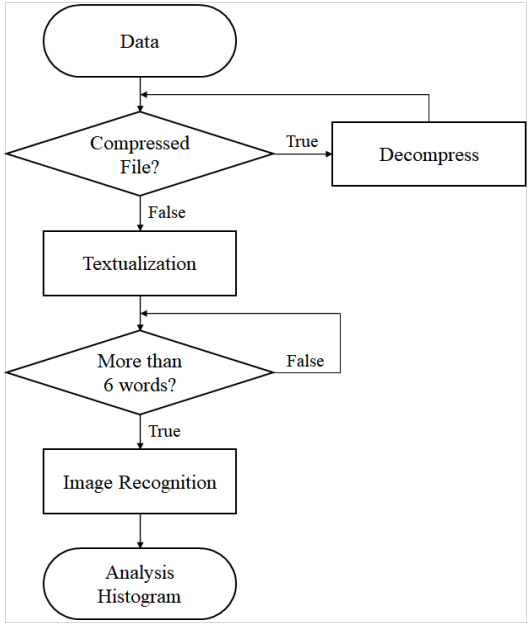


Fig. 2. Flowchart of Proposal System

주어진 시험 환경에서 제안 시스템으로 제작한 프로그램을 실행하면 분석을 시작할 수 있고, 분석에 대한 순서도는 Fig. 2와 같다.

지원하는 확장자 중에서 압축 파일인지 검사하고, 압축 파일이면 먼저 압축을 해제한다. 이후 비교가 가능한 모든 파일을 작업 데이터베이스 테이블에 등록하면 Fig. 3과 같다.

IDX	FILENAME	STATUS
572	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291rlat...	-1
573	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291rose...	-1
574	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291rou...	-1
575	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291sas...	-1
576	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291sent...	-1
577	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291shtn...	-1
578	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291silve...	-1
579	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291soe...	-1
580	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291soo...	-1
581	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291sora...	-1
582	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291sun...	-1
583	D:₩Sample2₩LESSON_ONLY_1291₩1₩LESSON_ONLY_1291sunf	-1

Fig. 3. List of Files to Analyze

각 전자문서의 경로가 파일이름 열에 저장되어 있고, 모든 전자문서가 분석 대상이라는 상태 값인 -1을 저장하고 있는 상태를 보여준다. 테이블 각 열의 정의는 Table 6과 같다.

Table 6. Database list of FILE table

Column Name	Data Type	Explain
IDX	INTEGER	Index
FILENAME	VARCHAR	Local Path of Electronic Documents
STATUS	INTEGER	Value of Status -1 : Analysis Object 0 : Analysis Done 1 : Access Denied

색인 값은 데이터베이스 내 검색 속도를 위한 값이고, 분석 대상인 전자문서의 경로를 FILENAME 열에 저장한다. 상태 값은 분석 대상인지, 분석이 완료되었는지, 오류가 발생한 파일인지 알려주는 값이다. 이후 작업 데이터베이스에 저장된 모든 문서를 불러와서 텍스트 파일로 만든다. 텍스트 파일로 만들어진 문서를 이미지로 제작하는 것은 6어절 이상으로 구성된 문장만 이미지로 인식한다. 이미지로 인식된 문장은 스레드 풀에서 각 스레드로 분배하여 일대다 분석을 하는데, 소스는 Fig. 4와 같다.

분배 스레드로부터 받은 한 문장의 이미지를 이용하여 반복문을 통해 1:1 문장 유사도 비교를 N번 실행한다. N은 비교하고자 하는 문장이 속한 문서가 아닌 다른 모든 전자문서의 문장 개수이다.

```
106  double SentenceCompare1ToN(int iParentIDX, int& iObjIDX, WCHAR* pwImgPath)
107  {
108      list<LPSEINFO>::iterator iter;
109      int iCount = 0;
110      double dMaxVal = 0.0f;
111      double dAvg = 0.0f;
112
113      // 반복문을 통해 가져온 이미지 파일과 타 전자문서를 모두 비교한다.
114      for (iter = s_listSEInfo.begin(); iter != s_listSEInfo.end(); iter++)
115      {
116          SEINFO* pSEInfo = *iter;
117          double dRet = 0.0f;
118
119          if (pSEInfo->iParentIDX != iParentIDX)
120          {
121              CHAR szSrcImgPath[MAX_PATH] = { 0, };
122              CHAR szTrgImgPath[MAX_PATH] = { 0, };
123
124              WideCharToMultiByte(CP_ACP, 0, pwImgPath, -1, szSrcImgPath, MAX_
125              WideCharToMultiByte(CP_ACP, 0, pSEInfo->wImgPath, -1, szTrgImgPa
126
127              // 문장 유사도 계산
128              dRet = CalculateCompareHistogram(szSrcImgPath, szTrgImgPath);
129
130              if (dMaxVal < dRet)
131              {
132                  dMaxVal = dRet;
133                  iObjIDX = pSEInfo->iSEIDX;
134                  iCount++;
135              }
136          }
137      }
```

Fig. 4. A Part of Sources of 1 : N Analysis

4.2 성능평가

본 연구의 시험 환경에서 제안 시스템을 구동하였을 때 자원 사용량은 Table 7과 같다.

Table 7. Resource Usage

CPU Usage per Thread	Memory Usage per Thread
21.2 %	24 MB

분석 스레드 개수가 증가할수록 CPU 사용량은 점점 더 큰 비중을 차지하지만, 메모리는 분배 스레드에서 다른 스레드로 작업을 할당하는 것이므로 큰 영향이 없다. 그래서 변동 없이 낮은 메모리 사용량으로 효율적인 분석이 가능하다. 실제로 구동하고 출력된 로그는 Fig. 5와 같다.

```
[8852] UCmpHist: --run thread: 0--   스레드 분석 시작
[8852] UCmpHist: CSEMgr::TextFileToSentenceImgFile, after delete blank strReadData(인터넷
[8852] UCmpHist: CSEMgr::TextFileToSentenceImgFile, iMinimum (21)
[8852] UCmpHist: CSEMgr::TextFileToSentenceImgFile, strSeperate(인터넷 등 자료를 인용하여
[8852] UCmpHist: CDBMgr::FindSEIDXFromDB, iIDX (750) iParentIDX(22) iSENum(2)
[8852] UCmpHist: AnalyzeFunction, Add Work iIDX(749) ParentIDX(22) wImgPath(C:₩Temp₩k
[8852] UCmpHist: WorkerThreadFunction, RESULT!!! tWork.fRate(72.1)
[8852] UCmpHist: --wait thread: 0--   스레드 작업 완료 후 대기
```

Fig. 5. Runtime Thread Debug Log

0번 스레드에서 작업을 시작하여 한 문장을 데이터베이스에 존재하는 다른 모든 문장과 비교한 결과이다. 유사도 결과가 72.1 %로, 정상적으로 구동되고 있음을 확인할 수 있다.

4.3 비교분석

제안 시스템, 상용 소프트웨어, 기존 연구와 분석 시스템을 비교한다. 문장을 위에서 언급한 타입으로 분류하여 분석하였다. Table 8부터 Table 10까지는 각각의 타입별로 분석한 결과를 나타낸 것이다.

Sample Type 1은 어절의 순서만 바뀐 경우이며, 분석 결과는 Table 8과 같다.

Table 8. Sample Type 1 : The number of sentence that only changed the order of phrases

Sentence	Number of sentences	80 ~ 90 %	90 ~ 100 %	Detection Ratio
Common Use Software	1,598 EA	834 EA	322 EA	**72.3 %**
Proposed system		859 EA	459 EA	**82.5 %**

어절의 순서만 변경된 경우는 제안 시스템만 아니라 모든 시스템에서 비교적 검출률이 높았다. 상용 소프트웨어는 약 72.3 %의 유사도 결과가 검출되었고, 제안 시스템에서는 유사도 결과가 약 82.5 %로 상용 소프트웨어보다 약 10.2 % 높게 검출되었다.

Sample Type 2는 어절이 아닌 음절 단위의 변화가 있는 유형이다. 예를 들어서, 오타를 내거나 인위적으로 조사 변형 등 음절 단위로 변형된 유형이고, 결과는 Table 9와 같다.

Table 9. Sample Type 2 : The number of sentence that changed the syllable

Sentence	Number of sentences	80 ~ 90 %	90 ~ 100 %	Detection Ratio
Common Use Software	2,010 EA	778 EA	311 EA	**54.2 %**
Proposed system		790 EA	492 EA	**63.8 %**

상용 소프트웨어에서는 한 음절만 다르더라도 완전히 다른 어절로 인식하기 때문에 54.2 %의 표절 검출률을 보였고, 제안하는 기술에 의한 표절 검출률은 63.8 %로 상용 소프트웨어보다 9.6 % 높게 나타냈다. Sample Type 2는 다른 유형들에 비해 상용 소프트웨어와 차이가 근소하다.

Sample Type 3은 비슷한 의미의 단어로 대체되는 경우이다. 일반적으로 이 경우가 가장 검출하기 어려운 유형이고, 결과는 Table 10과 같다.

Table 10. Sample Type 3 : The number of sentence that replaced with similar words

Sentence	Number of sentences	80 ~ 90 %	90 ~ 100 %	Detection Ratio
Common Use Software	802 EA	249 EA	47 EA	**36.9 %**
Proposed system		541 EA	109 EA	**81.1 %**

이 유형에서는 상용 소프트웨어에서 다른 유형보다 더 낮은 36.9 % 검출률을 보였다. 그러나 제안 기술로는 표절을 분류한 효율이 81.1 %로 44.2 % 더 높은 검출 결과를 보였다. 제안 시스템이 Sample Type 3에서 더 높은 검출 효율성을 가지고 있다. 모든 유형을 종합적으로 분석해보면 Table 11과 같다.

Table 11. Result of All

Sentence	Number of sentences	80 ~ 90 %	90 ~ 100 %	Detection Ratio
Common Use Software	4,410 EA	1,861 EA	680 EA	**57.6 %**
Data Structure Analysis System[8]		N/A	N/A	**74.0 %**
Proposed system		2,190 EA	1,060 EA	**75.9 %**

전체적인 결과를 보았을 때, 제안 시스템으로 분석했을 때, 인위적으로 문장을 무단으로 가져와 쓰는 행위를 검출하는 정확도가 더 높았다. 다른 논문에서 연구한 기술도 마찬가지로 기존 상용 소프트웨어보다 높은 검출률을 보였지만, 근소하게 제안 시스템의 탐지 비율이 높았다.

5. 결론

본 논문에서 기존 서비스 및 형태소 분석 방법에 대한 문제점을 제기하였고, 해당 문제점을 보완할 수 있는 이미지를 활용한 문장 유사도 분석 방법을 제시하였다. 제안하는 시스템은 기존 서비스와 성능적인 부분에서 유사한 문장을 검출할 수 있는 능력은 비슷하나, 기존 상용 소프트웨어에서는 검출할 수 없던 유사한 문장을 검출할

수 있었다.

실험 결과, 제안 기술은 이미지 기반 유사도 검색 시스템이므로, 기존 기술보다 전체적으로 높은 유사도 값을 보였다. 하지만 적절한 식을 세워서 유사한 문장과 유사하지 않은 문장 간의 분산을 높였다. 그래서 유형별로 각각 유사한 문장 검사에 대한 효율이, 문장의 어순만 바꾼 유형에서는 10.2 % 더 정확하게 유사한 문장을 찾았고, 문장에서 몇 음절들을 변환하였을 때는 9.6 %, 문장의 단어를 비슷한 의미의 단어로 교체하였을 때는 44.2 %가 향상된 검출률을 보였다.

본 논문 이후에, 제안 기술을 활용한 유사도 분석 서비스를 새롭게 시작하면 인위적으로 무단 표절을 한 전자문서를 검출할 수 있다. 이는 사회적으로 많은 지식인의 지식재산권을 보호할 수 있는 역할을 할 수 있다고 생각한다. 더 나아가서 제안하고 있는 기술과 색인 기술을 합쳐 기존에 존재하지 않는 전자문서에 대한 분류 시스템을 구축할 수 있을 것이다[11]. 이렇게 전자문서의 바이너리 이미지화 기술을 문서 유사도 검사뿐만 아니라 문서 서식 분류 등, 더 다양한 분야에 적용할 수 있을 것이다.

References

[1] J.-P. Park, et al. "Web-based Video Monitoring System on Real Time using Object Extraction and Tracking out", *Journal of the Institute of Electronics Engineers of Korea CI*, Vol 41, No. 4, Jul. 2004.

[2] M.-J. Song, J.-K. Lee, "Analysis of the impact of the visualization methods on deep learning-based malware classification performance", *Korean Journal of Military Art and Science,* 77, 1, pp. 511-530, Feb. 2021.
DOI: http://dx.doi.org/10.31066/kjmas.2021.77.1.019

[3] Y.-C. Hwang, H.-J. Mun, "Detection Model based on Deeplearning through the Characteristics Image of Malware", *Journal of Convergence for Information Technology*, Vol. 11. No. 11, pp. 137-142, 2021.
DOI: http://doi.org/10.22156/CS4SMB.2021.11.11.137

[4] J.-K. Baek, Y.-S. Jee, J.-P. Park, "A Personal Information Security System using Form Recognition and Optical Character Recognition in Electronic Documents", *Journal of the Korea Academia-Industrial,* Vol. 21, No. 5 pp. 451-457, 2020.
DOI: http://doi.org/10.5762/KAIS.2020.21.5.451

[5] CopyKiller, "Copy-killer used by 7 million people, what technology is applied?", muhayu, [cited Mar. 31, 2022], Available From: https://www.copykiller.com/notice/c/13951573/s/166625766?page=1 (accessed Apr. 4, 2022)

[6] K.-J. Woo, S.-H. Jung, "Comparison of Korean Morphology Analyzers According to the Types of Sentence", *Korea Information Science Association Korea Software Symposium*, Korean Society of Information Sciences, Korea, pp. 1388-1390, 2019.

[7] J.-S. Maeng, et al. "Implementation of A Plagiarism Detecting System with Sentence and Syntactic Word Similarities", KIPS Trans. Softw. and Data Eng, Vol.8, No.3, pp. 109-114, 2019.
DOI: http://doi.org/10.3745/KTSDE.2019.8.3.109

[8] J.-Y. Park, "Design and Implementation of the Similarity Detecting System using Structural Information of Documents", Master's thesis, Soongsil University of Computer Engineering, Korea, p34-42.

[9] K. Okubo, et al. "Binary Document Classification Based on Fast Flux Discriminant with Similarity Measure on Word Set", *Industrial Engineering & Management Systems*, Vol 18, No 2, pp. 245-251, Jun. 2019.
DOI: https://doi.org/10.7232/iems.2019.18.2.245

[10] B.-W. Ko, Young-Chul Kim, "A Similarity Valuating System using The Pattern Matching", *Journal of The Korea Society of Computer and Information*, 15(1), 70, pp. 185-192, Jan. 2010.
DOI: http://doi.org/10.9708/jksci.2010.15.1.185

[11] J.-S. Jeong et al. "Related Documents Classification System by Similarity between Documents", *The Korean Institute of Broadcast and Media Engineers*, JBE Vol. 24, No. 1, Jan. 2019.
DOI: http://doi.org/10.5909/JBE.2019.24.1.77

김 영 식(Yeong-Sik Kim) [준회원]

- 2021년 6월 : 아주대학교 금융공학과 (공학학사)
- 2021년 8월 ~ 현재 : 숭실대학교 정보과학대학원 정보보안학과 (석사과정)

〈관심분야〉
정보보안, 암호학, AI, 임베디드

백 종 경(Jong-Kyung Baek) [정회원]

- 2011년 2월 : 숭실대학교 정보과학대학원 정보보안학과 (공학석사)
- 2020년 2월 : 숭실대학교 대학원 컴퓨터학과 (공학박사)
- 2019년 8월 ~ 현재 : ㈜인피니솔루션 연구소 연구소장

〈관심분야〉
정보보안, 정보통신, 암호학, AI, Cloud

박 재 표(Jae-Pyo Park) [종신회원]

- 1998년 2월 : 숭실대학교 대학원 컴퓨터학과 (공학석사)
- 2002년 8월 : 숭실대학교 대학원 컴퓨터학과 (공학박사)
- 2010년 3월 ~ 현재 : 숭실대학교 정보과학대학원 교수

〈관심분야〉
정보보안, 보안평가 및 인증, 디지털포렌식, FinTech