공학석사 학위논문

효과적인 전자증거개시 처리를 위한 질의 추천 시스템

지도교수 신 상 욱

이 논문을 공학석사 학위논문으로 제출함.

2015년 2월

부 경 대 학 교 대 학 원

정보보호학 협동과정

이 헌 민

공학석사 학위논문

효과적인 전자증거개시 처리를 위한 질의 추천 시스템

지도교수 신 상 욱

이 논문을 공학석사 학위논문으로 제출함.

2015년 2월

부 경 대 학 교 대 학 원

정보보호학 협동과정

이 헌 민

이헌민의 공학석사 학위논문을 인준함.

2015년 2월 27일

주심　　이학박사 이 경 현　(인)

위원　　이학박사 신　　원　(인)

위원　　이학박사 신 상 욱　(인)

차 례

그림 차례

표 차례

A Query Recommending System for an Efficient
Evidence Search in e-Discovery

Heon-min Lee

Interdisciplinary Program of Information Security, The Graduate School,
Pukyong National University

Abstract

In order to effectively respond to the future litigation and to win a case, the most important task is to secure the highly relevant evidence produced by the proper search and review in respect of litigation issues during a whole e-Discovery procedure. At this point, the role of search is to reduce the amounts of document which should be additionally reviewed as potential evidence, so the poor search result caused by the use of wrong keywords can bring unexpected time and cost problems. These keywords, in general, are selected by analysis about the content of complaint or related documents at the beginning stage of e-Discovery and this stage is called ECA(Early Case Assessment) in EDRM(Electronic Discovery Reference Model). Ultimately, the success of e-Discovery depends very much on how well the participants performed essential tasks of ECA, but it has mostly depended on the ability of specific human like lawyer because existing e-Discovery solutions did not support this kind

of function in priority.

This thesis, therefore, suggests the machine learning based litigation preparing method for effective early case assessment on e-Discovery procedure. The suggested method extracts and collects the meaningful information from the complaints and related documents which were retained by the litigant. This information can be used for identifying the main issues of litigation, discussion in meet-and-confer session, creating a request for production, or writing another related complaint. Also, special experiment and evaluation result using the real complaint introduced by TREC Legal Track will be described for analyzing the availability of this method.

I. 서 론

1. 연구배경

1938년, 미국은 민사소송이 제기되어 판결이 끝날 때까지 발생하는 절차, 즉, 민사 소송 운용 절차를 제시하는 FRCP(Federal Rules of Civil Procedure)를 제정하며, 증거개시(Discovery)제도를 규정했다. 이 증거개시제도는 소송 당사자가 상대방이나 제 3자로부터 소송과 관련된 증거를 수집하기 위한 변론 전의 절차를 통칭하는 개념으로 주로 진술녹취서, 서면 질의에 대한 회답을 요구하는 질문서, 문서와 물건의 제출, 상대방에 대한 검사, 상대방의 사실 자백 요구와 같은 5가지 수단을 통해 이루어지게 된다.[20]

하지만, 최근 대부분의 업무가 워드프로세스, 데이터베이스, 전자메일, 휴대폰 등을 이용한 전자화된 문서(ESI)로 처리됨에 따라, 기존 증거개시제도의 운용에서 문서의 방대한 양, 변조 및 복제의 용이성과 같은 여러 가지 문제점들이 발생하게 되었고, 이러한 문제를 해결하고자 2006년 12월 1일 FRCP 개정안을 통해 증거개시의 대상을 종이문서로 제한했던 것을 확대하여 ESI를 포함하게 되었고, 전자증거개시(e-Discovery)가 등장하게 되었다.[5]

이처럼 2006년부터 미국에서는 전자증거개시 제도가 의무화되어 민사소송 당사자들이 전자증거자료를 120일 이내에 제출해야 한다. 이러한 절차가 적절하게 수행되지 않는다면, 브로드컴(Broadcom)사와의 소송에서 전자증거개시 절차를 잘못 처리하고 관련된 전자메일을 제시하지 못하여 $850만의 벌금 징수한 퀄컴사와 같은 결과를 초래

하게 될 것이다.[1]

이러한 전자증거개시의 중요성에도 불구하고, 소송 한 건을 수행하는데 평균 $150만이 소요되는 등 천문학적 비용과 방대한 기업 내부 전자문서에 대한 연관성 및 소송 관련 자료들을 120일 이내에 찾는 것은 결코 쉽지 않다. 따라서 기업들은 전자증거개시 솔루션을 구축하여 기업정보관리 및 문서관리 정책과 연계하거나, 디지털 포렌식 기술을 보유한 대형 법률사무소를 찾고 있다. 하지만, 기존의 처리 방식은 수많은 비용뿐만 아니라 법률 대리인에 대한 의존도가 매우 높기 때문에 회사의 내부 사정에 정통한 법률 대리인이 아니면 효율적인 처리가 진행되기 힘들 것이며, 이것은 곧 추가적인 시간적, 비용적인 부담이 발생함을 의미한다.[17]

이 논문은 소송 직후 소송 당사자에게 제공되는 소장을 분석하여, 소송에 관련된 쟁점 파악 및 소송과 관련된 질의를 추천하는 시스템을 제안, 구현 및 실험 한다. 이 시스템은 기계학습에 기반을 두고 있으며, 이를 위해 소장 분석으로 생성된 학습 집합을 이용해 기계학습하고, 학습 집합에 존재하지 않았던 추가적인 데이터 획득함으로서 해당 소송 관련 질의를 생성하고 사용자에게 소송과 관련된 질의를 추천한다. 또한 이 시스템의 평가를 위해 TREC Legal Track[2]에서 제공하는 다양한 소장과 EDRM Enron v1 Dataset[3]을 이용하여 시스템을 실험하고 그 결과를 평가 및 분석한다.

2. 연구 내용 및 구성

이 논문에서 제안하는 시스템의 목적은 소송 당사자가 소송에서 발생하게 되는 추가적인 부담을 줄이는 것이며, 이를 위해 소장 분석

을 기반으로 소송에 관련 쟁점 파악 및 질의 추천을 위한 시스템을 제안한다. 이를 위해 제안하는 시스템은 소송 발생 직후 소송 당사자에게 제공되는 소장을 분석하여 소송에 대한 핵심 쟁점을 파악하고, 이를 통해 소송 관련 주요 단락 선정, 대표 문장 선택 및 주요 용어를 추출한다. 또한, 소장 분석을 통해 추출된 용어를 이용하여 초기 질의를 생성하고, 이를 학습 집합으로 구성함으로서 소장과 직결되는 문서들을 획득한다. 하지만 이 과정으로 획득 된 문서 집합은 소장을 통해 드러난 정보만 이용하여 획득된 문서 집합이기 때문에, 이 집합만으로 실제 소송 처리에는 다소 부적합 할 수도 있다. 따라서 초기 질의에 해당되는 문서들과 유사한 문서들을 기계학습 기법을 이용하여 추가적으로 획득 및 분석하여 추가적인 소송 관련 정보를 획득한다.

이 논문의 구성은 2장에서 전자증거개시 절차, 일반적인 소장 구성 분석을 시작으로 문서 분석을 위한 다양한 기법들을 소개한다. 또한 이 논문의 초기 버전인 초기 질의 추천 시스템[16]을 관련 연구로서 소개한다. 3장에서는 초기 질의 시스템이 가진 한계점을 분석하고, 이를 개선하기 위한 질의 추천 시스템의 구조를 설계한다. 4장에서는 제안한 질의 추천 시스템에 대한 구현 및 실험 결과를 평가 하며, 마지막으로 5장에서 이 논문의 결론을 맺는다.

II. 관련 연구

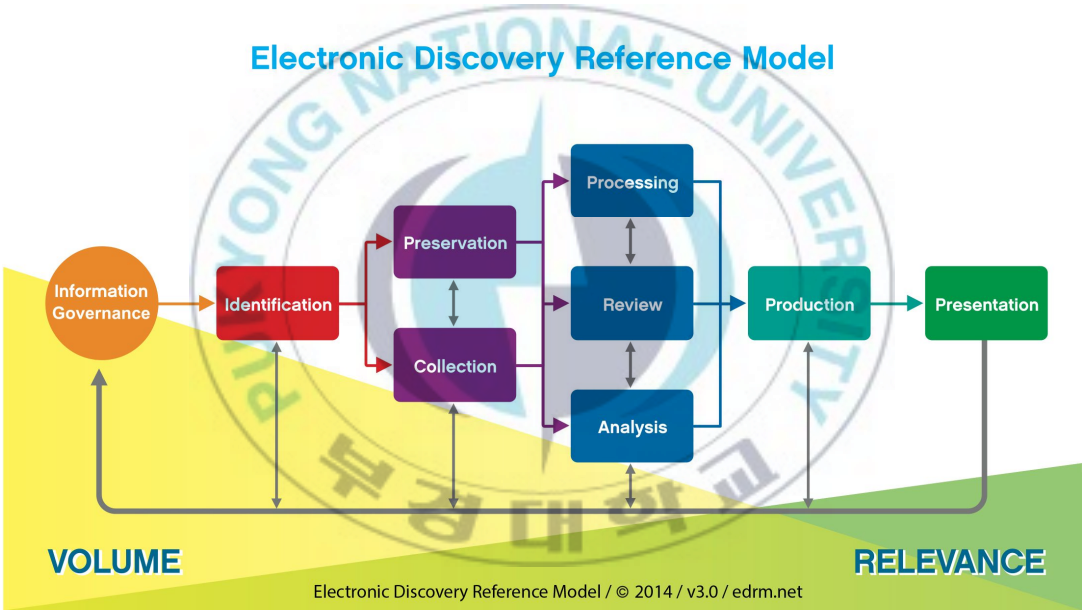
이 장에서는 전자증거개시(e-Discovery)의 정의 및 일반적인 처리 절차, 소장의 구조 분석을 시작으로 소장 분석을 위한 다양한 문서 처리 기법을 소개한다. 문서 처리를 위한 대표적인 기법에는 데이터 마이닝의 한 분야인 텍스트 마이닝, 문서의 주요 단락 추출 및 요약, 자연어 처리가 있다[26]. 또한, 보편적으로 전자증거개시에서는 엄청난 양의 문서들이 처리가 되기 때문에, 많은 양의 데이터를 효과적으로 검색, 분석, 처리하기 위해 정보 검색 기법, 기계 학습, 빅데이터 처리와 같은 구조를 소개와 정보 검색 시스템을 평가하기 위해 사용되는 몇 가지의 척도들을 소개한다. 마지막으로 비용 절감과 같은 효과적인 전자증거개시 처리와 법률 대리인에 의한 의존도를 줄이기 위해 제안한 초기 버전의 질의 추천 시스템을 소개한다.

1. 전자증거개시(Electronic Discovery)

2006년 12월 1일, FRCP의 개정으로 등장한 전자증거개시 제도는 기존의 증거개시제도에서 증거의 범위를 전자화된 문서(ESI)까지 확대하여 개정한 것으로, 이 제도에 따라 전자화된 문서들이 소송 개시일로부터 120일 내에 반드시 제출 되어야 한다. 이와 같은 전자증거개시 절차를 위해 보편적으로 EDRM[3]을 참조하며, 절차와는 다른 관점에서 전자증거개시의 효과적인 검색에 대한 연구가 진행되었던 대표적인 컨퍼런스로서 TREC Legal Track[2]이 있다.

가. EDRM(Electronic Discovery Reference Model)

EDRM은 전자증거개시의 절차 및 해당 절차에 대한 프레임워크 가이드를 작성하는 프로젝트로서, 프레임워크 뿐만 아니라 전자증거개시와 관련된 다양한 프로젝트 또한 계속 진행되고 있다. [그림 1]은 전자증거개시 절차를 개념적으로 나타낸 것이다.[4] EDRM에서 제공하는 이 참조 모델은 순환적인 구조를 가지며, 모든 과정이 순차적으로 반드시 진행 되어야 하는 구조는 아니다.



[그림 1] Electronic Discovery Reference Model 2014

각 단계별 프레임워크 가이드[4]를 요약하면 다음의 [표 1]과 같다.

나. TREC Legal Track

문서 검색 컨퍼런스(TREC)는 1992년 DARPA TIPSTER 프로

그램으로 시작되어 매년 다양한 분야의 정보검색(IR)에 대한 기법 및 평가 방법들을 연구 및 제시되고 있는 워크숍이다.[2] 특히 TREC의 다양한 Track 중 Legal Track은 효과적인 전자증거개시를 위한 검색 기술을 개발하기 위해 2006년을 시작으로 2012년까지 연구되었다.

[표 1] EDRM의 각 단계 별 수행 내용

단계	설명
정보관리	소송이 시작되기 전에 조직에서 정보 관리 정책에 의해 ESI가 관리하는 단계
식별	수집해야 할 ESI의 범위를 결정하고, ESI의 데이터 맵을 작성하는 단계
보존	식별된 ESI를 변경, 훼손되지 않게 보관하는 단계
수집	다양한 대상으로부터 ESI를 수집하는 단계
처리	중복되는 ESI를 제거하고 검토 분석을 위한 포맷으로 변환하는 단계
분석	수집된 ESI를 소송과 관련하여 평가하고, ESI 관련된 사건의 주제, 인물, 문서의 중요성을 작성하는 단계
검토	기밀성 데이터를 식별 및 전자증거개시 전략을 수립하기 위해 ESI를 검토하는 단계
산출	상호 식별 가능한 형태로 ESI를 산출하는 단계
제출/발표	법정에서 제출된 ESI들을 효과적으로 보이게 시각화 하는 단계

이 Legal Track은 전자증거개시의 절차를 제안하는 EDRM과는 달리 전자증거개시의 핵심 기능이라 할 수 있는 소송에 관련된 문서의 검색에 중점을 두고 연구되었으며, 초기에는 법률 전문가에 의한 소장에 대한 분석을 바탕으로, 증거 데이터 확보를 위해 이진 검색 기법(Boolean Search)을 시작으로 여러 연구가 진행 되었다. 특히 2010년부터 검색 분야의 최신 기술이라 할 수 있는 기계 학습(Machine

Learning) 기반의 자동화된 문서 검색 기능에 대한 성능평가 기법이 연구되었다.[5] Legal Track에서 진행 된 각 Task들은 몇 가지의 실제 발생한 소송 사례와 문서 집합을 대상으로 하여, 다양한 참가팀들에 의해 다양한 실험 결과를 제공하고 있으며, 이 결과를 이용하여 여러 정보검색을 위한 시스템들과 벤치마크 자료로서 활용될 수 있다.

다. 소장 구조 분석

민사 소송은 원고에 의해 피고의 혐의, 피고에게 원하는 것 등의 내용을 소장에 작성함으로서 시작된다. 또한 이 소장에는 원고가 피고에게 원하는 것, 피고의 혐의 등을 반드시 진술되어야 하며, 즉, 이는 발생한 소송에 대한 대부분의 증거는 소장에 명시 된다는 것과 소송에 활용 될 수 있는 키워드들은 법률 전문가의 검토에 의해 생성되는 것을 의미한다[6]. 또한, 소장은 소송이 제기 된 개요(Introduction), 사법권(Jurisdiction) 등과 같은 소송에 대한 각 단락들로 구성 되어 있으므로, 각 단락 별 분석을 통해 실제 소송에 대한 주요 키워드를 추출하는데 도움을 얻을 수 있다.

다음의 [표 2]는 코넬 법학 대학원(LII)과 TREC에서 제공하는 소장을 분석 한 것이다. [표 2]를 통해 알 수 있듯이, 실제 소송과 관련된 키워드들은 주로 General Allegations 단락에서 존재 한다는 사실을 알 수 있다. 하지만, General Allegations 단락에서만 소송과 관련된 키워드가 존재 하는 것은 아니며, 소송과 관련된 키워드가 존재하는 단락 및 해당 부분을 추가적인 분석이 추가적으로 병행 되어야 한다.

[표 2] 일반적인 소장의 단락 구성

단락 명		설명
LII	TREC	
N/A(Caption of Heading)		소송에 대한 개요
Preliminary Statement	Nature of the action	소송을 제기한 이유
Jurisdiction and Venue		해당 법원이 선택 된 이유
General Allegations	Plaintiff's Allegations	원고가 소송을 제기한 실제 주장 및 사실
	Substantive Allegations	
Count	Cause of Actions	피고의 행동이 위법인 이유 및 위법한 법령
Demand for relief		원고가 피고에게 바라는 점 및 선처

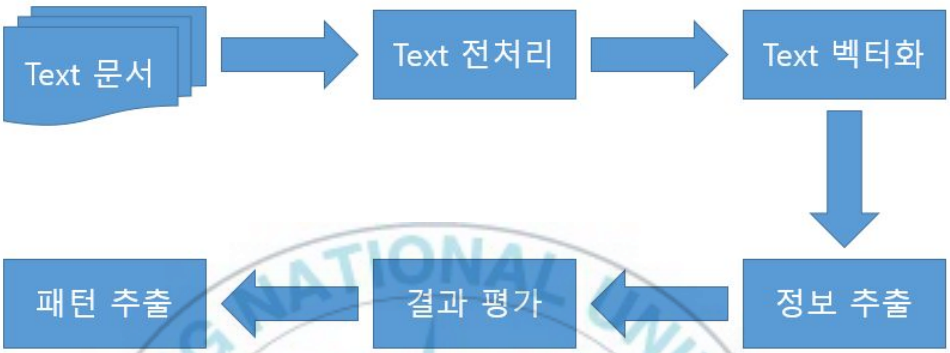
2. 문서 처리 기법(Text Processing)

문서 데이터는 일반적인 데이터와는 달리 비/반 정형화 되어있는 특징 때문에, 처리 성능을 높이기 위해 불용어 제거, 핵심 문장 및 단락의 추출, 자연어 처리 등의 전처리 과정을 통해 정형화가 된 이후, 정보 및 지식을 위한 분석이 진행 되어야 높은 정확률 및 신뢰성을 보장 할 수 있다. 따라서 이번 절에서는 소장 분석을 위해 텍스트 마이닝, 불용어 처리, 문서 요약/추출 기법, 자연어 처리 기법을 소개한다.

가. Text Mining과 불용어(stop word) 처리

텍스트 마이닝(Text Mining)은 비/반 정형화 된 문서를 자연어 처리 기술에 기반을 두어 유용한 정보를 추출 및 가공하는 기법인 데

이터 마이닝의 한 기법이다. 이 기법은 특히 최근 페이스북, 트위터와 같은 소셜 미디어 등 비정형 데이터의 증가로 인해 크게 주목받고 있으며 지속적인 연구가 진행 되고 있다. 일반적으로 처리 절차는 [그림 2]와 같은 과정을 거쳐 문서 데이터에서 정보 및 패턴을 추출한다.



[그림 2] 일반적인 텍스트 마이닝 처리 절차

또한 pdf, hwp, doc와 같은 다양한 문서 파일을 적절하게 처리하기 위해 전처리 과정을 필수적으로 거치게 되며, 이 과정을 통해 여러 문서 유형을 통합하는 과정, 불용어 제거, 단어의 분리와 같은 처리가 이루어진다. 따라서 전처리를 통해 비/반 정형화된 문서로부터 추출되는 정보의 정확성을 증가시키게 되며, 특히 전처리 단계에서 가장 중요하게 여겨지는 부분 중 하나가 불용어 제거 부분이다. 불용어 제거 과정은 문서 분석에 악 영향을 미치는 불용어를 제거함으로서 문서 분석을 용이하게 한다.

불용어(stop word)란 색인 및 검색에 의미가 없는 단어들을 일컫는 말로서, 의미를 갖는 어휘만을 색인 및 검색을 위해서 제외시켜야 하는 단어들을 의미한다. 이러한 불용어를 제거하게 되면 검색 속도 및 효율성, 색인 공간의 축소, 정보 추출의 정확성 등을 향상 시킬 수 있다[7]. 대표적인 영어의 불용어로 대명사 it, he, she 등과 관사인

the, a, an 등이 불용어에 해당한다. 이런 불용어들을 추출 및 제거하기 위하여 많은 기법들이 연구되고 있으며, 각 언어는 다양한 문법적인 차이가 있기 때문에 불용어 목록들은 서로 다르게 구성된다.

하지만 기존의 불용어 처리에 대한 연구들은 일반적인 문서에 국한 된다는 한계점이 있고, 소장이나 법률 관련 분야에 대한 불용어 목록에 대한 연구 및 목록은 존재하지 않고 있기 때문에, 소장에서만 자주 등장하게 되는 'complaint', 'plaintiff', 'allegation'과 같은 법률 관련 용어들은 일반 불용어로서 처리가 되지 않는 문제점이 발생한다. 이는 곧 소장 분석을 어렵게 하는 요인이 된다. 따라서 소장 분석을 위해 법률 관련 특수 불용어가 불용어 목록에 반드시 추가 및 제거 되어야 소장 분석을 더욱 더 용이하게 할 것이다.

나. 문서 요약 : 생성 요약 과 추출 요약

문서 요약(Text Summarization) 및 키워드 추출(Keyword Extraction)기법은 문서 처리의 가장 대표적 기법이다. 문서 요약 기법은 전체 문서를 요약하는 생성 요약(Abstractive Summarization)과 문서 내에서 중요한 단락, 문장을 추출하는 추출 요약(Extractive Summarization)으로 나뉜다.[8]

생성 요약은 문서를 이해하기 위해 분석 단계, 변형 단계, 통합 단계를 거치는 전문적인 요약이며, 이에 반해 추출 요약은 새로운 텍스트를 생성하는 일련 과정을 포함하지 않고, 주요한 문장, 단락만을 이용하는 요약 기법이다.[9] 생성 요약 및 추출 요약 모두 대부분 통계적 기법을 이용하여 전체 및 단락에 대한 가중치 계산을 통해 주요 단락 및 문장을 획득하고 요약하거나, 샘플 데이터를 이용하여 학습 집합을 구성한 후, 기계 학습 기법을 이용하여 주요 문장의 획득 및 요

약하게 된다. 국내의 경우 단락을 자동으로 구분 후, 단락을 자동으로 구분하여 문서를 요약하는 시스템[9]과 중요 문장을 추출하는 시스템[10]에 대한 연구되기도 하였다.

앞선 [표 2]와 같이 소장은 각 단락 별 특정한 양식을 갖고 있는 문서이다. 즉, 실제 증거와는 무관한 단락이 다수 존재한다. 이러한 의미를 갖지 않는 단락들은 소장의 통계적, 수치적 분석의 정확성을 저해하는 요인이 되기 때문에, 의미를 갖지 않는 단락들을 소장 분석의 전처리 단계에서 반드시 제거되어야만 분석의 신뢰성을 높일 수 있다.

다. 자연어 처리(Natural Language Processing)

정확한 텍스트 문서의 분석에 있어서, 자연어 처리는 필수적인 요구 사항이다. 자연어 처리란, 인간이 발화하는 언어 현상을 기계적으로 분석해서 컴퓨터가 이해할 수 있는 형태로 만드는 자연 언어 이해 혹은 유사한 형태를 다시 인간이 이해할 수 있는 언어로 표현하는 기술을 의미하며, 컴퓨터가 이해할 수 있는 형태로 표현하고자 한다는 점에서 인공지능과 비교되기도 한다[11]. 이 자연어 처리는 형태소(stemming) 분석, 품사 부착(POS tag), 구절 단위 분석 등에 사용된다. 이 자연어 처리가 연구되고 있는 프로젝트들로서, 자연어 처리에 위한 Java API를 제공하고 있는 Apache OpenNLP[12], Python API를 제공하고 있는 NLTK[13]등 여러 오픈소스 라이브러리를 통해 연구가 진행되고 있다. 특히, 이 Apache OpenNLP는 Max Entropy 학습 알고리즘을 이용하여, 여러 분야에 대한 학습 집합을 사전에 제공함으로서, 자연어 처리를 보다 쉽게 처리 가능하게 한다.

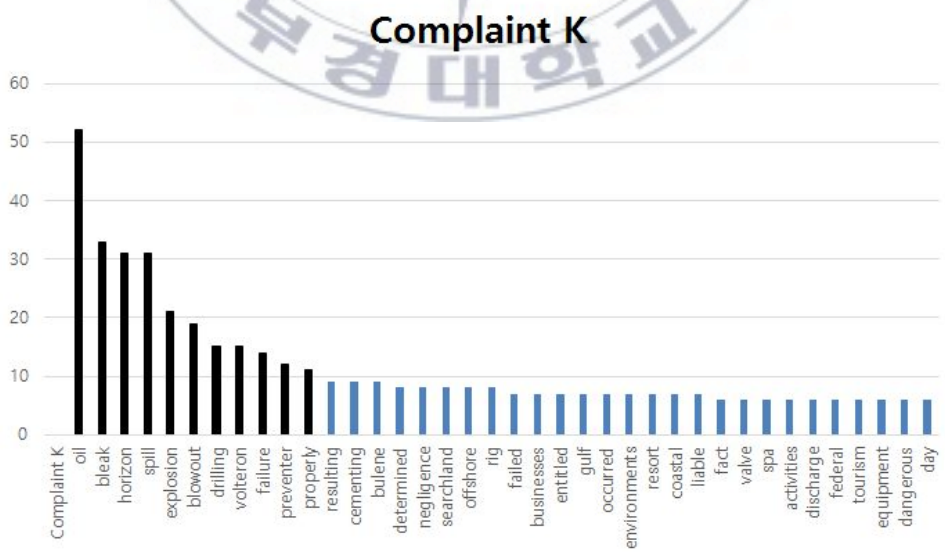
또한, 이 자연어 처리를 이용하여 거의 모두가 불용어로 취급 될 수 있는 대명사, 전치사 같은 특정 품사들을 품사 부착(POS Tagger)

을 이용하여 불용어를 제거한다. 또한 부가적으로 소장의 구절 단위 분석 및 주요 단락을 추출 등 여러 방면으로 사용 될 수 있다.

라. 문서의 용어 분포

앞에서 살펴 본 것과 같이 소장은 각 단락 별 개략적인 의미가 정해져 있는 반구조화 된 문서이며, 소송 당사자에 의한 사실 및 진술이 명시되는 문서이다.

또한, 일반적으로 주장 및 서술이 명시된 문서는 이에 해당하는 주요 단어가 반복적으로 출현하는 현상을 갖는다.[9][10] 즉, 소장에는 소송에 대한 진술 및 주장에 해당하는 주요 단어가 반복적으로 드러나는 특징을 갖게 되며, [그림 3]과 같이 지수 형태의 용어 빈도 분포를 갖는다. [그림 3]의 경우, 가장 빈번하게 등장한 몇 용어들을 이용하여 문서 전체의 의미를 개략적으로 이해 할 수 있으며, 나머지 일반 용어들은 문서의 구체적인 의미 이해를 돕는 역할을 한다.



[그림 3] 불용어가 제거된 Compliant K의 용어 빈도 분포

하지만, 용어의 빈도 분포에서 문서 전체 내용을 파악하는데 도움이 되는 주요 용어와 주요 용어를 서술하여 문서를 구체화 하는 일반 용어의 임계치(threshold) 결정은 아직 연구가 진행되지 않았으며, 또한 문서 별, 상황 별 경우에 따라 다르기 때문에 쉽게 정할 수 없는 한계점이 있다.

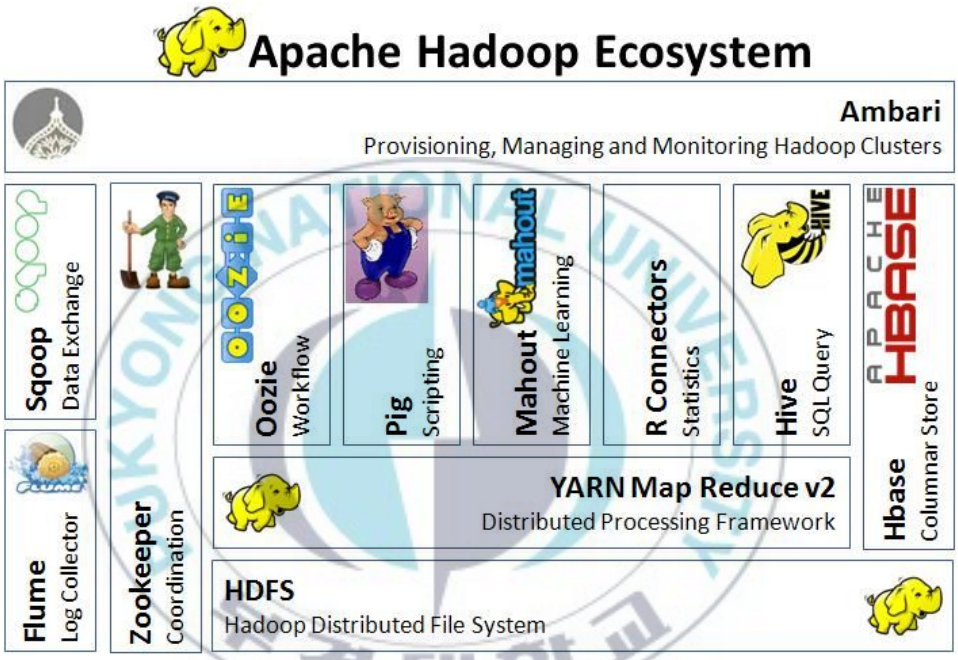
3. 대용량 문서와 빅 데이터

일반적으로 전자증거개시를 처리하기 위해선, 기업 내의 방대한 양의 문서를 처리 하는 것은 자명한 사실이다. 하지만, 일반적인 대용량 파일들과는 달리 보편적인 문서 파일은 작은 용량으로 구성되어 많은 양의 독립적인 파일로 존재한다. 이와 같이 구성된 문서 파일은 보편적인 처리 방식만으로는 시스템 메모리 부족, 많은 처리 시간과 같은 문제점이 발생한다. 따라서 이번 절에서 대용량 문서 집합을 처리를 위한 빅 데이터 기법과 이 플랫폼을 활용하여 효과적인 문서 색인과 검색을 가능하게 하는 검색 엔진 및 분석을 용이하게 돕는 기계 학습 기법을 소개한다.

가. 하둡 생태계(Hadoop Ecosystem)

아파치 하둡 (Apache High Availability Distributed Object Oriented Platform)은 대량의 자료를 처리 할 수 있는 큰 컴퓨터 클러스터에서 동작하는 분산 응용프로그램을 지원하는 프리웨어 자바 소프트웨어 프레임워크이다.[27] 이 플랫폼은 대용량 파일을 처리하기 위해 HDFS(Hadoop Distributed File System)이라 불리는 분산 확

장 파일 시스템을 사용하며, 빅 데이터 처리를 용이하게 하고 처리에서 발생 할 수 있는 여러 문제점을 최소화 한다. 또한 이 하둡에서 다양한 효과적인 처리를 위해 아파치에서는 하둡의 여러 하위 프로젝트와 같이 운용 및 연구되고 있다. 다음 [그림 4]는 하둡 하위 프로젝트의 대표적인 것들이며, 이를 보통 하둡 생태계(Hadoop Ecosystem)라고 한다.



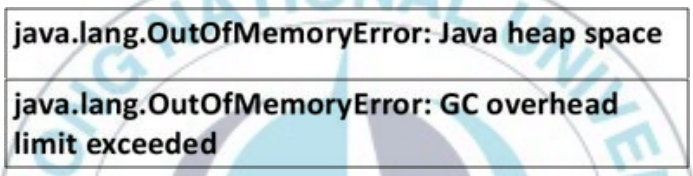
[그림 4] 하둡 생태계(Hadoop Ecosystem)

이 중 데이터의 특성을 추출하기 위한 기계 학습 기법(Mahout) 또한 많은 연구가 진행되고 있다. 보편적으로 기존의 기계 학습 연구들은 빅 데이터가 아닌 일반적인 데이터 처리를 위해 연구되었던 것들이나, 하둡 생태계에서 하둡과 머하웃을 함께 사용한다면 대량의 문서를 처리가 용이해지고 또한 많은 양의 문서를 처리해야 하는 전자증거 개시의 관점에서 적합하다.

나. 하둡과 빅 데이터 처리

하둡의 파일시스템인 HDFS는 주/종(master/slave) 구조를 가진다. HDFS 클러스터는 하나의 네임노드와 파일 시스템을 관리하고 클라이언트의 접근을 통제하는 마스터 서버로 구성된다. 또한, 클러스터의 각 노드에는 데이터 노드가 하나씩 존재하여, 실행될 때마다 노드에 추가되는 스토리지를 관리한다.

하지만 하둡의 동작을 위한 최적화 설정 없이 단순히 기본 설정만을 이용하여 운용하게 되면 [그림 5]과 같은 문제점이 발생한다.



[그림 5] 빅 데이터 처리 시 발생하는 에러

이러한 문제는 작은 크기의 많은 파일의 처리에 따라 발생하게 되는 문제로서, 하둡 상에 발생하는 문제점이 아니라 자바 상에 발생하는 문제점이다. 이러한 문제점은 각 노드들과 JVM의 최적화를 통해 충분히 해결 할 수 있다. 문제점을 해결하기 위한 대표적인 몇 가지 옵션은 다음 [표 3]과 같다.

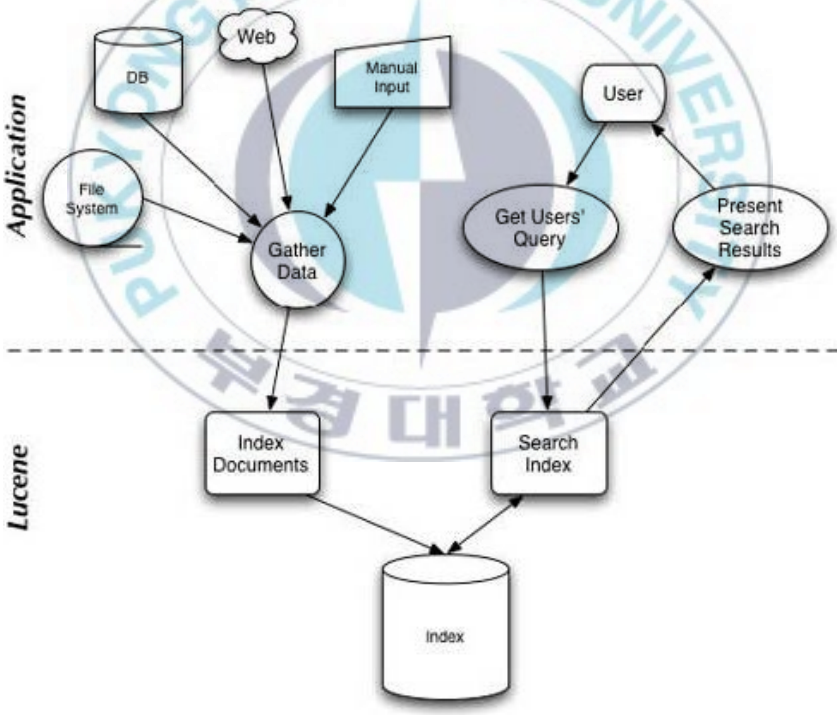
[표 3] 원활한 하둡 구동을 위한 대표적인 옵션

설정 파일	변경 옵션
hadoop-env.sh	HADOOP_HEAPSIZE
mapred-site.sh	mapred.tasktracker.map.tasks.maximum
	mapred.tasktracker.reduce.tasks.maximum
	mapred.child.java.opts
	mapred.job.reuse.jvm.tasks

이외에도 다양한 옵션들이 존재하며, 시스템 환경마다 최적화 된 옵션을 부여해야만, 원활한 처리가 가능하다.

다. 문서 색인과 검색 엔진

다양한 많은 문서의 검색을 위해 효과적인 검색 엔진은 필수적이다. 검색 엔진 개발을 위한 대표적인 오픈소스 라이브러리 중 하나인 루씬(Lucene)은 아파치의 한 프로젝트로서 색인 및 검색을 간단하고 효과적으로 구현 가능하게 한다.



[그림 6] 루씬(Lucene)의 전체 프레임워크[19]

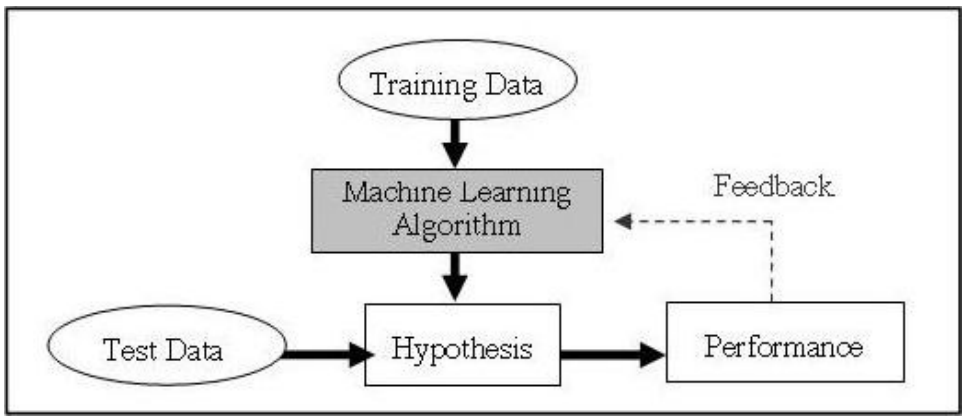
[그림 6]은 루씬이 동작하고 있는 전체 프레임워크를 보여주고 있다. 이 프레임워크는 다양한 문서들의 수집을 통해 역색인(Inverse Indexing)을 생성하고, 이렇게 생성된 색인을 이용하여 문서를 검색한

다. 또한, 이 루씬은 가장 간단한 이진 검색(Boolean Search)부터 고급 검색 방법인 유사 검색(Proximity Search)까지 다양한 질의(Query)를 가능하게 하며, 효과적인 질의 생성을 위해 정규 표현 또한 지원하고 있다.[22]

이 뿐만 아니라, 루씬은 역색인을 통해 색인 파일을 생성하기 때문에, 전체 문서 별 문서 빈도(Document Frequency), 단어 빈도(Term Frequency)등 분석을 위한 다양한 척도 또한 기본적으로 제공하기 때문에 문서 집합의 분석에 유용하게 사용 될 수 있다.

라. 특성 추출을 위한 기계 학습

기계 학습은 인공지능의 한 분야로서 한 분야로서, 샘플 데이터의 특성을 분석 및 추출하여 규칙(Rule)을 생성하고, 이를 이용하여 데이터들을 추천, 분류, 군집화 한다. 이러한 기계 학습은 높은 정확성을 위해 많은 연구가 진행되었으나, 이는 소규모 데이터를 위한 연구였다. 최근 들어 빅 데이터가 이슈화되며 여러 연구가 진행 되고 있으며, 아파치의 머하웃(Mahout)이 대표적인 한 프로젝트이다.



[그림 7] 기계 학습을 위한 일반적인 절차

[그림 7]은 기계 학습이 처리되는 일반적인 과정이며, 샘플 데이터(Training Data)를 기계 학습 알고리즘으로서 규칙(Hypothesis)를 생성한다. 이후, 이 규칙의 성능을 개선하기 위해 실험 데이터(Test Data)를 이용하여 규칙에 대한 성능을 평가하고, 이 결과를 반영하여 규칙을 강화한다. 대표적으로 사용되는 기계학습 알고리즘으로 지지벡터모델(SVM), 단순 베이지안 분류(NB Classification), K-means등이 있다.

4. 정보 검색 성능 평가 척도

정보 검색(Information Retrieval)에서 성능 평가를 위해 사용되는 대표적인 척도로서, 정확도(Precision), 재현율(Recall), F-척도가 있으며 이 척도들은 다음과 같이 계산된다.[15]

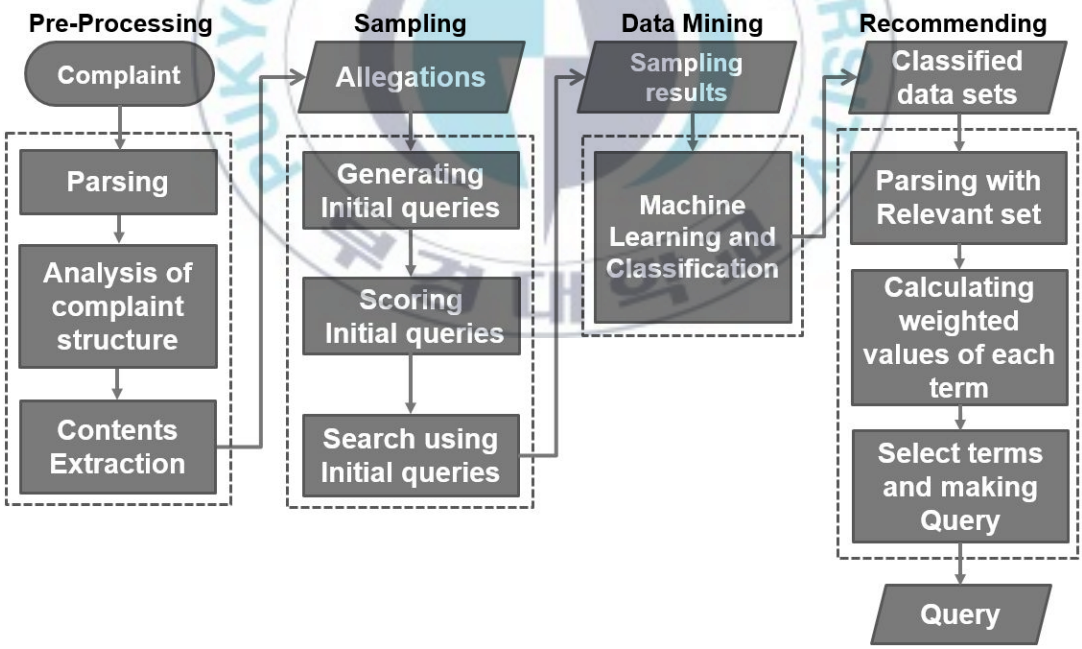
- 정확도$(Precision) = \frac{|\text{검색된 문헌 중 적합 문헌 중}|}{|\text{검색된 문헌}|}$
- 재현율$(Recall) = \frac{|\text{검색된 문헌 중 적합 문헌}|}{|\text{적합 문헌 전체}|}$
- $F1$ 척도 $= \frac{2(\text{정확도} \times \text{재현율})}{(\text{정확도} + \text{재현율})}$

보편적인 정보 검색 시스템에선 재현율에 비해, 검색 결과의 높은 정확도에 집중되지만 전자증거개시는 실질적인 증거 뿐 아니라, 잠재적으로 영향을 끼칠 수 있는 많은 증거를 찾는 것이 주목적이다.[17] 따라서 높은 정확도보다 높은 재현율이 우선시 되어야한다. 또한, 이 두 척도를 모두 고려한 F-척도 역시 성능 평가를 위한 한 척도가 될 수 있다.

5. 질의 추천 시스템

전자증거개시에 발생하는 비용을 줄이고 효과적인 처리를 위해 이 논문의 초기 버전인 질의 추천 시스템을 ICACT 2014[16]와 MIST 2014[17]를 통해 제안, 구현 및 실험 되었다.

초기 버전의 질의 추천 시스템은 소장의 처리를 위한 전처리 단계, 소장 분석을 통해 초기 질의 생성 및 학습 집합을 생성하는 샘플링 단계, 학습 집합을 이용해 기계 학습으로 관련 문서를 분류하는 데이터마이닝 단계, 분류된 관련 문서의 분석을 통해 초기 질의를 추천 질의로 확장하는 추천 단계의 4 단계로 구성되며, 세부적인 동작은 다음 [그림 8]과 같다.



[그림 8] 초기 질의 추천 시스템의 동작 방식

이 초기 시스템은 주어진 소장을 분석에 적합하게 파싱 및 구조를 분석하며, 보편적으로 소송 당사자에 의해 제기된 사실 및 주장들이

명시되는 General Allegation 단락의 각 문장들을 이용해 초기 질의를 생성한다. 이후, 이 결과를 기계학습을 위한 학습 자료로서 분류 규칙(Rule) 및 소송에 유사한 문서들을 분류하는데 사용한다. 이렇게 분류 된 유사 문서의 추가적인 분석으로 초기 질의를 확장하고 사용자에게 최종적으로 질의를 추천한다.

이 초기 버전의 질의 추천 시스템은 TREC Legal Track에 참가한 각 팀들에 비해 전반적으로 다소 낮거나 유사한 증거 검색 능력을 보였으나, 많은 부분을 자동화된 방법의 처리하며 및 법률 전문가의 개입 없이도 각 참가팀의 결과와 유사한 검색 성능을 보였다는 측면에서 의미를 부여 할 수 있었다.

III. 전자증거개시를 위한 질의 추천 시스템

이번 장에서는 초기 버전의 질의 추천 시스템이 가진 한계점에 대한 분석을 통해 한계점에 대한 대처 방안을 제안하고, 이를 적용한 개선된 질의 추천 시스템의 전체적인 프레임워크를 설계한다. 이 개선된 질의 추천 시스템은 전처리, 소장 분석, 학습 집합 생성, 기계 학습, 질의 확장의 5단계의 처리를 통해 소송 관련 질의를 추천한다.

1. 초기 질의 추천 시스템의 한계점 분석

TREC Legal Track에 참가한 각 팀에 비하여, 초기 버전의 질의 추천 시스템은 전반적으로 다소 낮은 검색 능력을 보였다. 하지만 법률 전문가의 개입 없이 반자동화 된 방법으로 처리를 통해 각 참가 팀과 유사한 검색 성능을 보였다는 측면에서 의미를 부여 할 수 있었다. 또한, 소장 분석으로 생성된 학습 집합을 기계학습을 통해 어느 정도 성능이 개선된다는 점 역시 의미 부여가 가능하였다.

이전 시스템의 성능 분석 결과, 불용어 처리 및 부수적인 문제점들이 있었으며 다음은 초기 버전의 질의 추천 시스템이 가졌던 한계점들을 나열 및 분석한 것이다. 또한 드러난 한계점들 대부분은 초기 질의를 생성의 성능을 저하시켜 전체적인 성능 저하를 시키는 원인이었음을 알 수 있다.

가. 소장 문서 파싱

소장의 문서는 pdf, doc, hwp, txt등과 같은 다양한 형태로 존재

할 수 있다. 따라서 초기 버전의 질의 추천 시스템은 이를 위해 문서 처리를 위한 오픈소스 라이브러리인 Apache Tika project[21]를 이용함으로서 다양한 문서의 일관된 처리가 가능하였다. 하지만, 각 데이터 형식에 따른 인코딩 방식 처리의 부재로 '◆' 와 같은 대치문자가 적용되어 전반적인 성능 저하의 한 원인이 되었다.

나. 불용어 처리

초기 시스템은 루씬[22]에서 제공하는 가장 기본적인 불용어 목록만을 이용한 처리를 하였다. 루씬에서 제공하는 기본적인 불용어 목록은 보편적인 영문서 처리에서 적합 할 수도 있다. 하지만 이 목록들은 가장 기본적인 불용어만 처리 할 뿐, 소장에서 빈번하게 사용되는 원고(plaintiff), 피고(defendant)와 같은 특수 법률 용어들이 불용어로서 처리가 되지 않기 때문에, 심각한 잡음(noise)으로 작용하여 시스템 성능을 크게 저하시키는 원인이 되었다. 또한, 루씬의 기본 불용어 목록으로는 용어를 기반으로 질의가 생성되는 초기 시스템의 적용에는 사용자 개입이 필요하다는 한계점이 발생하여, 완전 자동화의 한계점도 다소 존재하였다. 따라서 시스템의 성능 증대를 위해 소장 분석 및 질의 생성에 사용 되는 특수 불용어 목록의 생성 및 적용되어야 한다.

다. 주요 단락의 선정

기존의 시스템은 여러 증거들이 소장의 'Allegation' 단락에서 존재한다는 단순 가정으로 주요 단락이 선정되었다. 하지만, 실제 증거가 되는 많은 내용들은 'Allegation' 뿐만 아니라 'Count' 단락

에서 존재하는 경우가 존재 했으며, 특히 소송에 대한 쟁점이 다양한 경우 이러한 현상이 종종 발생하였다. 따라서 쟁점 파악 및 주요 단락의 선정을 위해 각 단락별 통계적 분석을 함으로써, 단락 선정의 신뢰성을 보장 할 필요가 있다.

라. 주요 단락의 선정

초기 시스템은 핵심 용어와 일반 용어를 구분하는 임계치를 계산하기 위해 단순히 주요 단락에서 가장 많이 등장한 단어 빈도의 제곱근을 임계치로 적용했다. 이 기법은 어떠한 의미를 갖기보다 대략적인 경험에 근거한 방법으로서, 임계치에 대한 신뢰도를 보장하기에 다소 모호함이 존재하였고, 핵심 용어와 일반 용어의 분리의 결과 편차가 다소 크게 나타났다. 따라서 핵심 용어와 일반 용어를 구분하기 위한 명확한 기법을 적용하여 임계치 선정의 신뢰성을 높여, 결과의 편차를 최소화 시켜야 한다.

마. 질의에 사용 될 대표 문장 선정

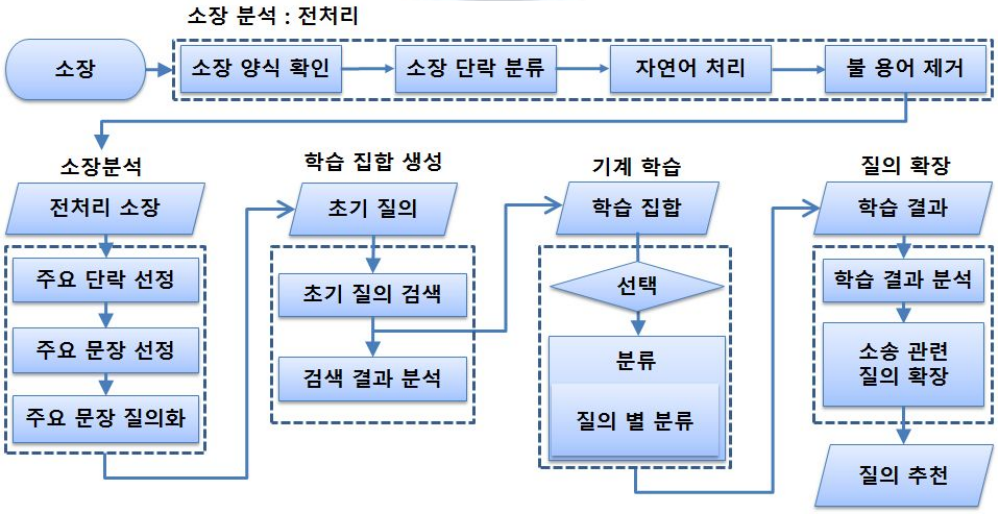
초기 시스템은 소장에 명시된 문장 번호를 기준으로 무작위로 질의를 생성하여, 시스템 전반적인 정확성을 떨어뜨리는 문제점이 발생했다. 또한, 소장의 한 문장 번호 별 문장은 독립된 하나의 문장이 아닌 여러 문장들이 결합되어 있는 형태이므로 개별적 문장 처리가 어려웠고, 많은 용어의 발생에 의한 잡음 문제가 야기되었다. 이러한 문제점들을 해결하기 위해 전체 단락을 문장 번호가 아닌 독립적인 문장으로 분리하고, 각 문장의 연관성을 분석하여 단락을 대표할 수 있는 문장이 질의에 사용 될 문장으로 선정 되어야 한다.

바. 기계 학습 처리 방식

기존 시스템의 기계 학습 방식은 전체 문서 집합을 초기 질의와 유사한 문서들을 분할(Partition)하는 방식으로 처리되었다. 이 방식은 많은 양의 문서를 한 번에 처리 가능한 장점은 있으나, 중복된 문서가 포함 될 수 없는 분할을 특징을 사용하는 만큼 개별 질의 별 분류(classification)에 대한 높은 신뢰성을 보장 받기는 힘들다. 따라서 기존의 시스템에서 사용된 방식과 달리 개별 질의 별 분류를 이용하여 질의를 확장하여 추천한다.

2. 개선된 질의 추천 시스템 : 전체 프레임워크

이 논문에서 제안하는 질의 추천 시스템은 초기 시스템의 프레임워크와 의미적으로 유사하며, 초기 시스템의 전처리 단계를 두 단계로 세분화하여 소장 분석에 대한 정확성을 향상 시킨다. 또한, 기계 학습은 여러 질의를 이용한 전체 문서의 분할이 아닌 각 질의 별 관련 문서의 분류를 적용하여 처리한다.



[그림 9] 개선된 질의 추천 시스템의 전체 프레임워크

[그림 9]는 본 논문에서 제안하는 시스템의 전체 프레임워크이며, 이 시스템의 목적은 초기 시스템과 동일하게 소장 분석을 통해 해당 소송과 관련된 쟁점을 파악 및 소송 당사자에게 질의를 추천하는 것이다. 이 시스템은 총 5단계로 구성되어 있으며, 이전 시스템에서 드러난 한계점을 개선하기 위해 단순 전처리(Preprocessing)과정을 문서 처리를 위한 전처리 과정과 소장 분석의 2단계로 확장한다. 전처리와 소장 분석의 두 단계를 거치며 단락을 대표하는 문장 채택 및 초기 질의를 생성하며, 이 초기질의를 이용하여 전체 문서 집합 중 소장과 직접적으로 관련된 문서들을 학습 집합으로 구성 및 기계 학습을 적용한다. 이후, 초기질의의 결과인 학습 집합과 기계 학습 결과를 분석하여 획득한 추가적인 정보를 초기질의에 적용 및 확장하여 최종 결과를 사용자에게 추천하게 된다.

3. 개선된 질의 추천 시스템 : 단계 별 처리과정

가. 소장 분석 : 전처리

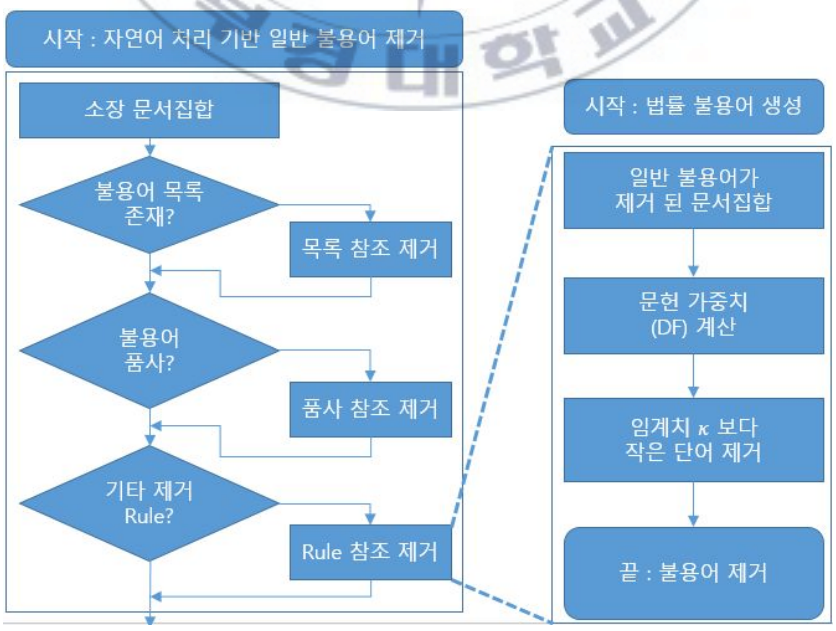
소장은 pdf, doc, txt와 같은 다양한 문서 형태로 제공되며, 각 파일이 제공되는 어떠한 형식도 정형화 되어 있지 않기 때문에, 문서 파일을 처리하기 위한 선행적인 처리가 필요하며, 이는 곧 높은 성능을 위해 필수적으로 거쳐야 하는 단계이다. 또한, 소장 역시 문서이기 때문에 텍스트 마이닝에서 사용하는 전처리가 적용되어야 분석의 신뢰성을 높일 수 있다.

▪ 소장 양식 확인 및 단락 분류

이전 시스템과 동일하게 소장들의 일괄된 처리를 위해 Apache Tika project에서 제공하는 API를 사용함으로써, 여러 형식들의 문서를 처리한다. 또한, 시스템의 문서 입/출력의 인코딩 형식을 설정하여 대치문자 발생을 예방한다. 이 시스템에서 사용하는 인코딩 형식은 모든 유니코드를 표현 할 수 있는 UTF-8을 사용한다. 마지막으로 소장은 [표 2]와 같이 특정 양식에 따라 여러 단락으로 구성되기 때문에, 정확한 분석을 위해 소장 단락이 선행적으로 분리 한다.

▪ 자연어 처리와 불용어 제거

이전 시스템 한계점 분석을 통해 불용어 처리의 중요성과 전체 시스템에 미치는 영향과 루씬에서 제공하는 기본 불용어 목록만 이용하는 것은 부적절 하다는 것을 알 수 있었다. 따라서 분석 대상이 소장이란 특수성을 반영하여 법률 관련 불용어가 반드시 제거되어야 한다.



[그림 10] 일반 및 법률 불용어 제거 과정

[그림 10]은 법률 관련 불용어 제거를 위한 프레임워크이며, 이와 같은 과정을 거치며 일반 불용어는 물론 법률 관련 불용어를 제거 할 수 있을 뿐만 아니라, 처리의 효율성 또한 증가 시키게 된다. [그림 10]에서 사용되는 불용어 목록이란 사전에 정의되어지는 목록[18]을 말하며, 이 목록을 사용함으로서 처리를 간소화 할 수 있다. 또한, 자연어 처리의 품사 부착(POS Tagger)을 이용하여 접속사, 관사, 전치사 같은 불용어 품사들을 제거한다. 이 두 과정을 거치며 대부분의 불용어들은 제거되지만, 구두점, 특수 문자와 같은 잡음(noise)들을 기타 제거 규칙을 참조하여 추가적으로 제거한다.

제안하는 시스템의 경우, 소장의 특수성에 해당하는 용어들이 대다수를 차지하기 때문에, 소장 문서 집합을 추가적으로 구성하여 법률 관련 불용어를 제거한다. 보편적으로 불용어 처리를 위한 가중치 계산 방식은 'TF-iDF' 를 사용하지만, 소장는 원고(Plaintiff), 피고(Defendant)와 같은 소송에서만 주로 사용되는 단어들이 높은 문서 빈도 뿐 아니라, 높은 용어 빈도 (TF:Term Frequency)를 갖는 특징 때문에 기존의 'TF-iDF' 를 적용하기 힘들며, 따라서 소장 문서 집합에 불용어 처리를 위해 문서 빈도(DF:Document Frequency)가중치를 계산하여, 70%를 임계치로 채택하여 임계치 이상의 문서에서 등장한 단어들을 불용어로서 제거한다. 이 과정을 통해 제거되는 대표적인 용어들로서 plaintiff, defendant, allege, trec, legal, track등이 여기에 해당한다.

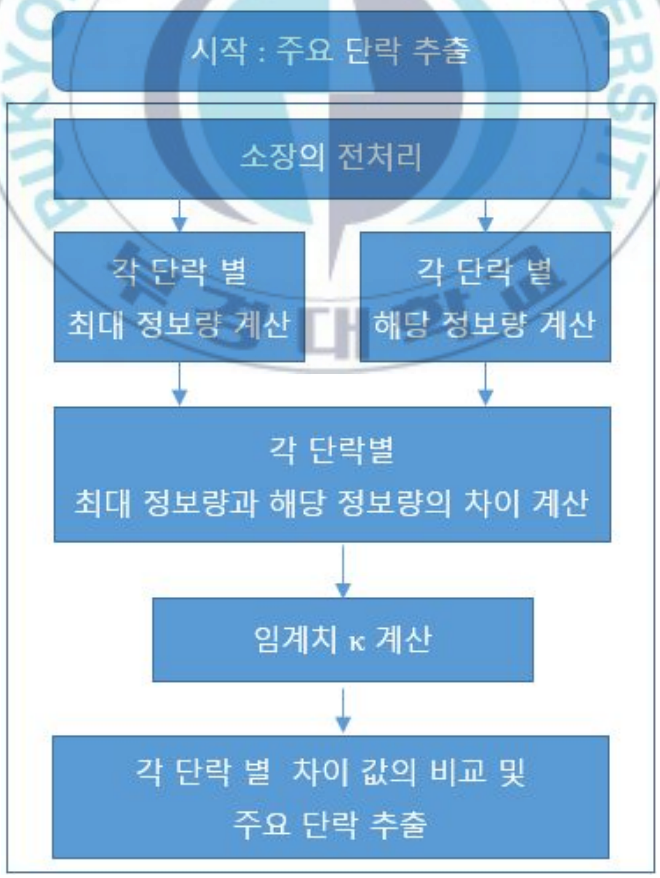
나. 소장 분석

소장 분석은 질의 추천을 위해 소장이 분석되는 단계로서, 주요 단락의 추출, 대표 문장 선정 및 이를 이용한 질의를 생성하는 단계

이다. 이 단계를 통해 이전 시스템에서 분석된 한계점들이 개선되며, 신뢰성을 높이는데 중요한 역할을 한다.

▪ 주요 단락 선정

소장은 각 단락에 따라 내용들과 용어들이 대략적으로 결정되는 특징과 특히 소장에선 어떠한 혐의에 대하여 주장이 반드시 명시되어야 하므로 특정 단어의 반복적 출현과 뒷받침되는 부연 설명을 위한 단어들이 등장하는 특징 때문에, 불용어 처리와는 상관없이 많은 중복 단어가 발생하게 된다. 따라서 각 단락의 용어에 대한 통계적 분포와 제공하는 정보량을 이용한다면, 소송과 관련된 주요 단락의 선택 가능하다.



[그림 11] 소장의 주요 단락 추출 과정

[그림 11]은 주요 단락을 결정하기 위한 처리 절차이며, 이를 위해 소장의 전처리 후, 각 단락에 대한 최대 정보량의 계산 및 실 분포에 대한 정보량을 계산을 하며, 최대 정보량 및 해당 정보량은 다음의 식과 같이 계산한다.

$$\text{단락의 정보량} = -\sum_{i=1}^{cnt} p(w_i)\log p(w_i)$$

이때, $p(w_i)$는 단락의 단어 분포 중 i번째 단어 빈도이며, cnt는 분포에 존재하는 단어들의 수이다.

$$\text{단락의 최대 정보량} = -M\sum_{i=1}^{cnt}\frac{1}{M}\log\frac{1}{M} = -\sum_{i=1}^{cnt}\log\frac{1}{M}$$

이때, M은 단락의 단어 분포의 빈도에 대한 평균이며, cnt는 분포에 존재하는 단어들의 수이다.

해당 정보량이 최대 정보량에 가까워질수록, 단어 분포가 고르게 분포되어 있다는 것이며, 해당 단락이 소송에 대한 혐의를 주장하는 부분이 아닐 가능성이 높음을 의미한다. 따라서 해당 정보량과 최대 정보량의 차이가 최대가 되는 단락을 핵심 단락으로 선택한다.

$$\text{정보량 } \Delta = \text{단락의 최대 정보량} - \text{단락의 해당 정보량}$$

또한, 소송과 관련된 내용이 한 단락이 아닌 여러 단락에 걸쳐 주장 되는 경우도 존재하기 때문에, 최대 정보량을 갖는 단락 이외의 나머지 단락에 대한 분석도 필요하다. 이를 위해 임계치 κ를 설정하여, 임계치보다 높은 단락들을 후보 단락으로 선정한다.

$$\text{후보 단락 } CP = \{P_i | \Delta_{P_i} \geq \Delta_C\}$$

이때 P_i는 각 단락들을 의미하며, C 는 소장을 의미한다.

소장 전체에 대한 정보량 Δ보다 높은 단락을 후보 단락의 평균을 임계치 κ로 하고, 임계치 κ보다 높은 단락을 주요 단락으로 선정한다.

$$\text{주요 단락 } MP = \{CP_i | CP_i \geq \kappa = avg(CP_i)\}$$

- 대표 문장 선정

이전 시스템은 수많은 질의들이 생성되고, 독립된 하나의 문장이 아닌 여러 문장들이 결합되어 있는 형태를 사용해 전반적인 성능이 좋지 못하였다. 따라서 이 단계에서 이전 시스템에서 발생한 한계점 개선을 위해 해당 단락을 대변할 수 있는 대표 문장들을 선택한다. 대표 문장 선택을 위한 처리 과정은 [그림 12]와 같다.

[그림 12] 단락의 대표 문장 선정 과정

주요 단락이 선정 후, 대표 문장을 선정하기 위해 자연어 처리의 문장 검출(Sentence Detection)기법을 이용해 각 문장을 분리한다. 분리된 각 문장들을 이용하여 각 문장들의 관계 및 유사성을 다음의 식과 같이 계산한다.

$$\text{문장 별 유사성 } CS_{i,j} = \Pr(s_i|s_j)\Pr(s_j|s_i)$$

$CS_{i,j}$가 1에 가까울수록 두 문장 s_i, s_j는 완전 일치함을 의미하게 되고, 0에 가까울수록, 두 문장 s_i, s_j 일치하지 않음을 의미한다. 따라서 어떤 i번째 문장 s_i가 단락 내에서의 영향력은 s_i의 모든 유사성의 합으로 생각 할 수 있다. 하지만, 편중된 값에 의해 높은 영향력을 갖는 상황을 방지하기 위해, 동일한 영향력을 갖는 경우 문장 s_i에서 분산이 낮은 경우를 우선적으로 주요 문장으로 선택한다.

이와 같은 방식으로 선택되는 주요 문장은 해당 단락에서 출현하는 단어들이 많이 나타나는 것을 의미하고, 각 단락을 대표 할 수 있는 문장으로서 사용될 수 있다. 이전 시스템과는 달리, 주요 문장 수가 사용자에 의해 결정되어 질의 자체 수를 조절 할 수 있는 장점과 모든 문장을 질의로 만들지 않고, 문장들의 유사성에 따른 우선순위에 따라 정해지기 때문에, 이전 시스템에 비해 높은 정확률을 기대 할 수 있다.

- 대표 문장 질의화

이 단계는 소장 분석을 통해 선택된 대표 문장들을 이용하여 질의를 생성한다. 이전 시스템에서 사용된 질의 생성 규칙과 크게 다르진 않지만, 핵심 용어와 일반 용어를 분류하기 위한 임계치 결정이 해당 단락의 제곱근을 이용했던 방식에서, 해당 단락에서 출현한 단어들 중

높은 빈도로 출현한 용어 20%를 핵심용어, 나머지 80%를 일반용어로 분류 한다. 이와 같은 임계치 설정은 상위 20%에 의해 하위 80%가 결정된다는 파레토 법칙[23]에 기반을 두고 있으며, [그림 3]의 검정에 해당하는 부분이 상위 20%이다.

선정된 대표 문장을 이용한 초기 질의 생성은 다음의 [표 4]에 따라 초기 질의를 생성한다.

[표 4] 초기 질의 생성 규칙

규칙	규칙 상세
#1	초기 질의는 재결합 된 문장에서 등장한 단어로 구성된다.
#2	질의 수는 재결합 된 문장 수와 같다.
#3	초기 질의는 *A*와 *O*의 두 영역으로 구성되며, 두 영역은 AND 연산으로 조합된다.
#4	*A* 영역에 포함되는 단어는 선택된 주요 문장의 용어 중 해당 단락의 20% 내에 있는 용어들의 모임이다. 이 영역에 해당하는 단어들은 AND 연산으로 조합된다.
#5	*O* 영역에 포함되는 단어는 선택된 주요 문장의 용어 중 해당 단락의 20% 내에 없는 용어들의 모임이다. 이 영역에 해당하는 단어들은 OR 연산으로 조합된다.

다. 학습 집합의 생성

이 단계는 소장 분석을 통해 생성된 초기 질의를 이용한 검색 결과와 초기질의와 전혀 관련 없는 Dummy집합을 학습 집합으로 구성하는 단계로서, 이 단계를 통해 생성된 데이터들은 다음 단계의 기계학습을 위한 입력 데이터로 사용된다. 또한 이 단계를 통해 학습 집합으로 구성된 문서들을 추가적으로 분석함으로서, 기계학습의 결과 데이터와 비교하여 추가적인 정보를 획득하는데 사용한다.

라. 기계 학습

이 단계는 이전 단계에서 생성된 학습 집합을 이용하여 기계 학습 및 유사 문서를 찾는 단계이다. 또한 이전 시스템과는 달리 사용자에 의해 개별적인 질의에 대하여 문서들이 분류된다.

- 초기 질의 의한 유사 문서 분류

선정된 대표 문장에 대한 초기 질의를 이용하여 유사 문서를 분류하는 방식으로, 일반적인 기계학습 처리방식과 유사하다. 하지만 일반적인 분류 기법은 학습 집합에 대한 목적 변수가 정해져 있어 효과적인 분류가 가능한 반면, 질의 추천 시스템은 소송과 관련된 집합만 획득 가능하기 때문에, 추가적인 Dummy 집합을 추가적으로 구성해야 한다. 이 Dummy 집합은 초기 질의에 반대되는 질의를 이용하여 구성할 수 있으며, 학습 집합과 동일한 크기의 집합으로 준비한다.

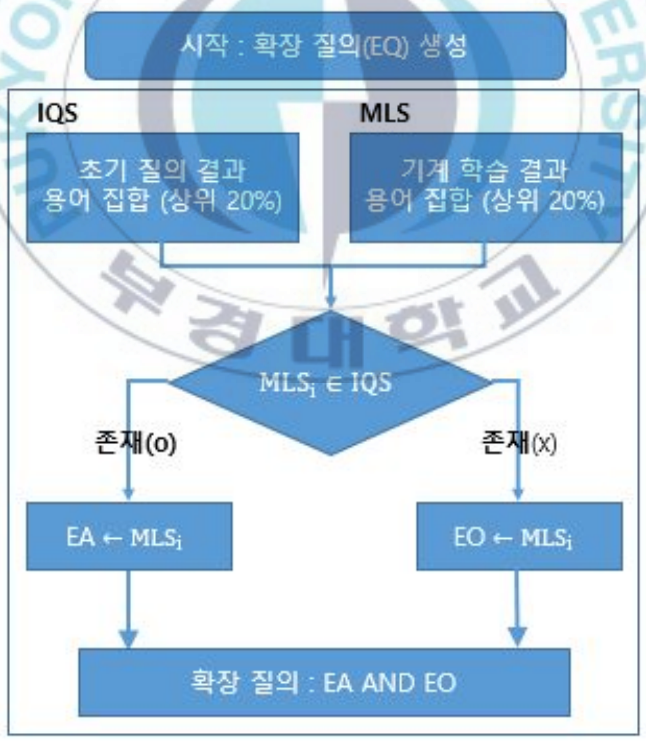
[그림 13] 기계 학습 처리 절차

마. 질의 확장

질의 확장 단계는 학습결과 분석과 소송관련 질의 확장의 2 단계로 구성되며, 기계학습 단계를 통해 분류된 학습 결과를 분석 후 초기 질의를 확장 및 사용자에게 질의를 추천하게 되는 시스템의 마지막 단계이다.

- 학습 결과 분석 및 질의 확장

이 단계는 초기 질의 결과에서 획득 할 수 없었던 추가적인 정보를 획득하여 초기 질의에 적용 및 확장하는 시스템의 마지막 단계 단계이다.



[그림 14] 확장 질의(EQ) 생성 절차

초기 질의를 이용해 구성한 용어 집합(IQS)은 소장에 존재하는

용어들과 직접적으로 관련된 문서들로서 구성된 용어 집합이다. 또한, 기계 학습 결과를 통해 구성된 용어 집합(MLS)는 IQS와 간접적으로 관련이 있다고 할 수 있다. 따라서, MLS에 존재한 용어 중 IQS에 존재하는 용어를 핵심 용어로서, 나머지 용어들은 핵심 용어의 서술을 위한 일반 용어로서 분류하여 확장 질의를 생성한다.

[표 5] 추천 질의 확장 규칙

규칙	규칙 상세
#1	추천 질의는 초기 질의를 포함하고 있으며, 초기 질의와 추천 질의를 갖는다.
#2	추천 질의 수는 초기 질의 수와 같다
#3	확장 질의 EQ는 '[IQ] OR [RO]' 구조를 갖는다. 여기서 IQ는 초기질의, RO는 추가 단어들이 OR연산으로 묶인 형태로 구성된다.
#4	EQ의 임계치는 초기 질의 생성 규칙과 동일하게 상위 20% 만을 데이터를 이용한다.

확장 질의(EQ)를 생성하기 위해, 초기 질의를 이용해 획득한 학습 집합의 전체 단어 빈도 중 상위 빈도 20%만을 추출한다. 전체의 단어가 아닌 20%만을 사용하는 이유는 문서들에서 출현한 많은 양의 용어들을 모두 사용하는 것은 현실적으로 무리일 뿐만 아니라, 오히려 시스템의 성능에 악영향을 끼칠 수 있기 때문이다. 따라서 용어들의 상위 20%만을 사용함으로써, 핵심적인 내용의 비교 및 확장 할 수 있게 된다. 이 과정을 거치며 생성된 확장 질의는 다음의 [표 5]에 따라 초기 질의를 확장한다.

IV. 질의 추천 시스템 구현 및 실험 결과

이번 장에서는 3장에서 제안한 기법들을 이용하여 질의 추천 시스템을 구현한다. 이 시스템은 크게 문서 분석을 위한 모듈, 문서 검색을 위한 모듈, 기계학습을 위한 모듈로 총 3가지의 모듈로 구성되어 질의를 추천하게 된다. 또한 이번 장에서는 TREC Legal Track에서 제공하는 한 소장과 EDRM 문서 집합을 이용하여 실험함으로써, 추천된 질의에 대한 실험 결과를 초기 시스템의 성능 결과[17]와 벤치마킹하고 그 결과를 분석한다.

1. 질의 추천 시스템 : 구현 환경 및 구성 모듈

이번 절에서는 시스템 구현에 사용된 개발 환경 및 오픈소스 라이브러리를 소개하고, 시스템을 위해 구현된 모듈 및 동작 방식을 소개한다.

가. 개발 환경

질의 추천 시스템을 개발 및 실험에 사용된 환경은 다음과 같다.

[표 6] 구현 및 실험 개발 환경

H/W	CPU : Intel(R) core(TM) i5-2500 CPU @3.30HZ Memory : 16 GB
OS	Linux CentOS 6.5(Final) 64bit
IDE	Eclipse 4.4.1 Luna
언어	Oracle JDK 1.7.0_67

나. 오픈소스 라이브러리

시스템을 구현하기 위해 아파치의 다양한 프로젝트들이 사용되었다. 이전 시스템과 동일하게 문서 처리를 위해 Tika, 검색 시스템을 위해 루씬, 기계학습 및 빅 데이터 처리를 위해 하둡[24], 머하웃[25]을 사용했으며, 추가적으로 효과적인 소장 분석을 위해 OpenNLP가 사용되었다.

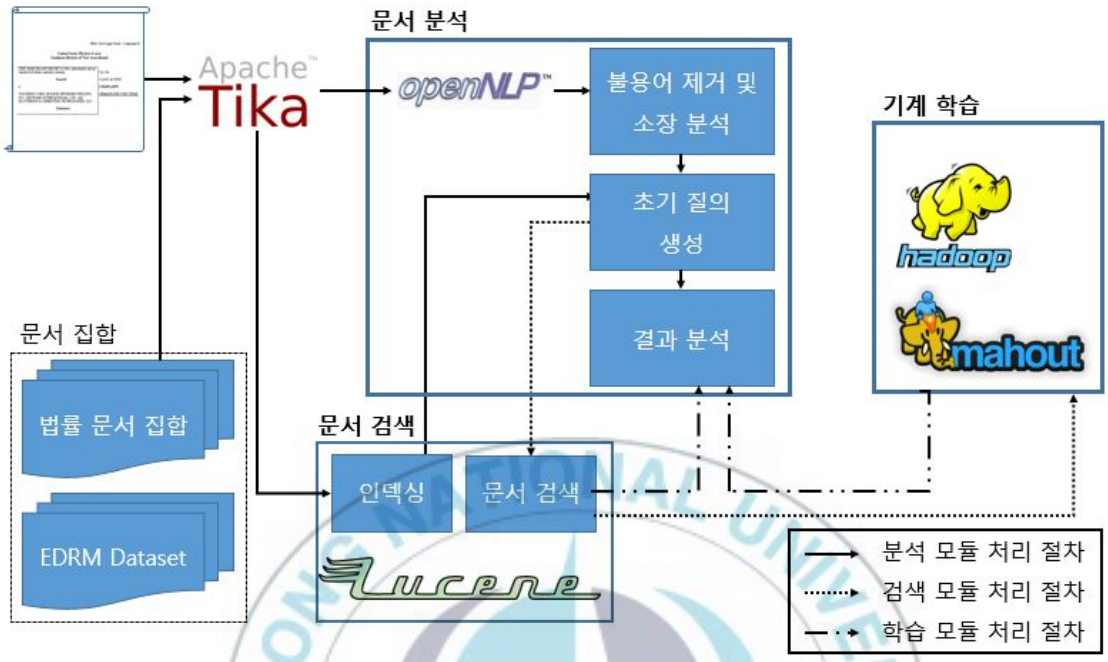
[표 7] 구현에 사용된 오픈소스 라이브러리

오픈소스 라이브러리		
Apache	Tika 1.6	- 다양한 문서 형식의 소장을 처리
	Lucene 4.10	- 문서의 유사도 및 가중치 계산 - 초기/ 추천 질의를 이용한 문서 검색
	Hadoop 1.2.1	- 많은 양의 문서들의 효과적인 처리
	Mahout 0.9	- 학습 집합을 이용한 기계학습
	OpenNLP 1.5.3	- 비정규화 된 소장 문서를 분석

다. 질의 추천을 위한 모듈

질의 추천 시스템을 구성하는 모듈은 문서 분석을 위한 모듈, 문서 검색을 위한 모듈, 기계학습을 위한 모듈로 총 3개의 모듈로 구성되어 질의를 사용자에게 추천하게 된다. 또한 대부분의 모듈은 이전 시스템에서 사용된 모듈 구성과 유사하게 진행되며, 소장 분석을 위해 새롭게 제안된 기법들 역시 이 3개의 모듈 내에서 모두 처리되며, 전

체적인 흐름은 다음 [그림15]과 같이 구성된다.



[그림 15] 질의 추천 시스템의 각 모듈 별 처리 과정

▪ 분석 모듈

이전 시스템의 한계점을 개선하고, 성능을 향상시키기 위한 대부분의 기능들이 대부분 분석 모듈 내에서 처리된다. 이 모듈 구현을 위해 Tika, 루씬, OpenNLP 라이브러리가 사용되며, 문서의 효과적인 처리, 각종 가중치 계산과 소장 분석을 통해 학습 집합을 생성하기 위한 초기 질의를 생성하는 역할을 한다. 초기 질의의 생성은 전체적인 성능에 가장 큰 영향을 미치는 요인으로서 이 과정에 많은 오픈소스가 사용되며, 많은 절차가 필요한 이유이기도 하다. 또한, 검색 모듈을 이용해 처리된 학습 집합과 기계학습 모듈에 의해 처리된 유사문서집합을 분석하여 확장 질의(EQ)를 생성한다.

▪ 검색 모듈

이전 시스템과 동일하게 질의에 해당하는 문서를 검색하는 역할을 한다. 검색 모듈은 문서의 효과적인 검색을 위한 색인 기능 및 검색 기능으로 구성된다. 이 모듈을 통해 소장과 가장 연관된 문서들이 검색되며, 기계학습 처리를 위한 학습 집합 및 Dummy 집합을 구성하게 된다. 또한, Dummy 집합은 초기 질의에 반대되는 질의 결과들 중 질의와 문서의 유사성이 높은 순서에 의해 학습 집합의 문서 수만큼 구성되며, 이때 유사성은 루씬에서 제공하는 유사도 함수를 이용하여 계산된다.

▪ 기계학습 모듈

전자증거개시를 처리하기 위해 많은 양의 비정형화된 문서들의 원활한 처리를 위해 하둡과 같은 빅 데이터 처리가 병행되어야 한다. 또한, 기계학습을 위한 scikit-learn, PyML과 같은 다양한 오픈소스 라이브러리들이 존재하지만, 이들 라이브러리는 대용량의 문서들을 처리하기에 다소 어려움이 존재한다. 따라서 많은 문서의 기계 학습 처리를 위해 설계된 머하웃 오픈소스 라이브러리를 이용하는 것이 적합하다.

이 모듈은 이전 시스템의 처리 방식과 동일하게, 검색 모듈로 생성된 학습 집합을 HDFS(HaDoop File System)에 적재하고, 시퀀스 파일 변환 및 기계 학습시킨다. 이 결과로 유사 문서를 분류하기 위한 모델이 생성되며, 생성된 모델에 의해 유사 문서와 비 유사 문서가 분류된다. 분류된 유사 문서는 문서 분석 모듈에 의해 학습 집합과 비교되어 확장질의(EQ)를 생성하게 된다.

2. 질의 추천 시스템 : 초기 질의 실험 결과 및 분석

이번 절에서는 질의 추천 시스템에서 소장 분석을 위해 새롭게 제안한 불용어 목록 및 처리 결과, 각 단락 별 정보량을 이용한 주요 단락 선정, 주요 문장 선정에 대한 실험 결과와 이를 이용해 생성된 초기 질의에 대한 정확률, 재현율, F-척도를 계산 및 이전 버전의 실험과 벤치마킹 한다. 다음의 실험을 위해 TREC Legal Track의 여러 소장 중 Complaint K를 대상으로 분석하였으며, 문서집합으로 EDRM Enron V1을 사용한다.

가. 불용어 처리

불용어 처리를 위해 기존 시스템에서 사용했던 Lucene 불용어 목록과 위키에 제시된 불용어 목록을 이용해하여 일차적으로 불용어 목록을 제거한다. 여기서 제거되는 단어들은 보편적인 영문장에 사용되는 불용어로서 "a", "about", "above", "across"와 같은 대부분의 문서에서 볼 수 있는 단어들이다.

하지만 이 불용어 목록에 모든 불용어가 모두 포함된다고 가정 할 수 없기 때문에 추가적인 불용어 제거가 필요하다. 이를 위해 전처리 과정에서 '품사 부착' 을 이용하여 전치사, 접속사, 관사와 같은 문맥적 의미를 갖지 않는 품사를 제거한다. 또한 OpenNLP의 경우, 특수문자 및 구두점들에 대한 처리방법이 없기 때문에 추가적으로 1글자로 구성된 목록들을 제거 한다.

이 두 과정을 거치면서, 많은 불용어가 제거되며 분석의 많은 잡음들이 제거 된다. 하지만, 법률 용어가 빈번하게 사용되는 소장에서 법률 관련 특수 불용어가 추가적으로 제거 되어야 한다. 이 특수 불용

어는 plaintiff, defendant와 같이 일반적인 문서에서는 사용되지 않는 법률 관련 단어들이다. 다음 [그림 16]은 세 과정의 불용어 제거 결과 중 일부로서, 많은 양의 불용어가 제거되었음을 알 수 있다.

Complaint K(원본)		Complaint K(일반 불용어)		Complaint K(법률 불용어)	
단어	빈도	단어	빈도	단어	빈도
the	344	class	63	oil	52
,	256	plaintiff	55	bleak	33
.	234	defendants	53	horizon	31
and	228	oil	52	spill	31
of	199	members	50	explosion	21
to	142	new	48	blowout	19
a	84	searchland	40	drilling	15
in	69	bleak	33	volteron	15
class	63	horizon	31	failure	14
that	59	spill	31	preventer	12
or	56	damages	23	properly	11
plaintiff	55	action	22	resulting	9
defendants	53	explosion	21	cementing	9
oil	52	blowout	19	bulene	9
members	50	caused	15	determined	8
new	48	drilling	15	negligence	8
by	46	volteron	15	searchland	8
on	45	failure	14	offshore	8
as	41	defendant	12	rig	8
searchland	40	preventer	12	failed	7
is	37	business	12	businesses	7
bleak	33	properly	11	entitled	7
horizon	31	result	10	gulf	7
spill	31	including	10	occurred	7
)	31	resulting	9	environments	7
from	27	cementing	9	resort	7

[그림 16] 일반 불용어 제거 및 법률 불용어 제거 결과

나. 주요 단락 선정

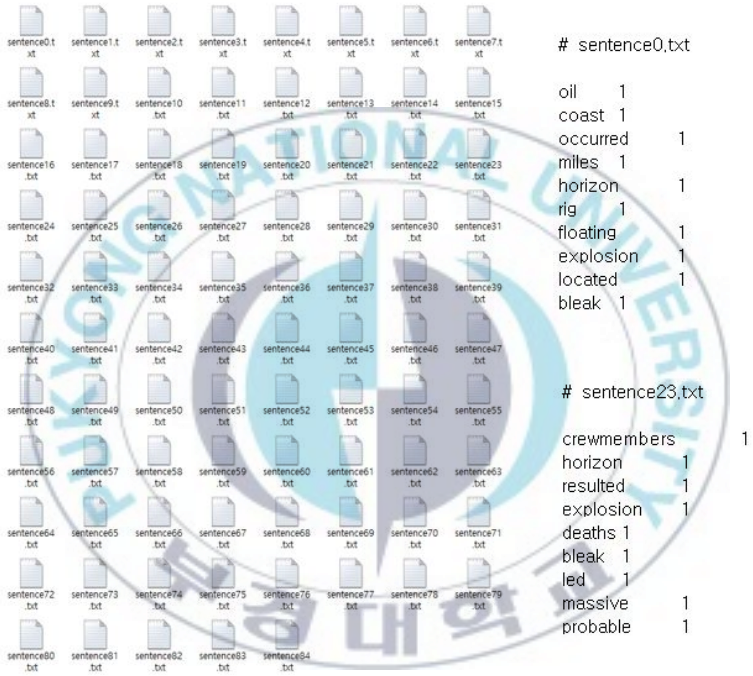
불용어 처리가 된 소장 전체 및 각 단락 별 정보량에 실험 결과는 다음 [그림 17]과 같다.

실험 결과 소장 전체의 정보량보다 높은 단락으로 Count I, Count II, Factual Allegation이 후보 단락으로 선정되었고, 핵심 단락으로서 후보 단락의 평균치(4.8650)보다 높은 Factual Allegation 단락만이 핵심 단락으로서 선정되었다.

	Complaint K		Abstract	N/A	Parties	Count I	Count II	Count III	Count IV	F/A	J/V	P/A	P/R
entropy(real)	3.5794		1.3749	1.5689	1.5637	1.4781	1.5263	1.5800	1.4925	1.5036	1.4890	1.5678	1.5097
entropy(max)	6.2110		2.1808	1.9668	2.0848	4.5237	1.6764	5.4284	1.6463	9.2047	2.2071	3.0682	1.6105
gap	2.6316		0.8058	0.3979	0.5211	**3.0456**	0.1501	**3.8484**	0.1538	**7.7011**	0.7182	1.5005	0.1009
C : Comp.(Gap)<Para.(Gap)			NC	NC	NC	C	NC	C	NC	C	NC	NC	NC
Core : Avg(C)<C						not-Core		not-Core		Core			

[그림 17] 각 단락별 정보량 수치 및 주요 단락 선정

다. 주요 문장 선정



[그림 18] 주요 문장 선정을 위한 문장 별 분리

[그림 18]와 같이 중요 문장을 선정하기 위해 주요 단락으로 선정된 F/A를 자연어 처리기법을 이용해 단락 전체를 개별적인 문장으로 세분화하며, 각 문장은 불용어가 제거된 용어들로 구성된다.

이 과정을 통해 F/A는 85개의 문장으로 구분 되었으며, [그림 18]은 전체 문장 중 일부의 문장에 대한 내용이다. 각 문장 별 대표성 척도 결과 중 가장 높은 상위 5개의 문서는 sentence0, sentence4,

sentence35, sentence40, sentence49가 결정되었으며, 각 문장들에 대한 연관성 수치는 다음 [표 8]과 같다.

[표 8] 문장 대표성 척도 중 상위 5개의 문장

문장	문장 연관성 척도	최대 연관 수치	문장 대표성 척도
#0	1.041	0.114	0.927
#4	0.945	0.114	0.831
#35	0.962	0.088	0.874
#40	1.018	0.082	0.936
#49	0.911	0.114	0.797

위의 대표성 척도를 기준으로 분석으로 40번째 문장이 F/A 단락을 가장 잘 대표하고 있고, 또한 이 문장 대표성 척도에 따라 가장 높은 값을 우선으로 초기 질의가 결정된다.

라. 주요 단락의 핵심 용어 및 일반 용어 분류

이 실험 결과는 소장 전체 용어 출현 분포에서 파레토의 법칙의 적용을 통해 핵심 용어와 일반 용어를 구분한 실험 중 일부 결과다.

아래의 [그림 19]와 같이 불용어가 제거된 소장에서 출현한 전체 단어는 1250건의 단어가 존재했고, 이중 상위 20%에 해당하는 단어 수는 250건이다. 따라서 상위 20%에 해당하는 용어들은 최대 출현 용어인 oil을 시작으로 properly까지가 주요 단어로 취급되며 나머지 목록들은 일반 용어로 구분된다.

Complaint K	
oil	52
bleak	33
horizon	31
spill	31
explosion	21
blowout	19
drilling	15
volteron	15
failure	14
preventer	12
properly	11
resulting	9
cementing	9
bulene	9
determine	8
negligenc	8
searchlanc	8
offshore	8
rig	8
failed	7
businesses	7
entitled	7
gulf	7
occurred	7

sum	1250
k=20%	250
sum(20%)	254
k	11

[그림 19] 핵심 용어와 일반 용어의 임계치

마. 초기질의 신뢰성 평가

다음 [표 9]은 초기 질의에 대한 실험 결과로서, 앞 선 핵심 단락 및 주요 문장 결정 후 상위 5개의 주요 문장에 대한 실험 결과이다.

- 생성한 초기질의 신뢰성 평가

[표 9]를 통해 알 수 있듯이, 가장 높은 대표성 척도를 갖는 두

문장 sentence0과 sentence40이 9~15%의 정확률, 나머지 문장들의 경우 3~4%의 정확률을 갖는다. 또한 초기질의에 의한 실험 결과, 낮은 재현율에 의해 F-척도가 다소 낮은 결과로 등장하였다.

[표 9] 초기 질의 실험 평가

Sent.	Hit	Rel.	Prec	Recall	F1
#0	243	22	0.0905	0.0073	0.0135
#4	33	1	0.0303	0.0003	0.0007
#35	2	0	0	0	0
#40	554	87	0.1570	0.0288	0.0487
#49	21	1	0.0476	0.0003	0.0007

문장 대표성
0.927
0.831
0.874
0.936
0.797

하지만 초기 질의의 경우, 소장에서 등장한 용어만으로서, 생성된 질의이므로 이러한 결과는 당연한 결과이다. 이러한 결과는 높은 정확률을 바탕으로 추가적인 정보들을 이용한 확장질의를 이용해 추가적인 정보가 획득되므로 최종적인 추천질의의 재현율과 F-척도 모두 증가하게 된다.

- 이전 시스템과의 벤치마킹 비교

[표 10]은 이전 시스템에서 사용했던 초기 질의 및 그 질의에 대한 실험 결과이다. 하지만, 제안하는 시스템은 이전 시스템과 달리 많은 불용어가 제거되었으며, 선택된 문장 및 질의 생성 방식이 다르기 때문에, 직접적인 비교가 될 수는 없기 때문에, 간접적인 비교를 통해 생성된 초기 질의에 대한 성능을 벤치마킹 한다.

[표 10]을 통해 알 수 있듯이 이전 시스템의 초기질의는 많은 양의 검색 결과와 적은 양의 적합문서에 의해 3~4%의 정확률과 3~10%의 재현율에 의해 제안한 질의추천 시스템에 비해 높은 F-척도 결과를 보였다.

[표 10] 이전 시스템의 초기 질의 실험 평가

Line No.	Hit	Rel	Prec	Recall	F1
#19	3943	182	0.0462	0.0599	0.0521
#23	5413	256	0.0473	0.0842	0.0606
#24	2258	97	0.0430	0.0319	0.0366
#26	10426	321	0.0308	0.1056	0.0477

따라서 이전 시스템에 비하여, 재현율과 F-척도는 낮지만 정확률이 중요하게 여겨지는 초기 질의의 맥락에서 새롭게 제안한 시스템의 초기 질의가 더욱 더 의미 있다고 평가 될 수 있다.

3. 질의 추천 시스템 : 학습 집합 구성 및 기계 학습

지난 절에서는 소장의 분석을 통해 생성한 초기 질의 결과를 이용하여, 전체 문서 집합에서 소장과 직접적으로 관련된 문서를 검색하고 그 검색 결과에 대한 평가 했다. 이번 절에서는 전체 문서 집합에서 초기 질의 검색 결과와 유사한 문서들을 분류한다. 많은 양의 문서 중 유사 문서의 효과적인 분류를 위해 하둡, 머하웃과 같은 오픈소스 라이브러리를 이용하여 실험한다.

가. 학습 집합의 구성

전체 문서의 분류를 위해 선행적으로 규칙(Rule)의 생성이 필요하다. 이 규칙은 전체 문서 집합을 목적 변수(분류하고자 하는 대상)에 따라 분류하기 위한 어떠한 기준으로서 반드시 사용된다.

질의 추천 시스템의 경우 이전 단계에서 생성된 소장 분석을 통해

생성된 초기 질의 결과와, 초기 질의에 대한 반대 결과를 목적 변수로서 학습 집합을 구성하고, 이는 전체 대상을 분류하기 위한 일종의 샘플 데이터로서 생각 될 수 있다. 예를 들면, 다음 [표 11]는 초기 질의 중 학습 집합을 구성하는 한 예시다.

[표 11] 학습 집합을 구성하기 위한 질의

목적 변수	질의
R(적합)	(oil* AND spill*) AND (release* OR bulene* OR failed* OR attempts* OR contain* OR stop* OR control*)
NR(비적합)	/.*/ NOT (oil* AND spill*) AND (release* OR bulene* OR failed* OR attempts* OR contain* OR stop* OR control*)

위의 [표 11]에서 알 수 있듯이, R은 소장의 분석으로 생성한 초기 질의이며, NR은 초기 질의와 상반 된 문서들의 샘플링을 위한 질의로서, 문서 전체에 해당하는 '/.*/' 를 정규표현식으로 이용하여 상반 된 문서 집합을 구성하기 위한 질의다.

```
indexDir : /home/galois8609/dataset/edrm/convert/convert_IndexVersion
query : /.*/ NOT (oil* AND spill*) AND (release* OR bulene* OR failed* OR attempts* OR contain* OF

Counting the Indexed File : 685592
Counting hit Documents : 85916

>> path of hit documents
3.916141.O4NBS0AHCHH5CRY1BX5XIIYAALYNOR4DA.1.txt, score : 2.1718278
3.918680.LPQUKPLZ40YRGSTSCNZTG2ZT2SGGUJG3B.1.txt, score : 2.1718278
3.923467.EZBQBPH22LWMILY0MXIYFPQXM3S3FEYNB.1.txt, score : 2.1718278
3.921975.H12PO1QDMBC2BDYVDPZI5LN2BGI0ZH3LB.1.txt, score : 2.1718278
3.929174.EAX4TPODL4C5NN53WBTLLSGPMZQP1MN2A.1.txt, score : 2.1718278
3.727685.DEJEJ3M42IIGAPKR0RURVNGRLII1AUWSB.txt, score : 2.1718278
3.720916.M5OC1KC52VGIATCP4LKVITEADFZE54BGB.txt, score : 2.1718278
3.723900.BYVRYTDZ3X4KXXNJVWRAXCZH2MAB1ZVLB.txt, score : 2.1718278
3.727687.AUASSQBIMTXUVT5EMEQ5ZI113H202S3TA.txt, score : 2.1718278
3.727839.PF2FBUJSWK3HYZ1ANCUVGNYNKJBHZQ5NB.txt, score : 2.1718278

Searching Time : 2240 ms
```

[그림 20] NR 집합과 루씬의 가중치 함수

이 R과 NR의 문서 비율은 편향된 결과를 갖지 않기 위하여 R의 개수만큼 NR을 구성하며, 루씬의 가중치 점수에 의해 샘플링 한다.

나. 규칙(Rule) 생성

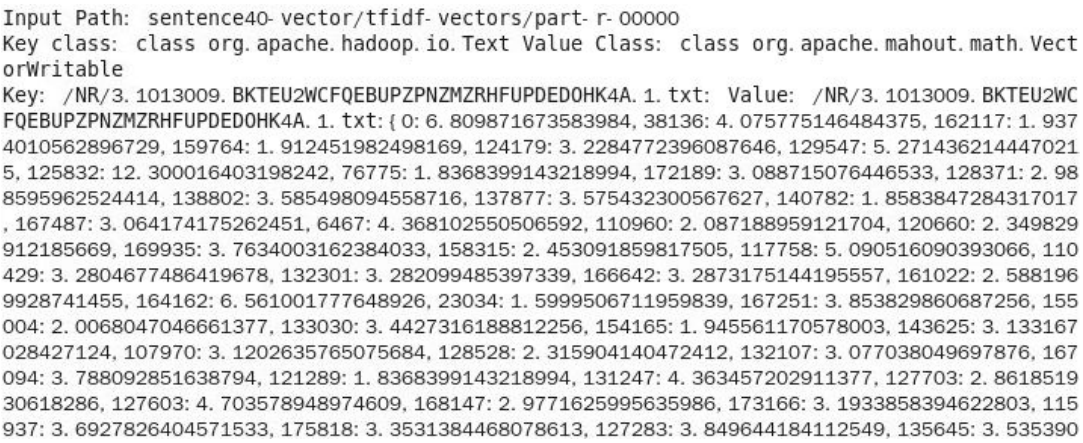
머하웃의 규칙을 생성하기 위해 몇 가지 전처리 과정이 필요하다. 우선 학습 집합에 대한 규칙 생성을 위해, HDFS에 학습 집합을 업로드 한 후, seqdirectory 명령어를 이용하여 하둡에서 사용되는 시퀀스 형태로 변환한다. 다음 [그림 21]은 seqdirectory를 사용한 결과 중 일부이다.

```
Input Path: sentence40-seq/part-m-00000
Key class: class org.apache.hadoop.io.Text Value Class: class org.apache.hadoop.io.Text
Key: /NR/3.1013009.BKTEU2WCFQEBUPZPNZMZRHFUPDEDOHK4A.1.txt: Value: UCLA Anderson Forecast
, June 2001 California-2.1
Will the California Energy Crisis Derail the State's Economy?
Summary
The California economy currently confronts a
number of difficult challenges. The high tech sector
has shifted from booming growth to a bust, dropping
the annualized rate of growth of employment in the
heart of Silicon Valley, Santa Clara County, from 5.9
percent in 2000 to a minus 3.0 percent so far in 2001.
The slowing of the US, Asian, and Mexican
economies is limiting the demand for virtually all of
California's products, causing a drop in the rate of
job growth in Los Angeles from 1.9 percent in 2000
to 0.4 percent so far in 2001.
Into this rather gloomy and probably fragile
economic outlook, insert an electricity supply shortage
that has caused a sharp increase in the wholesale
electricity costs and has prompted blackouts across
the state. CERA and the UCLA Anderson Business
Forecast Project joined together to assess the effects
```

[그림 21] seqdirectory를 이용한 R과 NR의 시퀀스 변환

이후, 이 결과를 seq2directory 명령어를 이용해 학습 집합에 사용될 입력 형태로 변환한다. 이 변환을 통해 문서들에 대한 tf, idf, tfidf, wordcount등과 같은 다양한 수치들이 규칙 생성 이전에 계산되어, 규칙 생성에 사용된다. 다음 [그림 22]은 seq2directory를 사용한

tfidf 결과 중 일부이다.



[그림 22] seq2directory를 이용한 벡터화 변환 중 tfidf 결과

이와 같은 전처리를 거친 후, 규칙을 생성하며, 각 초기 질의에 따라 규칙은 서로 다르게 구성된다. 이 과정에서 생성되는 규칙은 이용하여, 전체 문서 중 유사 문서와 유사하지 않은 문서를 분류한다. 또한 한번 생성 규칙은 한번 생성 된 후, 추가적인 계산이 필요 없기 때문에 유사한 소송의 경우에 재사용 될 수 있다는 장점을 갖는다.

다. 유사 문서의 분류

앞 절에서 생성한 모델을 이용하여 전체 문서 집합 중 소장과 유사한 문서들을 분류한다. 이렇게 분류된 유사 문서들은 학습 집합과 직/간접적으로 유사한 문서들로 구성된다.

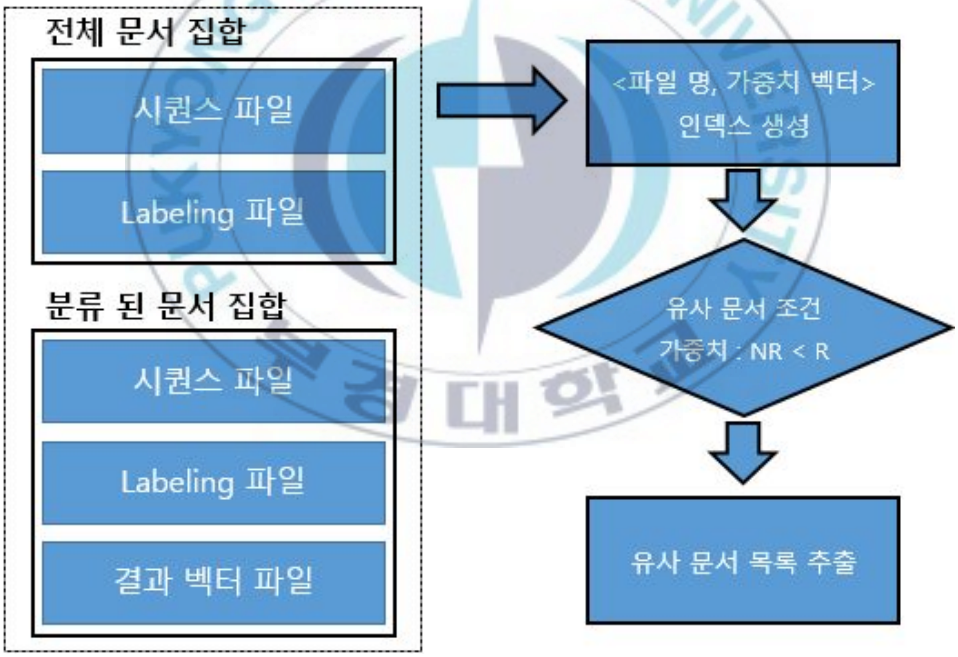
[그림 23]은 전체 문서를 대상으로 유사 문서를 분류한 한 예로서, R에 해당하는 11개의 문서가 초기 질의와 유사한 문서이며, 나머지 67만개의 문서는 유사하지 않은 문서들임을 알 수 있다.

```
=======================================================
Confusion Matrix
-------------------------------------------------------
a       b       <-- Classified as
0       0        |  0          a     = NR
674597  11       |  674608     b     = R

=======================================================
```

[그림 23] 규칙에 의한 유사 문서 분류

하지만, 머하웃을 이용한 분류 결과는 문서들의 분류 결과만을 제공 할 뿐, 어떠한 파일이 유사한 파일인지에 대한 결과는 제공하지 않는다. 따라서 유사 문서 목록을 획득하기 위해 다음 [그림 24]과 같은 추가적인 처리가 필요하다.



[그림 24] 유사 문서의 추출을 위한 처리 과정

이와 같은 처리를 통해 각 질의 별 유사 문서들의 분류 및 목록들을 추출 할 수 있으며, 다음 [표 12]는 전체 문서들의 학습 및 분류 결과이다.

[표 12] 각 질의 별 유사 문서 분류 결과

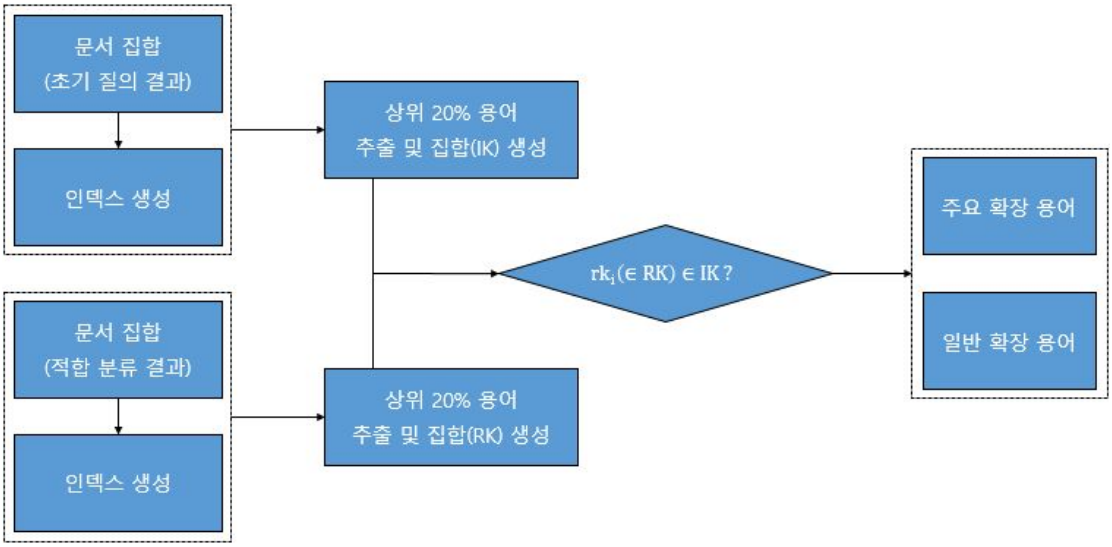
문장 번호	R(유사 문서 수)	NR(비유사 문서 수)
sentence0	11	674597
sentence4	90	674518
sentence35	17	674591
sentence40	673290	1318
sentence49	983	673625

4. 질의 추천 시스템 : 확장 질의 및 추천 질의 생성

이번 절에서는 초기 질의에 대한 문서들과 유사 문서들의 분석으로 유사 문서 집합에서 추가적인 정보(키워드)를 추출하고, 초기 질의와 확장 질의의 결합으로 최종적으로 추천질의를 사용자에게 추천한다.

가. 분류 결과 분석 및 확장 질의 생성

파레토 법칙처럼 문서들은 빈번하게 등장한 단어들에 의해 나머지 단어들이 결정된다고 할 수 있다. 따라서 초기 질의 생성 과정과 유사하게 상위 20%의 단어만을 이용하여 질의 확장을 위한 키워드를 추출하고, 이를 이용한다. 확장 질의는 [그림 25]과 같은 과정을 통해 생성 되며 이는 사용자에게 최종적으로 추천되는 추천 질의에 사용된다.



[그림 25] 분류 결과를 이용한 확장 질의 생성

확장 질의는 초기 질의와 동일하게 주요 용어와 일반 용어로 구성되고, 이 두 용어 집합의 AND 연산의 결합으로 생성된다. 다음 [그림 26]은 이러한 질의를 생성하기 위한 한 예시다.

Initial Keyword		Extend Keyword		
Keyword	Term Freq.	Keyword	Term Freq.	exist?
shall	197719	day	52	No
seller	169040	image	21	No
any	154692	total	18	No
purchaser	135110	bidflag	16	No
agreement	75514	time	15	No
unit	69635			
performance	67769			
equipment	64280			
type	62632			
all	57540			
do	56883			
area	55286			
please	53966			
text	53533			

[그림 26] 초기 질의 문서 집합과 확장 질의 문서 집합

[그림 26]은 초기 질의 결과(IK)들의 적합 분류 된 문서(EK)들의 상위 20% 에 해당하는 용어들 중 일부이다. 여기서 적합 분류에

해당하는 day, image, total, bidflag, time에 해당하는 키워드가 IK 집합에 존재하는지 판단하고, 이 결과를 이용해 핵심 용어와 일반 용어로 구분한다. 위 [그림 26]의 경우 핵심 용어는 존재하지 않고, 일반 용어만 존재하는 경우이며, 따라서 위 5개의 용어에 대한 OR 연산으로 조합하여 확장 질의를 구성한다.

나. 추천 질의 생성

질의 추천 시스템의 마지막 과정으로서 초기 질의와 확장 질의를 조합하여 사용자에게 추천하게 되는 추천 질의를 생성한다. 보편적인 정보 검색 시스템과는 달리 전자증거개시를 위한 검색 시스템의 경우 증거가 될 가능성이 존재하는 문서를 가능한 많이 검색되어야 하는 특성을 갖는다. 따라서 소송과 직접적으로 관련된 문서들을 포함하여 간접적으로 관련된 문서들 또한 포함하기 위하여 두 질의를 OR 연산을 이용하여 결합하여 추천 질의를 생성하고, 사용자에게 질의를 추천하게 된다.

5. 질의 추천 시스템 : 추천 질의 실험 결과 및 분석

이번 절에서는 초기 질의와 이 질의와 유사한 문서에서 획득한 추가적인 질의를 이용하여 생성한 추천 질의에 대한 실험 결과 및 그 결과를 분석 한다.

▪ 추천 질의 성능 실험 결과 및 결과 분석

다음의 [표 13]은 각 추천 질의에 대한 실험 결과로서 다음과 같은 검색 성능을 보인다. 이 결과를 통해 알 수 있듯이 대부분의 추천 질의는 해당 초기 질의가 높은 성능을 보일수록 높은 정밀도(Precision)와 적은 문서 수의 문서들을 검색한다.

[표 13] 추천 질의 실험 결과

Sentence	Hit	Rel.	Prec.	Recall	F1
#0	151268	2120	0.0140	0.7025	0.0275
#4	455449	3018	0.0066	1.0000	0.0132
#35	263	1	0.0038	0.0003	0.0006
#40	6878	468	0.0680	0.1551	0.0946
#49	68184	1431	0.0210	0.4742	0.0402

따라서 초기 질의 중 가장 높은 성능을 보인 sentence #40의 경우 다른 여러 질의들에 비해 가장 높은 정밀도와 F1 척도를 보였다. 또한, 유사 문서들의 분석으로 생성한 확장 질의를 통해 추가적인 많은 문서들이 검색이 되었고, 그 결과 초기 질의에 비해 다소 정확률이 감소했지만 재현율의 상승으로 인해 F1 척도 역시 초기 질의에 비해 상승하였다. 이 점은 신뢰성 보다 증거의 가능성을 갖는 많은 문서를 찾아야 하는 전자증거개시의 관점에서 당연한 결과이며, 소송에 소모되는 비용이 감소되는 점에서 의미를 부여할 수 있다.

▪ 초기 질의 추천 시스템과의 벤치마킹 분석

다음 [표 14]는 초기 질의 추천 시스템의 추천 질의 성능이며, 이 [표 14]는 개선된 질의 시스템의 결과와 벤치마킹한다. 하지만, 기존의 질의 추천 시스템은 어떠한 분석 없이 무작위로 선택한 문장을 이

용하여 생성된 결과이므로 직접적인 성능의 비교는 될 수 없지만, 전반적인 시스템의 성능의 비교를 위해 사용 될 수 있다.

[표 14] 초기 질의 추천 시스템의 추천 질의 성능

Line No.	Query	Hit	Rel.	Prec.	Recall	F1
No.19	IQ	3943	182	0.0461	0.0598	0.0521
	RQ(And)	3943	182	0.0461	0.0598	0.0521
	RQ(OR)	4369	182	0.0416	0.0598	0.0491
No.23	IQ	5413	256	0.0472	0.0841	0.0605
	RQ(And)	1342	58	0.0432	0.0190	0.0264
	RQ(OR)	30026	364	0.0121	0.1197	0.0220
No.24	IQ	2258	97	0.0429	0.0318	0.0366
	RQ(And)	1952	96	0.0491	0.0315	0.0384
	RQ(OR)	13204	187	0.0141	0.0614	0.0230
No.26	IQ	10426	321	0.0307	0.1055	0.0476
	RQ(And)	4070	128	0.0314	0.0420	0.0360
	RQ(OR)	10426	321	0.0307	0.1055	0.0476
Avg. : RQ(AND)		2827	116	0.0425	0.0381	0.0382
Avg. : RQ(OR)		14506	264	0.0246	0.0866	0.0354

제안한 질의 추천 시스템은 비교적 초기 질의 추천 시스템에 비하여 많은 문서들이 검색되었기 때문에 재현율을 비약적으로 향상 시킬 수 있었다. 초기 질의 추천 시스템의 경우, 무작위로 선출된 4개의 질의에 대한 평균 재현율은 3%와 8%인데 반하여, sentence 35를 제외한 나머지 모든 추천 질의가 이보다 훨씬 높은 재현율을 기록하였다. 또한, 많은 수의 문서가 검색되었기 때문에 2%와 4%인 초기 질의 추천 시스템의 정확률에 비해 다소 감소되었지만, 높은 재현율에 의해 F1 척도 역시 3%인 기존 시스템에 비해 제안한 질의 추천 시스템의 추천 질의가 유사하거나 높은 성능을 보였다.

일반 검색을 위한 추천 질의와는 달리 제안한 시스템은 효과적인 전자증거개시 및 소송에 소모되는 여러 비용을 줄이는 것이 목적인 점

에서 다소 낮은 정확률을 보였지만, 상대적으로 높은 재현율을 보이는 제안한 질의 추천 시스템이 더 효과적이라고 생각 될 수 있다.

V. 결론

2006년 FRCP 개정안을 통해 등장한 전자증거개시의 중요성은 많은 법률 관련 사건들을 통해 알려져 있다. 하지만, 이러한 중요성에도 불구하고 막대한 비용 및 시간과 같은 여러 비용적인 문제로 효과적인 처리를 할 수 없는 점, 복잡한 과정을 처리하기 위한 높은 법률 대리인 의존도에 따른 추가 비용 발생과 같은 한계점이 존재했다.

본 논문에서는 효과적인 전자증거개시 처리와 소송에서 발생하게 되는 추가적인 부담을 줄이기 위하여, 소장 분석을 기반으로 소송에 관련 쟁점 파악 및 질의 추천을 위한 질의 추천 시스템을 제안, 구현 및 실험 하였다.

소송 쟁점을 가장 잘 반영하고 있는 용어들을 추출하기 위하여, 의미를 갖지 않거나 보편적으로 사용되어 검색 성능을 저해하는 일반 불용어와 법률 관련 불용어를 제거했으며, 이를 바탕으로 소송에 대한 핵심 쟁점 파악, 주요 단락 및 대표 문장을 선택하여 초기 질의를 생성하였다. 이 초기 질의는 소장과 직접적으로 관련된 문서를 찾아주는 중요한 역할을 하며, 전체적인 시스템의 성능을 좌우하는데 큰 역할을 했다. 또한 이 초기 질의에 의한 결과를 학습 집합으로서 기계 학습하여 유사 문서를 분류하기 위한 규칙을 생성하였고, 이 규칙을 이용하여 유사 문서를 분류했다. 이 유사 문서들은 초기 질의 결과와 유사한 문서들로 구성되며, 초기 질의를 통해 획득하지 못한 추가적인 정보를 획득 하는 역할을 하였다. 이 초기 질의 및 유사 문서의 분석을 통해 획득한 정보들을 취합하여 추천 질의를 생성하고, 최종적으로 이 추천 질의를 법률 당사자에게 제공하게 된다.

본 논문에서 제안한 질의 추천 시스템의 성능을 평가하기 위해

TREC Legal Track에서 제공하는 소장 및 EDRM Enron 문서 집합을 이용하여 실험하였으며, 법률 당사자에게 추천하는 추천 질의의 성능 평가를 위해, ICACT 2014, Mist 2014에서 제안한 초기 질의 추천 시스템과 벤치마킹했으며, 상대적으로 정확률이 감소하기 했지만, 높은 재현율을 나타냈다. 이는 높은 정확률 보다 높은 재현율을 더 중요시 하는 전자증거개시의 관점에서 의미를 가질 수 있었다.

본 논문에서 제안한 질의 추천 시스템은 소송 당사자에게 질의 추천 뿐 아니라, 한번 생성한 규칙은 유사한 사건에 재사용이 가능하기 때문에, 생성된 규칙만으로도 많은 비용을 줄일 수 있다. 또한, 많은 소송 사건에 대한 데이터와 반복적인 학습이 이루어진다면 더 높은 신뢰성을 보장 할 것이며, 즉, 효과적인 전자증거개시 처리가 가능하다.

제안한 시스템은 많은 연구가 진행되지 않은 자연어 처리와 같은 문서 처리 기법 때문에 다소 낮은 정확도를 보이고 있지만, 이 부분은 자연어 처리 및 문서 처리 기법이 개선될수록 보다 높은 정확도를 보장 할 수 있을 것으로 기대한다.

참고 문헌

[1] 한국EMC 컨설팅, “CEO E-Discovery를 고민하다”, 전자신문사, 2011

[2] TREC Legal Track, http://trec-legal.umiacs.umd.edu/

[3] EDRM, http://www.edrm.net/resources/data-sets

[4] EDRM Workflow, http://www.edrm.net/resources/edrm-stages-explained/

[5] 이헌민, "e-Discovery 솔루션의 성능 평가", 제20권 제1호 pp.599-602, 정보처리학회

[6] ELECTRONIC CODE OF FEDERAL REGULATIONS(e-CFR). "the complaint, title 19: Customs duties, part 210, subpart c". http://www.ecfr.gov/, 2013.

[7] Luhn, Hans Peter. "A statistical approach to mechanized encoding and searching of literary information." *IBM Journal of research and development* 1.4 (1957): 309-317.

[8] Gupta, Vishal, and Gurpreet Singh Lehal. "A survey of text summarization extractive techniques." Journal of Emerging Technologies in Web Intelligence 2.3 (2010): 258-268.

[9] 김계성, 이현주, and 이상조. "단락 자동 구분을 이용한 문서 요약 시스템." 정보과학회논문지: 소프트웨어 및 응용 30.7 · 8 (2003): 681-686.

[10] 김계성, et al. "단락 자동 구분을 통한 중요 문장 추출." 한국정보과학회 언어공학연구회 학술발표 논문집 (2000): 233-237.

[11] 자연언어처리, http://ko.wikipedia.org/

[12] OpenNLP, https://opennlp.apache.org/

[13] NLTK, http://www.nltk.org/

[14] 역색인, http://ko.wikipedia.org/

[15] Manning, Christopher D., Prabhakar Raghavan, and Hinrich Schütze. Introduction to information retrieval. Vol. 1. Cambridge: Cambridge university press, 2008.

[16] Lee, Heon-min, et al. "A Query Recommending Scheme for an efficient evidence search in e-Discovery." Advanced Communication Technology (ICACT), 2014 16th International Conference on. IEEE, 2014.

[17] Lee, Heon-min, et al. "Query Recommending Shceme : Implementations and Evaluation.", Managing Insider Security Threats (Mist), 2014.

[18] List of English Stop Words (PHP array, CSV), http://norm.al/2009/04/14/list-of-english-stop-words/

[19] Hatcher, Erik, Otis Gospodnetic, and Michael McCandless. "Lucene in action." (2004).

[20] 김영수, 신상욱, 홍도원, "ESI 관점에서의 e-Discovery", 정보통신산업진흥원 주간 기술 동향, 2010

[21] Apache Tika, http://tika.apache.org/

[22] Apache Lucene, http://lucene.apache.org/

[23] Pareto Principle, http://en.wikipedia.org/wiki/Pareto_principle

[24] Apache Hadoop, http://hadoop.apache.org/

[25] Apache Mahout, http://mahout.apache.org/

[26] Gupta, Vishal, and Gurpreet S. Lehal. "A survey of text

mining techniques and applications." Journal of emerging technologies in web intelligence 1.1 (2009): 60-76.

[27] Hadoop, http://ko.wikipedia.org/