

Automated Concept Extraction to aid Legal eDiscovery Review

Prasad M Deshpande
IBM Research – India
prasdesh@in.ibm.com

Thomas Hampp
IBM Software – Germany
thomas.hampp@de.ibm.com

Manjula Hosurmath
IBM Software – India
mhosurma@in.ibm.com

Sachindra Joshi
IBM Research – India
jsachind@in.ibm.com

Seema Meena
IBM Software – India
seemeena@in.ibm.com

Abstract

E-Discovery is the process of discovering electronically stored information such as email that is relevant to a legal case. A typical e-discovery process incurs huge costs due to the large volume of information and the requirement of highly specialized and expensive human resources (legal professionals). In this paper, we examine how information management technologies can be used to reduce the high cost. We propose a set of concepts that are helpful in identifying relevant and not-relevant documents. We then develop a set of rule based annotators that automatically identify documents with these concepts and compare their performance with standard off-the-shelf classifiers for building the concept annotators. The rule based annotators have been integrated into the IBM product for e-discovery review called IBM InfoSphere eDiscovery Analyzer.

1. Introduction

Technological advancements of the past few decades have drastically changed the way we communicate and conduct businesses. There is a huge upsurge in electronically stored information or commonly referred to as ESI which is created, manipulated, stored and consumed within an organization. ESI is different from paper information because of its intangible form, volume, transience and persistence. Examples of types of data included in ESI are emails, instant messaging chats, word documents and Web pages. This change in the way we store information led Federal Rules for Civil Procedures (FRCP) [4][8] to codify the requirements of producing relevant electronically stored information and records in a legal

case. These amendments to FRCP gave rise to electronic discovery or e-discovery which is a process for providing ESI that is relevant to a case to the other party.

The unprecedented volume of ESI poses an enormously challenging problem of finding the relevant information to a case. Further, according to Socha Report [5], 60% of the total legal cases warrant some form of e-discovery and this number is going to increase further over the next few years. A typical e-discovery process involves huge costs due to the huge volume of ESI and the requirement of highly specialized and expensive human resource (legal professionals). In this paper, we examine how information management technologies can be used to reduce the high cost.

The process of e-discovery involves several stages as shown in the Figure 1. These stages and their functioning can be given as

- Identification Stage: This stage involves locating potential sources of ESI and determining its scope breadth and depth.
- Collection and Preservation Stage: This stage involves gathering of ESI for further use. This stage also ensures that the gathered data is protected against inappropriate alteration and destruction.
- Processing Stage: In this stage, the data gathered in the previous stages is first converted to forms more suitable for review and analysis. The volume of ESI is then reduced using context, keywords and patterns.
- Review Stage: In this stage the processed data is evaluated for relevance and privilege. This stage of the e-discovery process is most time consuming and expensive.

- Production Stage: The relevant data to a case is produced to the concerning parties in appropriate media form in this stage.

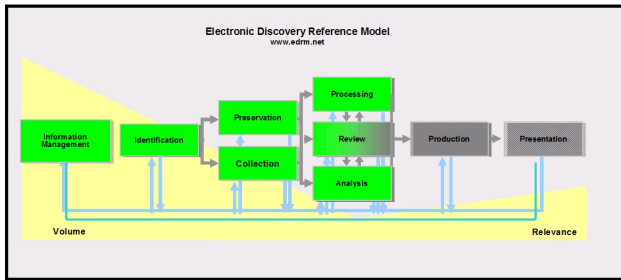


Figure 1. Stages of eDiscovery

The most resource consuming and costly stage is the review stage which involves evaluation of the processed data for relevance to the given case. At the most basic level document review is used to sort out responsive documents to produce and privileged documents to withhold. The responsive documents are the documents that are relevant to the case and are produced to the other parties. Privileged documents on the other hand are the documents that are protected from disclosure under special and exclusive legal right. Examples of privileged document include attorney work product and certain communications between an individual and his or her attorney which are protected from disclosure.

In this paper, we propose a set of concepts that are helpful in identifying responsive and privileged documents. We then develop a set of annotators that automatically identify documents with these concepts. The automatic identification of documents with these concepts expedites the review process and thus reduces the huge cost. We examine the use of standard off-the-shelf classifiers for building the concept annotators. We also build rule based concept annotators using the System T [7] and UIMA [1][9] framework. We find that the rule based concept annotators are better as they are human comprehensible and have similar or better accuracies. We have implemented the rule based concept annotators and integrated them in the IBM product for e-discovery review called IBM InfoSphere eDiscovery Analyzer.

2. Legal Concept Annotators

In this section, we describe the set of annotators that we developed to define the scope of relevance of mails associated to a particular case/set of cases. The legal concepts can be broadly divided into two categories as described below.

2.1 Focus Categories

As the name suggests, focus categories help to reduce the area of interest from a large pool of information. These categories identify mails that are relevant for a case. Focus category annotators identify emails that fall under this category. It is important to achieve high recall for focus category annotators since we would not want to miss out on relevant emails. Focus categories include the following:

- **Legal Content** – This category includes emails with content related to legal issues (other than privileged, intellectual property and harassment). For example, it will detect emails that contain mentions of agreements, contracts, litigations and other legal things.
- **Financial Communication** – This includes emails containing any mention of financial transactions (such as currency, expenditure, filing, reimbursement, price, cost, sale buy, inventory, purchase, stock, trading, traders, and supervising traders). This also includes emails mentioning inventory purchases and generic contracts.
- **Intellectual Property** – This category includes emails that contain mentions of patents, disclosures, license, copyright, trademark, idea, invention or innovation in the body of the email.
- **Job Solicitation** – This category comprises of emails indicating employees trying to find jobs or external people trying to hire from the company.
- **Harassment** – Emails relevant for a harassment case can be identified by any use of inappropriate and unwelcome language or racially/sexually/religiously discriminating language.
- **Audit Info** – This category includes emails containing any kind of compliance check (software, finances, etc) based on an external Certified Public Accountant or CA or internal processes/training.
- **Inappropriate Conduct** – This category includes emails that indicate inappropriate conduct. For example mails including terms like theft, breach of contract, interference, fraudulent, unfair, conspiracy, attack, terrorism etc.
- **Confidential Communication** – This category includes emails with confidential content. Confidential emails are detected based on explicit mention within the email itself that it is a confidential piece of information. The mention of confidentiality can be in the signature of the email.
- **Privileged Communication** – This category is mainly characterized by interactions (directly)

between client and attorney. This is detected based on explicit mention in the email (mainly subject and footer) that it is a privileged communication. This category is used to identify attorneys-client privileged communication.

- **Inappropriate Use Of Property** – Emails in this category includes mail that indicate wrong use of computing resources (like sending adult jokes) by company employees, company premises, strikes, unions, picketing, canvassing.

2.2 Filter Categories

Filter categories identify emails that are irrelevant and that can be filtered out to reduce the effort of manual review process. Filter category annotators identify emails matching the filter categories so that they can be filtered out. It is important to achieve high precision for filter category annotators to ensure that relevant emails are not filtered out. The filter categories includes following:

- **Private Communication** – This category includes all the emails that are personal in nature and not related to work. It could be the mails between employees of the same company.
- **Automated Message** – All automated/machine "sent" (as against "generated") emails such as marketing messages and news fall into this category.
- **Bulk email** – This category includes emails that are manually "sent" (as against "generated" emails) to a large group (greater than 20) of people.
- **Inappropriate Joke** – It includes emails containing jokes with inappropriate language.

We have developed annotators for six of these categories – Legal Content, Financial Communication, Intellectual Property, Confidential, Privileged and Automated Message. We will describe their implementation and results in the subsequent sections.

3. Implementation

3.1 Rule based annotators and System T

The concept annotators can be built using either machine learning or rule based approaches. The advantage of the rule based approach is that the rules are human comprehensible and can be tweaked to get the desired results. Our final goal was to integrate the concept annotators into IBM InfoSphere eDiscovery Analyzer, which is a product to aid e-discovery review. The eDiscovery Analyzer internally uses System T [7] as the framework for specifying and applying the rules.

System T is an information extraction system that enables relatively unsophisticated users to build powerful rule-based annotators that can operate over very large corpora. It enables text-centered enterprise applications by extracting structured information from unstructured text. Unlike previous systems for information extraction, System Text incorporates AQL, a declarative rule language that makes it easy to express precise specifications for complex patterns in text. Extracting information from text can be a CPU-intensive task, and making rules run efficiently has traditionally been a big problem for developers. System Text for Information Extraction solves this problem by relieving developers of the burden of performance tuning. Behind the scenes, the system automatically optimizes rule execution for maximum "throughput," allowing the developer to concentrate solely on building more accurate rules.

The AQL language is similar in flavour to SQL. In the AQL algebra [3], the annotations over a document are treated as tuples of a relation. A tuple is a finite sequence of spans where each span identifies a region of text with its "begin" and "end" positions. Each annotation type produces a different relation. The set of operators in the algebra can be categorized broadly into relational operators, span extraction operators, and span aggregation operators. Relational operators include the standard operators such as select, project, join, union, etc. The span extraction operators identify segments of text that match some pattern and produce spans corresponding to these matches. The two common span extraction operators are the regular expression matcher and the dictionary matcher. The regular expression matcher takes a regular expression, matches it to the input text and outputs spans corresponding to these matches. The dictionary matcher takes a dictionary dict, consisting of a set of words/phrases, matches these to the input text and outputs spans corresponding to each occurrence of a dictionary item in the input text. The span aggregation operators take in a set of input spans and produce a set of output spans by performing certain aggregate operations over the input spans. There are two main types of aggregation operators – consolidation and block. The consolidation operators are used to resolve overlapping matches of the same concept in the text. Consolidation can be done using different rules. Containment consolidation is used to discard annotation spans that are wholly contained within other spans. Overlap consolidation is used to produce new spans by merging overlapping spans. The block aggregation operator identifies spans of text enclosing a minimum number of input spans such that no two consecutive spans are more than a specified distance apart. It is useful in combining a set of consecutive input spans into bigger spans that represent aggregate concepts. For complete details of the algebra, please refer to [1].

3.2 Methodology

To build rules for the concept annotators, we first had to come to a common understanding of what each concept means. This required consulting legal experts to understand what they mean by each of the concepts. We looked up the standard definitions of these concepts from the sources pointed to by the legal experts. The next step is to codify the definitions of the legal concepts into System T rules. The main issues that we had to address were as follows:

Email Segmentation

Emails are divided into meta-data like sender and recipient information, and content fields like subject and body. The body of an email can be divided into different parts. For example, a typical email body contains a greeting (such as “Dear XXX”, “Hi” or “Hello”), the main content, salutation at the end (“Thanks”, “Regards”, etc followed by a name). It can optionally include the signature of the sender and a footnote text. The footnote can include standard disclaimers such as “The content of this email is confidential and subject to attorney-client privilege”. Additionally, many emails are either replies to previous email threads or contain forwarded emails. In this case, the email body includes the content of the email thread being replied to or forwarded. While identifying the concepts in an email, it is important to first segment the email body since the rules may be applicable to specific parts of the email. For example, to identify Privileged email, we need to identify the footnote and check if the footnote contains the privileged declaration. Segmenting the email body consists of two phases:

1. Splitting the email containing a thread of emails into individual email blocks, each corresponding to a single email.
2. For each block, identifying the various parts of the email such as the greeting, content, signature and footer.

Splitting the email into blocks is done by identifying how most of the common email clients include the content of the email being forwarded or replied to into a new email. These patterns are then encoded using the regular expression operators in AQL. Identifying various parts of a block is done similarly by identifying patterns that are typically used in emails. For example, the footer is most often separated from the rest of the email by separator line such as “-----” or “*****”.

Identifying patterns for concepts

For each of the concepts we identified a set of keywords and phrases that are indicative of that concept. For example, keywords such as “patent”, “copyright”, “NDA”, “tradeseecret”, “IP”, and “trademark” indicate that the email may be discussing about intellectual property.

Using the dictionary operator in AQL, we can find occurrences of such words in the text. The regular expression operator is also used to identify patterns that are indicative of certain concepts. For example, mention of currency figures in an email can be used to identify Financial Communication. We used regular expressions to encode such expressions to identify a currency amount in the email content. For each concept, we wrote multiple rules to identify the basic building blocks and the relationships between them.

Consolidation

The rules developed for each concept were independent of each other. Thus, the same email can match rules for different concepts and can be tagged with multiple concepts. In general, such a situation can happen in reality and is not a problem. For example, an email could contain mentions of Financial information as well as Legal content. However, for some other categories, we may have some constraints that preclude the same email from belonging to multiple categories. This could be either due to a “implies” or a “contradicts” relationship between the categories. The relationships for the categories we developed are listed below:

Category 1	Relationship	Category 2
Privileged	Implies	Legal Content
Intellectual Property	Implies	Legal Content
Privileged	Implies	Confidential
Automated Message	Contradicts	Confidential

To handle these constraints, we make a pass after the concepts have been identified and eliminate redundant (implied) or contradicting concepts. The “consolidate” operator in the AQL algebra is used for this consolidation.

3.3 System Architecture

Once the concepts have been discovered, the user should be able to search on these concepts. The three level architecture of our system is shown in Figure 2

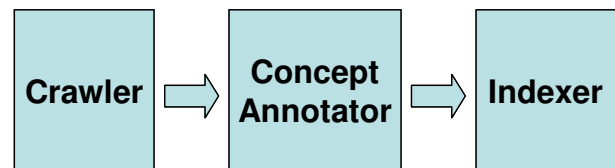


Figure 2. Architecture

The crawler iterates through the emails in the repository (e.g. an email archive) and passes them on to the concept annotator. The concept annotator is the System T runtime engine configured with the rules for identifying concepts.

It process the emails and outputs the concepts identified in the email. This is then indexed by the indexer and a faceted search mechanism is provided to the user to filter based on the concepts. The concepts extracted are displayed as facets in a search UI with faceted navigation elements as shown in the screen shot below:



Figure 3. Faceted search based on Concepts

4. Evaluation

In this section, we present our evaluation and empirical results of rule based annotators. We first describe the dataset and measurement used for evaluation and then compare the performance of rule based annotators with off-the-shelf machine learning algorithms.

4.1 Dataset

We use the Enron email corpus [2]. This data set was originally made public by the Federal Energy Regulatory Commission during its investigation in the Enron case and therefore is especially interesting for research in e-Discovery review. The data set contains 517,431 emails from almost 150 users.

We need to manually tag the data to measure the quality of annotators. For this we sampled some emails from the Enron corpus and tagged them manually. We sampled 2200 odd emails using a set of generic keywords and then manually labelled them in one or more of the following six categories: (1) Automated Messages, (2) Confidential, (3) Financial Communication, (4) Intellectual Property,

(5) Legal Content, and (6) Privileged Communication. The manual creation of this dataset provided us insights into the concepts and helped us in building the rules. We therefore refer to this dataset as “train dataset” in this section.

In order to check completeness of our rules we also asked two other candidates who were not involved in the building of rule based annotators to create a dataset which we refer to as “test dataset”. The following table provide details of “train” and “test” datasets.

Class	Train	Test
Automated Messages	436	22
Confidential	406	21
Finance	1215	54
Intellectual Property	153	22
Legal Content	553	41
Privileged	433	34

4.2 Methods and Measurement

We quantify the quality of our annotators using the precision and recall measures for each class which are defined as follows:

$$precision = \frac{common}{machine}$$

$$recall = \frac{common}{manual}$$

Here *manual* denotes the number of emails tagged with a class in the manually tagged dataset; *machine* denotes the number of emails identified by the annotator (rule based or machine learning based) with the same class label; and *common* denotes the number of emails that are tagged manually and by the machine with the same class label.

In order to see how our rule based approach compares with the machine learning based approach, we pose the annotation problem as a classification problem. We use naïve Bayes classifier and build a two class classifier for each class. We use the rainbow [6] toolkit implementation for naïve Bayes.

4.3 Results

For Rule based annotators we evaluate on “train dataset” as well as for “test dataset”. For naïve Bayes classifier we use “train dataset” for training the classifier and “test dataset” for testing the classifier. For each class we build a two class classifier by creating a class called “Others” by combining emails from all the other classes.

The table below presents the precision and recall numbers for rule based approach as well as for naïve Bayes classifier.

Class	R B (Train)		R B (Test)		Naïve Bayes	
	P	R	P	R	P	R
Automated Message	94	67	92	50	65	50
Confidential	77	87	100	57	22	71
Finance	91	92	52	92	63	94
Intellectual Property	83	92	75	100	79	46
Legal Content	83	82	58	95	34	51
Privileged	87	97	89	94	52	91

In the table R B refers to the rule based approach. The performance of rule based annotators on the “train dataset” is better than the performance on the “test dataset”. As targeted, we are able to achieve a high precision for the filter category of “Automated Message” and a high recall for the other focus categories. The table also illustrates that the performance of rule based annotators is always better than the one achieved by naïve Bayes classifier. Naïve Bayes performs particularly badly for Confidential and Legal Content classes. This is due to the consolidation rules. Many emails that have terms indicating confidentiality are also privileged. Since the Privileged category is given a higher priority, these emails are not included in the Confidential class and are included in the Other class of Confidential. This leads to a poor classifier for Confidential since the discriminating terms for the Confidential class occur in both the Confidential and the Other class used while training the classifier. Similar reasoning holds for the Legal Content class since it is also given a lower priority by the consolidation rules.

5. Conclusions

We have proposed a set of concepts that are helpful in identifying relevant and not-relevant documents for legal review. We have developed a set of rule based annotators that automatically identify documents with these concepts, thus reducing the cost of manually reviewing documents in the e-discovery process. Evaluation of these annotators based on test data shows that these annotators perform well with reasonable precision and recall numbers. We were able to achieve high precision for filter categories and high recall for focus categories. Comparison of rule based annotators with standard off-the-shelf classifiers for building the concept annotators show that the rule based annotators give better results. Rule based annotators have the additional advantage of being human comprehensible and can be fine tuned to get the desired results.

References

- [1] Apache UIMA. Available at <http://incubator.apache.org/uima>
- [2] B. Klimt and Y. Yang, “Introducing the enron corpus,” *Conference on Email and Anti-Spam*, 2004.
- [3] F. Reiss, S. Raghavan, R. Krishnamurthy, H. Zhu and S. Vaithyanathan. An Algebraic Approach to Rule-Based Information Extraction. *ICDE*, 933-942, 2008
- [4] Frcp - federal rules of civil procedure. 2007. Available at <http://www.law.cornell.edu/rules/frcp/>
- [5] G. Scoha and T. Gelbmann. The 2006 scohaelbmann electronic discovery survey report. *Report, Socha Consulting*, 2007.
- [6] A. K. McCallum. Bow: A toolkit for statistical language modeling, text retrieval, classification and clustering." <http://www.cs.cmu.edu/~mccallum/bow>. 1996.
- [7] System T available at <http://www.alphaworks.ibm.com/tech/systemt>
- [8] K. J. Withers. Electronically Stored Information: The December 2006 Amendments to the Federal Rules of Civil Procedure. *Northwestern Journal of Technology and Intellectual Property*, Vol.4 (2), 171 <http://www.law.northwestern.edu/journals/njtip/v4/n2/3/>
- [9] D. Ferrucci, A. Lally. UIMA: an architectural approach to unstructured information processing in the corporate research environment Source. *Language Engineering*. Volume 10 , Issue 3-4 2004.