Query Recommending Scheme : Implementations and Evaluation*

Hoen-min Lee, Taerim Lee, Kyung Hyune Rhee, and Sang Uk Shin†
Pukyong National Univ., Busan, Korea
{galois8609, taeri, khrhee, shinsu}@pknu.ac.kr

Abstract

In general e-Discovery processing, most litigants depend heavily on lawyers for dealing with a series of work required in the litigation process. Although there are other reasons that impel the lawyer to do so, the most serious problem is that lawyer cannot know the detail information of litigant's data from the beginning. It causes misuse of keyword and then it makes poor results of evidences search, so it will lower the efficiency of entire e-Discovery work. To solve these problems, we proposed the concept of QRS through the ICACT 2014. In this paper, we introduce how we develop the QRS as a system and evaluate the performance of QRS by experiments based on the two legal cases used in TREC Legal Track.

Keywords: e-Discovery, Query Recommending, Machine Learning, Evidence Search

1 Introduction

Electronic discovery, first introduced by Federal Rules of Civil Procedure amendments on December 1 2006, refers to discovery in civil litigation which deals with information in electronic format also referred to as ESI(Electronically Stored Information)[11]. This is the result that reflects the modern flow that Discovery's main target is ESI. According to these rules, each company has the responsibility to produce their own evidence for winning the suit, and the use of digital forensic tool is almost a necessity.

The general e-Discovery process consists of six to eight stages showed in Figure 1, depending on the particular focus and segmentation. Each steps, identification of content and its scope, data gathering, media restoration, data processing, on-line review, production and delivery of results and subsequent consulting on the part of the Service Provider, are processed sequentially and these steps are essential process for e-Discovery. In civil litigation or corporate matters digital forensics forms part of the electronic discovery process. Forensic procedures are similar to those used in criminal investigations, often with different legal requirements and limitations. Outside of the courts digital forensics can form a part of internal corporate investigations. Thus, e-Discovery process connotes that main focus of lawsuit is to find evidence.

In general, most litigants depend heavily on lawyers for dealing with a series of work required in the litigation process, thus it costs a large amount of money. Moreover, e-Discovery makes every litigant have a responsibility to produce their own evidence for themselves, so the use of digital forensic or e-Discovery tools becomes necessary. In the end, the most important thing is whether or not the litigant can find relevant evidence to prove his legitimacy for trial. Also, the keywords for evidence search are

Research Briefs on Informaiton & Communication Technology Evolution (ReBICTE), Vol. 1, Article No. 16 (January 15, 2015)

*This research was supported by Next-Generation Information Computing Development Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Science, ICT & Future Planning (No.2011-0029927)

†Corresponding author: postal address: Room 1314, Building 1, Department of IT Convergence and Application Engineering, Daeyeon Campus (608-737) 45, Yongso-ro, Nam-Gu. Busan, Korea, Tel : +82-51-629-6249, E-mail : shinsu@pknu.ac.kr

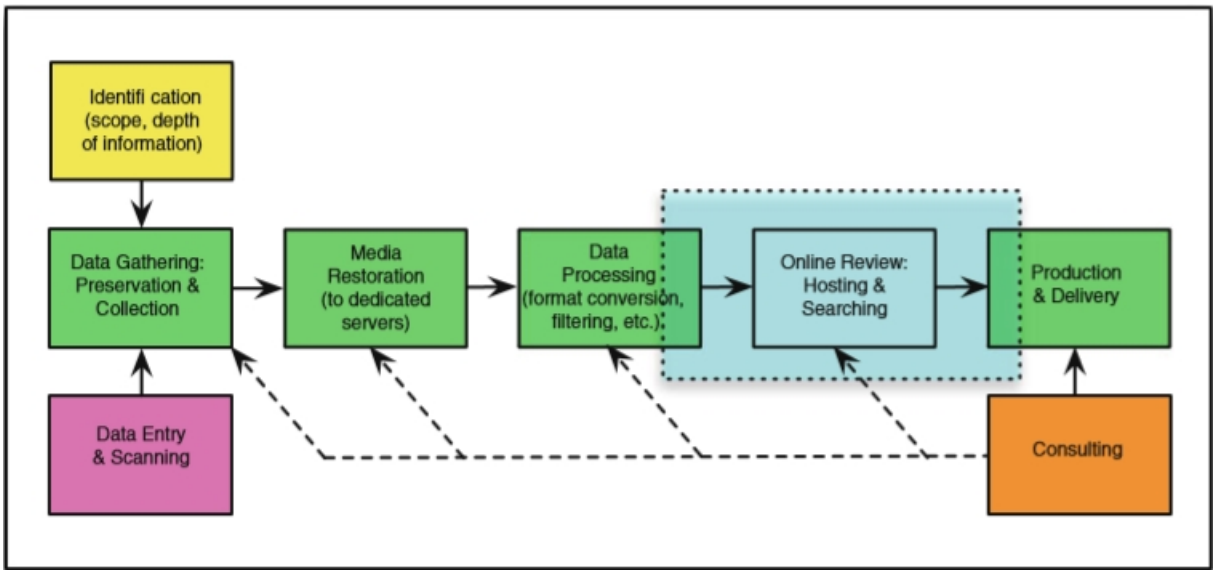

Figure 1: The e-Discovery process : multi-stage workflow[1]

primarily suggested by a lawyer from the analysis of complaint.However, those keywords are usually ambiguous and complicated because the lawyer quoted the contents of complaint without changes. Although there are many reasons that impel the lawyer to do so, the most serious problem is that he cannot know the detail information of litigant's data from the beginning. Misuse of keyword makes poor results of evidence search, so it will lower the efficiency of entire e-Discovery work. Of course, this problem can be solved by the steady counselling or the expert's advice, but additional cost and time will be required.

To solve these problems, we proposed the Query Recommending Scheme, shortly called QRS through the ICACT 2014. QRS creates initial queries by extracting the primary keywords from a complaint like lawyer did and makes some samples for machine learning. Using the samples, it classifies the litigant's data into relevant and non-relevant group. After that, it collects meaningful information from the relevant group and generates extended queries for recommending. However, this scheme was just a conceptual proposal for an efficient evidence search in e-Discovery Procedure as our previous work.

This paper, therefore, introduces how we develop the QRS as a system and explains the details of each implemented module in QRS workflow. Also, it describes the designed experiments for the performance evaluation based on the two legal cases used in TREC Legal Track. Final goal of these experiments is to find out the actual availability of QRS in reality. The analysis of experimental results is performed by comparing the TREC reports of other similar tools. At last, conclusions of this paper and our future work will be described.

2 Related Works

2.1 The main considerations about complaint

A plaintiff starts a civil action by filing a pleading called a complaint. A complaint must state all of the plaintiff's claims against the defendant, and must also specify what remedy the plaintiff wants [9]. This means that main issues of e-Discovery are almost included in this document, so a series of keywords for evidence search are created by the lawyer's review. This is very common way to take action against the lawsuit and to prepare the ensuring e-Discovery work, but it has a high-cost and low-efficiency problem

because of the complete dependence on a specific person like a lawyer. In order to solve the problems, we have to analyse form of complaint. Therefore, based on the analysis of complaint examples provided by Legal Information Institute(LII) in Cornell University Law School [3] and TREC Legal Track [10], the common components of complaint were organized in Table 1.

Table 1: Common components of complaint[4]

Title of Paragraph		Notes
LII	TREC	
N/A(Caption or Heading)		Outline (Jurisdiction, Plaintiff, Defendant, Type of action)
Preliminary Statement	Nature of the action	Summary of the causes for demand trial
Jurisdiction and Venue		Reason why the case should be heard in the selected court rather than some other court
General Allegations	Plaintiff's Allegations Substantive Allegations	List of facts that brougth the case to the court
Count	Cause of Action	A numbered list of legal allegations, with specific details about applications of the governing law to the each court
Demand for relief		The relief that plaintiff is seeking as a result of the lawsuit

2.2 TREC Legal Track

The Text REtrieval Conference(TREC) is an on-going series of workshops focusing on a list of different information retrieval(IR) research areas, or tracks, and the learning task has been conducted in legal track from 2010. The goal of this task is to determine which documents(email messages or attachments, treated separately) should be produced in response to a production request in the complaints. In order to evaluate the success of different systems and strategies for the e-Discovery problem, the organizers of the TREC Legal Track employ a set of both longstanding and newly devised information retrieval measures. Here are simple description of the measures in IR system[5].

- Precision : the ratio of responsive documents in a collection to those responsive documents retrieved.
- Recall : the ratio of responsive documents to the total number of responsive documents in the full collection.
- F-score : the a measure of a accuracy that considers both the precision and the recall.

Traditionally, the concern that discovery searches could be missing important evidence accounts for the TREC Legal Track's orientation towards recall. This tendency is quite natural because losing the case by omitting crucial evidence is the worst at the trial. However, considering the cost for human review, the search result that includes too many documents only for increasing the recall rate will be a big burden to the litigant. This is why the litigant should concern about the precision rate with recall at the same time. Therefore, in our experiments, F-score was calculated to grasp the meanings of both measures at once.

2.3 Query Recommending Scheme

The QRS consists of 4 steps and creates initial queries by extracting the primary keywords from a complaint like lawyer did and makes some samples for machine learning. Using the samples, it classifies the litigant's data into relevant and non-relevant group. After that, it collects meaningful information from the relevant group and generates extended queries for recommending.[4]

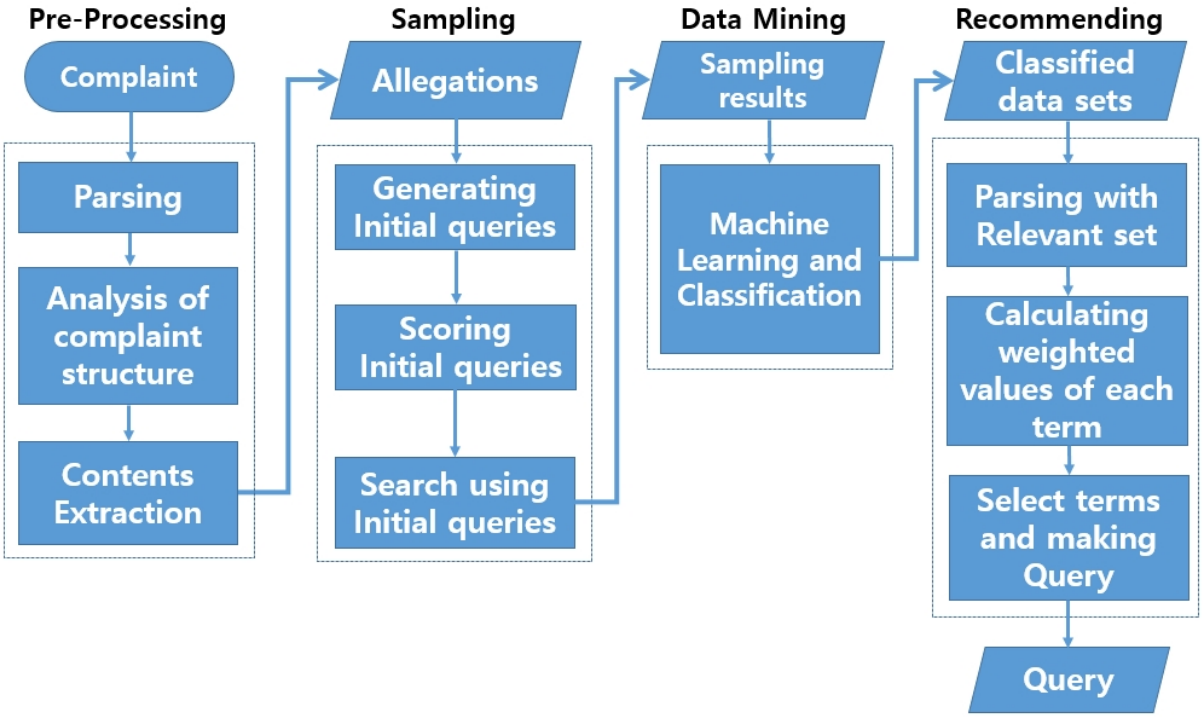

Figure 2: The workflow of Query Recommeding Scheme

- Pre-Processing : The contents of allegation are extracted from the original complaint in this step. To do this, QRS should perform a series of tasks such as text parsing, structure analysis, tokenizing and filtering. So, it is very similar to the general document parser for indexing tools.

- Sampling : This step is to make samples from the search results produced by initial queries. These samples will be used as the training sets in Data Mining phase. In order to generate initial queries, QRS should be able to perform a syntax analysis about extracted contents of allegation because each allegation is a long sentence in general.

- Data Mining : In this step, All documents in document set are divided into two groups, relevant set and non-relevant set. After training from the collected texts in Sampling, it starts to classify each document according to its likelihood. When the classification is completed, QRS only take care of documents belong to a relevant set for next phase.

- Recommending : As QRS did in the previous steps, it starts parsing and scoring with documents of relevant set to make different sets. Separately, it also calculates each weighted value in initial query set. After that, QRS can select the potentially useful terms for query expansion.

3 Implementation of QRS

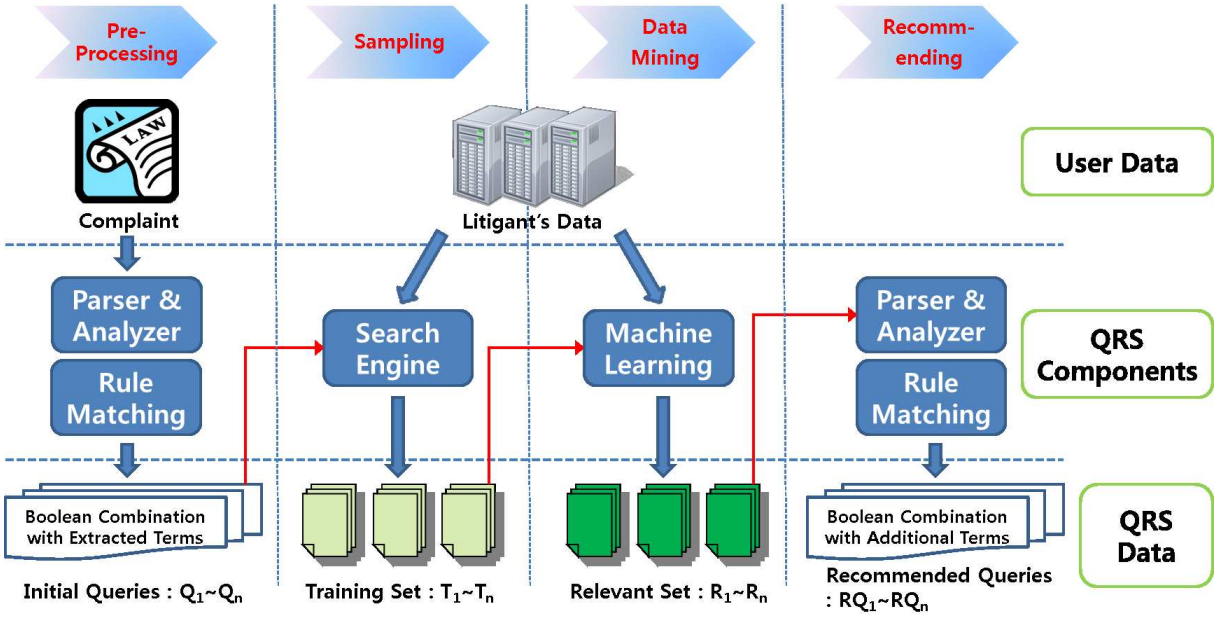

Figure 3: Interconnection between QRS Components

Figure 3 shows how the QRS works at a glance. Each QRS component has a connection with the other parts based on its role which was influenced by input and output data as explained above. Except for given user data, the generated output at every step of QRS becomes the input for the next step. In this section, development method and process for QRS will be introduced and what kinds of open-source libraries were used, the roles of each designed functions and how QRS works by interactions between those functions are described.

3.1 Development Environment for QRS Implementation

Our environment for developing QRS, configurations of test-bed and used open-sources are as follows:

- H/W : Two separated PCs.
  - Hadoop-dedicated system : Independent PC was configured for running Hadoop on single cluster mode, so it could be remained stable without disturbing for dealing with a time-consuming job.(Intel Core i7-2600 CPU, 16GB RAM)
  - Development system : Programming PC.(Intel Core i7-2600 CPU, 4GB RAM)
- OS : Ubuntu 12.04 LTS 64bit
- IDE : Eclipse Platform Version 3.7.2
- Programming Language : Java-7-oracle Version
- Open-source Libraries

- Apache Tika is used for converting document format of complaint from pdf to txt. Commonly, most complaints were written and submitted in PDF format and we also used original complaint files provided by TREC without changes. We did not consider any other formats because document set consists of all txt files in our experiments.[6]
- Apache Lucene is used for document parsing, contents analyzing, terms extracting and document indexing and searching.[7]
- Apache Hadoop is used for distributed processing for large document set and improvement of QRS performance in data mining.
- Apache Mahout provides Hadoop-based data mining algorithms for collaborative filtering, text classification and clustering. We used for creating a model from each training set and classifying all target documents by using a trained model.[8]

3.2 Design and Implementation of Primary Function

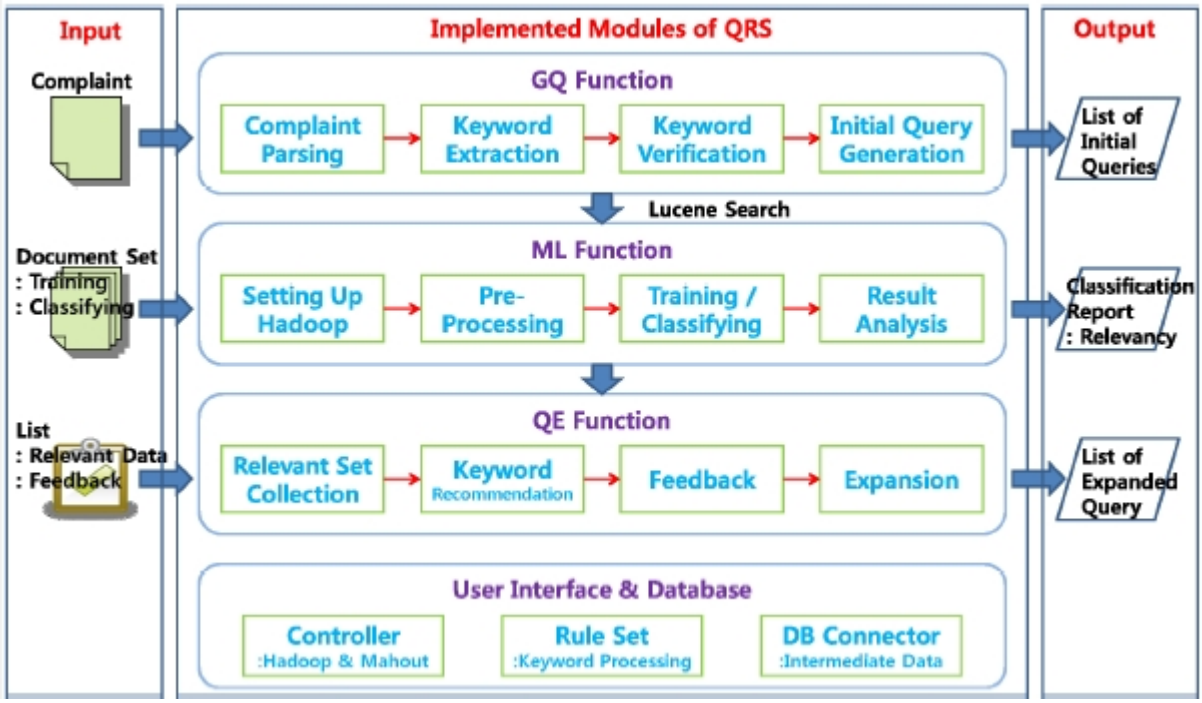

Figure 4: Implemented Modules of QRS

Figure 4 shows the implemented modules of QRS and the details about the roles of each function. In the this section, the developing methods and how to process the input and output data will be described.

- Generating query rules

  These rules on table 2 are applied to QRS system to automatically classify the keywords according to its importance based on the weighted values within a complaint. Especially, the Rule No.1 is to completely prevent outside interference by human in generating query procedure. The reason why we apply the Rule No.2 is that a complaint is usually written by itemizing the primary litigation issues to clarify the cause of legal action. Meanwhile, the rest five rules are to raise a precision rate of the search result for making a high-quality training set, so the Rule No.4 and 5 is to collect

Table 2: The Rules for Generating Initial Query

Rule No.	Details
Rule No.1	Initial queries should be a Boolean type combined by keywords extracted only from a complaint.
Rule No.2	The number of generated queries should be same as many numbered items in the count part of allegation list.
Rule No.3	Every query should be formed of 2 parts each named 'A' and 'O'.
Rule No.4	The keywords from the allegation part(First set) should be added to the 'A' part of query with the AND operation.
Rule No.5	Some of the keywords from the rest part(Second set) which have larger weighted values than the maximum in First set should be equally processed by the Rule No. 4.
Rule No.6	Some of the keywords from the rest part(Second set) which have interval weighted values between the maximum and the minimum in First set should be added to the 'O' part of query with the OR operation.
Rule No.7	Part 'A' and 'O' should be combined with the AND operation.

the keywords which must be included in 'A' part of query. That means the documents searched by this query contain all those keywords without exception. By contrast, although the searched documents do not have to contain all keywords collected by the Rule No.6 in 'O' part, but they must contain more than one. According to the Rule No.3 and 7, these two parts are merged into one query with the AND operation, and it plays an important role to filter the documents which have a relatively low correlation with litigation issues. Finally, QRS system user should create the set of training data for mining stage in person based on the generated initial queries.

- GQ(Generating Query)

  Function GQ is used at the beginning of Sampling in QRS workflow and it includes all tasks for the Pre-Processing step for complaint analysis. This function makes initial queries by using a result of complaint analysis for preparing training sets before Data Mining and it consists of 4 internal processes, Complaint Parsing, Keyword Extraction, Keyword Verification and Initial Query Generation. GQ performs exactly same with this sequence of operations, but processing approaches depend on the 3 different situations.

  Case 1 : The earliest stage in litigation process.

  This is a time of the lawsuit filed, so the defendant received only a complaint document from the plaintiff. Thus, QRS can be used at this time for preparing tasks of evidence search and GQ works as follows.

  - Complaint Parsing : GQ identifies the structure of a given complaint document, and divides its contents into paragraph units.
    1. GQ makes one of text file per separated paragraph by changing the encoding type UTF-8 for the compatibility.
    2. It starts the parsing of these text files with Lucene's analyzer and general stopwords are also removed in this procedure. In addition to general stopwords, GQ has a special term list that should be removed. This list includes legal terms appeared frequently in complaint such as Plaintiff, Defendant, etc.

- Keyword Extraction
  1. GQ extracts 2 keyword sets through the comparison between weighted values of all terms calculated by Lucene's TF(Term Frequency) method. First set is consists of extracted keywords from Allegation part among the separated paragraphs within a complaint. This part generally includes the list of facts that brought the case to the court and a numbered list of legal allegations, with specific details about application of the governing law to the each court. So, the main issues of the litigation could be clearly identified through this part and this means the chances are pretty good to use the first keyword set for evidence search.
  2. Extraction method should select only part of keywords with Top-K weighted values, and K is decided by finding a square root of the highest weight of term.
  3. Second set is generated by the same way as the first, and thus the extraction range is changed to the entire paragraphs of complaint. This will be used for the verification of the first keyword set in next verification.
- Keyword Verification : In most cases, the keywords extracted from the allegation part are more likely to be used as queries for evidence search, but the proof of which is the best way to extract keywords should be required because there is always a danger of missing some important keywords by skipping the rest parts of complaint. Therefore, GQ performs the verification process by comparing the weighted values of two different keyword sets.
  1. GQ removes the same keywords only from the second set, and the mean of "the same keywords" is that all keywords are belong to the both set in common.
  2. GQ gets the minimum and maximum value of weights from the first set to use it as a standard for the selection of additional keywords.
  3. If there are no keywords over the minimum value, all keywords in the second set are ignored. If not, those keywords will be used according to the designated rules in next Query Generation.
- Initial Query Generation : This is the last procedure of GQ and we established simple rules for making initial queries. Table 2 shows the details of those rules.

Case 2 : This is a case of the defendant received a complaint document along with a production request from the plaintiff. A request for production is a legal request for documents, electronically stored information, or other tangible items. It means the descriptions for queries of evidence search are included in that document, but they are not detailed. This is the best situation for using the QRS to find out the potentially valuable keywords.

In general, the sentences in the production request have specific patterns, and several keywords consisted of those patterns such as "All document", "discussing", "referencing", "relating" are not suitable as a query. For this reason, Function GQ deals with those keywords as stopwords and makes initial queries using the rest of keywords with merging by the AND operation. The number of generated initial queries is the same with the requests and GQ does not calculate the weights of terms because it uses the rest keywords without any changes. The following procedure of QRS after GQ is also same with the case 1.

Case 3 : This is a time of the defendant that already have a list of initial queries provided by the lawyers or e-Discovery experts after complaint analysis finished. In this situation, QRS can be

used for evaluating the availability of those queries and extracting additional keywords missed by the human's manual review about the litigation issues. This is the assistant function to reflect the real query negotiation procedure between the defendant and plaintiff to the QRS workflow, and it enables for those people to simulate the Meet-and-Confer session with the automatic system. To achieve this, QRS does not need anything to be added from a implementation point of view, and it can deal with the entire tasks for query recommending more quickly and simply because Function GQ has nothing to do at this time.

- ML(Machine Learning)

  Function ML runs at the stage of Data Mining in QRS workflow and it makes a relative set of documents for each query by using a training data. These relative sets will be reparsed and used for Query Expansion in next stage.

  The following procedure describes how the ML works.

  1. Setting Up Hadoop : ML sends a command to start five daemons for running Hadoop. After checking the Hadoop initialization completed, ML uploads two different set of documents to HDFS for training and classifying.
  2. Pre-Processing : Using the seqdirectory and seq2sparse command in Mahout, ML creates Sequence files from a given document set. This is the transformation process to usually merge the contents of every document into one file in forms of {key:value} and to make a vector representation for TF-iDF with the normalization.
  3. Training and Classifying : Classification model and Label Index are created by calculating term weights in training procedure. Using them, ML can test and get a classification result of every target documents.
  4. Result Analysis : Classification result is also produced in Hadoop Sequence file form, so ML converts the result file to the text format and extracts valuable information for making the classification report. Through the report, user can find all categories where each document belongs to and this will be used in Query Expansion stage.

- QE(Query Expansion)

  Function QE is used for the Recommending stage in QRS workflow and most of its tasks are incredibly similar to the Function GQ. QE makes the expanded queries by using the results of document classification received from the Function ML to recommend the potentially valuable keywords, from now on shortly called PVK, for the evidence search. The following procedure describes how the QE works.

  1. Relevant Set Collection : At first, QE collects all documents listed in the classification result file from the entire dataset and these documents were judged by ML as a relevant data for each query.
  2. Keywords Recommendation : QE shows the list of potentially valuable keywords per each query as a first output for QRS user, and the methods for document parsing, keyword extracting and selecting are exactly same with the function GQ.
  3. Feedback : QE should get the users feedback to decide how to merge the PVKs into initial queries. The purpose of using a search system in digital investigation is to enable for users finding and reviewing their data as evidence before the trial. If too many documents were

searched by the specific initial query, this result makes it difficult because the user has to read all contents of the documents one by one. In this case, user obviously wants to reduce the number of searched documents. On the other hands, user wants to get more in the case of too few documents were searched. Therefore, the feedback is essential for QE to apply the requirements of users.

4. Expansion : All of these requirements for evidence search are closely related to the precision and recall rate. According to the user feedback, when QE makes the expanded queries, it merges the PVKs into 'A' part of initial queries with the AND operation to raise the precision rate. In opposite case, QE makes the third part named 'AO'('AO' is the abbreviation of Additional OR) and PVKs are merged into this part with the OR operation for increasing the recall rate of evidence search.

4 Performance Evaluation

4.1 Experiment Design

The experiments for the performance evaluation of QRS were designed by using two legal cases introduced in TREC Legal Track, Complaint A(TREC 2006) and Complaint K(TREC 2010 task). Unlike any other cases, what is interesting about these requests is that query negotiation process is open between the defendant and the plaintiff. So,The queries proposed by the defendant can be used as initial queries for the QRS experiments.

The experiments are basically performed to check the efficiency rate of change when the recommended queries are used versus initial queries used. As mentioned early, QRS has three types of method for initial query making. According to the method, the experiments were classified three cases and the purpose of each experiment is as follows.

- Case 1. Complaint Only : Based on the rules suggested by the Table 2, QRS makes initial queries only using the contents of complaint document, especially the allegation part. The used document is TREC 2010 Complaint K, and this is the opposite experiment for comparing its effectiveness with case 2.

- Case 2. Production Request : Complaint K will be used again for initial query making in this case. But, the difference is only using the requests for production in that document. These queries are created by removing the terms which were consisted of specific patterns, so they are more simple, but full of implications.

- Case 3. Defendant Proposal : Defendant already have a list of initial queries provided by the lawyers or e-Discovery experts after complaint analysis finished. These queries are usually open at Meet-and-Confer session for negotiating with plaintiff. For this case, the queries included in TREC 2006 COMPLAINT A will be used without changes.

4.2 Document Collection and QRELS

In order to evaluate the efficiency of Information Retrieval System, the approved document collection as the object of search and data for identifying the answers of each topic should be required. All experiments were designed based on the two legal cases introduced in TREC Legal Track, so the same document collections were used in these experiments.

- IIT CDIP 1.0 : This collection contains 6,910,192 XML records describing documents that were released in various lawsuits against the US tobacco companies and research institutes and the records contain both text and metadata.
- EDRM Enron v2 dataset : This collection consists of 685,979 files. Although EDRM provides several types of it for the various application, such as PST or XML version, but TXT version was selected. Each TXT file was made by data extraction from emails and its attachment files. It is closely related to the topics of Complaint K in TREC 2010 task.
- QRELS : This is a file for the representation of relevance judgments. In TREC parlance, qrels are judgments made by humans as to whether a document is relevant to an information need(i.e., topic).

4.3 The Results and Analysis of Experiments

Table 3, 4 and 5 show the results of each experiment and the calculated values of three evaluation measure, Precision, Recall and F-measure. And the queries used for Test case 1 and 3 were randomly selected from the entire set of initial queries and we limited the number of queries, only 4 to simplify the experiment. But, in Test case 2, we used all topics of production requests as initial queries without exception. The reason why the topic 202 and 208 were not included is that the number of searched document was just one.

Table 3: The Results of Test Case 1, Complaint Only

Line No.	Query	Hit	Rel	Precision	Recall	F1
No.19	IQ	3943	182	0.0461577	0.05985	0.052119
	RQ(AND)	3942	182	0.046169	0.05985	0.05213
	RQ(OR)	4369	182	0.0416571	0.05985	0.049123
No.23	IQ	5413	256	0.047294	0.084183	0.06056
	RQ(AND)	1342	58	0.0432191	0.019073	0.026466
	RQ(OR)	30026	364	0.0121228	0.1197	0.022016
No.24	IQ	2258	97	0.0429584	0.031897	0.036611
	RQ(AND)	1952	96	0.04918	0.031569	0.03845
	RQ(OR)	13204	187	0.0141624	0.06149	0.023022
No.26	IQ	10426	321	0.0307884	0.10556	0.04767
	RQ(AND)	4070	128	0.03145	0.042091	0.036001
	RQ(OR)	10426	321	0.0307884	0.10556	0.04767

- Result analysis of Case 1 & 2

  Although Test case 1 and 2 were designed by using the same document collection and complaint, but the experimental results have differences as well as similarities. Broadly speaking, the results of case 1 are better, especially the search efficiency of initial queries used. This is because the extracted terms from production requests for initial query made in case 2 are not only fewer but also more implicative than case 1. However, the difference of efficiency was narrowed when the recommended queries were applied. That means QRS found proper terms for the query expansion through the data mining process and its effectiveness was indirectly proved. Also, analyzing the impacts of using the recommended queries on the evaluation measures, the increase of recall rate is striking throughout the entire experiments. Of course, the litigant has to bear the expense of the

number of document growth that should be reviewed, but the QRS can mitigate a risk of missing important evidence. According to TREC's point of view about the performance of IR system for e-Discovery, since the recall rate is the most significant measure, QRS is very suitable for the litigation supporting system.

Table 4: The Results of Test Case 2, Production Request

Topic No.	Query	Hit	Rel	Precision	Recall	F1
Topic 201	IQ	138	5	0.036232	0.00164	0.003146
	RQ(AND)	136	5	0.03676	0.00164	0.00315
	RQ(OR)	169	5	0.029586	0.00164	0.003115
Topic 203	IQ	1801	91	0.05053	0.02992	0.03759
	RQ(AND)	1637	79	0.048259	0.025978	0.033775
	RQ(OR)	1803	91	0.050471	0.02992	0.037572
Topic 204	IQ	1785	81	0.045378	0.026636	0.033568
	RQ(AND)	1661	80	0.04816	0.026307	0.034028
	RQ(OR)	5779	271	0.046894	0.08912	0.06145
Topic 205	IQ	3750	198	0.528	0.06511	0.05831
	RQ(AND)	2174	137	0.06302	0.045051	0.052541
	RQ(OR)	4202	199	0.047358	0.06544	0.05495
Topic 206	IQ	9932	372	0.037455	0.122328	0.05735
	RQ(AND)	7734	335	0.04332	0.110161	0.06218
	RQ(OR)	158321	1877	0.011856	0.61723	0.023264
Topic 207	IQ	806	1	0.001241	0.00033	0.00052
	RQ(AND)	788	1	0.00127	0.00033	0.00052
	RQ(OR)	4283	1	0.000233	0.00033	0.000273

Whereas if we look at the changes of the F1 measures, the values were maintained or increased in case 2, but those were not in case 1. This result was the inevitable consequence of tradeoff between precision and recall. In case 1, the decreasing rate of precision were much larger for the increasing rate of recall because the used initial queries were more detailed and complicated than case 2. All these facts are intimately related with the cost problem. If the users have a burden of increased cost for review, they can use the QRS strategically. Considering the payable cost, after determining a threshold number of documents in advance, the user selects a Boolean operation being used for query expansion according to the number of hit document for each initial query. Operation AND will be selected to increase the precision rate for the case of too many documents were searched and OR will be chosen for vice versa. It is perfectly possible because the review cost per document was usually fixed. Two recommended queries generated by using the same terms and different Boolean operation in each experiment result reflect these features quite well. Actually, it is same as QRS made some queries with the aid of law expert in case 2 because the production requests were written by the plaintiff's lawyer. This means that case 2 is the best situation for using QRS in real litigation.

- Result analysis of Case 3

  The result of case 3 shows the poor effect of QRS on the efficiency of search, but we can find the increase of F1 rate in a minor way. That means there is a chance that the newly extracted

terms which were not included in the defendant proposal(initial query) will be used for the query negotiation in the meet-and-confer session of the litigation process.

Table 5: The Results of Test Case3, Defendant Proposal

Topic No.	Query	Hit	Rel	Precision	Recall	F1
Topic 7	IQ	35209	31	0.0009	0.1879	0.0018
	RQ(AND)	2611	29	0.0111	0.1758	0.0209
	RQ(OR)	35503	31	0.0009	0.1879	0.0017
Topic 8	IQ	9640	38	0.0039	0.2054	0.0077
	RQ(AND)	9631	38	0.0039	0.2054	0.0077
	RQ(OR)	10287	38	0.0037	0.2054	0.0073
Topic 9	IQ	585	3	0.0051	0.0233	0.0084
	RQ(AND)	533	3	0.0056	0.0233	0.0091
	RQ(OR)	586	3	0.0051	0.0233	0.0084
Topic 13	IQ	2059	41	0.0199	0.2563	0.0370
	RQ(AND)	2059	41	0.0199	0.2563	0.0370
	RQ(OR)	2103	41	0.0199	0.2563	0.0362

Table 6: The Reported Results of Each Participant at TREC 06 Task[2]

Test ID	Topics	Hit	Rel	Precision	Recall
humL06t	39	111724	2168	0.0523	0.1612
NUSCHUA1	39	180295	1087	0.0134	0.0559
SabLeg06aa1	39	194996	2174	0.0916	0.1575
UmdComb	39	195000	1025	0.0661	0.0757
UMKCB	39	145847	1762	0.0422	0.1116
york06la01	39	195000	2621	0.0655	0.1115
Avg	39	170477	1086	0.0552	0.01122

Table 6 shows various experiments reported by all participants of TREC Legal Track. Thus, we can benchmark our results. Through Table 6, we can expect Precision rate is 0.0552 and Recall rate is 0.01122 approximately. Comparing the results of participants in TREC Legal Task with our test cases, there seem to be little difference in the values of measures. But, considering the QRS is almost automatic in which the user is not going to get deeply involved, these outcomes are more significant than the TREC reports because they spend a lot of time and effort on that. Even QRS does not need the expert's help like a lawyer. These facts mean that QRS can get a similar level of result at a substantially less cost. Consequently, this is the greatest strength of QRS for e-Discovery in which time and cost which are accounted very important.

5 Conclusion

The object of QRS is reducing the cost and improving the efficiency of e-Discovery work by lessening the dependence on lawyers. To achieve this, we introduced how we develop the QRS as a system. In order to develop the QRS, we designed and implemented primary three functions, and proposed rules for

generating initial query. After the generating initial query, QRS selected and expanded the potentially useful terms for query expansion through the each steps. Also we experimented QRS for the performance evaluation based on the two legal cases used in TREC Legal Track. Through our results, there seem to be little difference for the results of participants in TREC Legal Task. But QRS is a almost automatic and does not need the expert's help, thus, QRS can be accessorily utilized for human activity for the analysis of complaint and litigant's data set, the meet-the-confer and a similar case in future.

Fundamentally, whether the recommended queries are high quality or not depends on the initial queries generated by QRS at the beginning stage because it uses a machine learning method and the training data is the set of searched documents by using the initial queries for extracting the potentially valuable terms. If the litigants are able to use the QRS repeatedly, the chances will be better for getting good queries. It is close to impossible in the real case of e-Discovery, so additional experiment on this were excluded. Eventually, in order to improve the effectiveness of QRS, the researches about the complaint analysis method, the enhanced rules for making initial query and the advanced technologies for data mining should be followed as the future works.

References

[1] Jack G Conrad. E-discovery revisited: the need for artificial intelligence beyond information retrieval. *Artificial Intelligence and Law*, 18(4):321–345, 2010.

[2] TREC Papers from Participating Teams. Trec 2006 proceedings. `http://trec.nist.gov/pubs/trec15/appendices/legal.results.html`, 2006.

[3] Legal Information Institute in Cornell University Law School. Complaint overview, fictional complaint for trespass to land. `http://www.law.cornell.edu/wex/example`, 2010.

[4] Heon-min Lee, Su-bin Han, Taerim Lee, and Sang Uk Shin. A query recommending scheme for an efficient evidence search in e-discovery. In *Proc. of the 16th International Conference on Advanced Communication Technology (ICACT'14), Phoenix Park, PyeongChang Korea*, pages 1237–1241. IEEE, 2014.

[5] Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[6] Chris Mattmann and Jukka Zitting. *Tika in Action*. Manning Publications Co., 2011.

[7] Michael McCandless, Erik Hatcher, and Otis Gospodnetic. *Lucene in Action: Covers Apache Lucene 3.0*. Manning Publications Co., 2010.

[8] Sean Owen, Robin Anil, Ted Dunning, and Ellen Friedman. *Mahout in action*. Manning, 2011.

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

[10] TREC. Trec legal track. `http://trec-legal.umiacs.umd.edu/`, 2006–2012.

[11] Jack Wiles. *TechnoSecurity's Guide to E-Discovery and Digital Forensics: A Comprehensive Handbook*. Elsevier, 2011.

Author Biography

Heon-min Lee received his B.S. degree in Major of Computer and Multimedia Engineering and applied mathematics from Pukyong National University, Busan, Korea in 2013. He is currently pursuing his master's degree in Department of Information Security, Graduate School, Pukyong National University. His research interests include machine learning, data-mining and e-Discovery.

Taerim Lee received his Bachelor and Master of Engineering degrees from Pukyong National University, Busan Korea in 2008 and 2010, respectively. He is currently doing a Ph.D. program in Department of Information Security, Graduate School, Pukyong National University. His research interests include digital forensics, e-Discovery, cloud computing, and machine learning.

Kyung-Hyune Rhee received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Daejeon, Korea from 1985 to 1993. He also worked as a visiting scholar in the University of Adelaide in Australia, the University of Tokyo in Japan, the University of California at Irvine in USA, and Kyushu University in Japan. He has served as a Chairman of Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Department of IT Convergence and Application Engineering, Pukyong National University, Busan, Korea. His research interests center on multimedia security and analysis, key management protocols and mobile ad-hoc and VANET communication security.

Sang Uk Shin received his M.S. and Ph.D. degrees from Pukyong National University, Busan, Korea in 1997 and 2000, respectively. He worked as a senior researcher in Electronics and Telecommunications Research Institute, Daejeon Korea from 2000 to 2003. He is currently a professor in Department of IT Convergence and Application Engineering, Pukyong National University. His research interests include digital forensics, e-Discovery, cryptographic protocol, mobile/wireless network security and multimedia content security.