

Designing a Predictive Coding System for Electronic Discovery

Dhivya Soundarajan and Sara Anne Hook^(✉)

Department of Human-Centered Computing,
Indiana University School of Informatics and Computing,
535 W. Michigan Street, Indianapolis, IN 46202, USA
dsoundar@umail.iu.edu, sahook@iupui.edu

Abstract. This paper presents the preliminary results of a pilot project to design a predictive coding system for electronic discovery (e-discovery) that will be able to handle potentially relevant evidence in a myriad of formats and that will have the features and functionality that lawyers and members of the legal team will find most useful. We developed our predictive coding system to combine available software tools with particular emphasis on usability and in making the user interface as intuitive, attractive and user-friendly as possible. Future work will include a survey and interviews with potential users, testing the system with larger sets of files and documents, and continued refinement of the user interface and backend processing.

Keywords: Electronic discovery · Predictive coding · Technology-assisted review · Legal technology · Law firms

1 Introduction

Over the past year, we have been designing a predictive coding system based on readily available software and natural language processing. Our paper provides a brief overview of existing predictive coding/TAR systems, comparing these systems to what lawyers who are responsible for an electronic (e-discovery) process actually need and highlighting the shortcomings in existing systems that our system will attempt to address in its special features and functionality. We demonstrate the current iteration of our predictive coding system using a typical set of documents and file formats, with screenshots to show our system's information architecture and user interface, how the results of a query are presented to users and the balance between recall and precision. Particular issues that guided our decision-making as we designed our system were security, especially given the lawyer's duties with respect to confidentiality, ease of use, ability to limit access to certain materials and how to train the system itself as well as to provide guidance to potential users. As part of the paper, we outline plans for future work, including more extensive development and testing of our system with larger sets of documents and a broader range of types of files and formats beyond traditional text-based materials and incorporating continuous active learning and built-in alerts, reminders and tips for using our system more effectively.

2 Background

Although questions about evidence in digital format were raised in cases as early as the 1980s and 1990s, the emerging area of law known as electronic discovery (e-discovery) did not begin to find its way into the typical lawyer's lexicon until the mid-2000s. Two major events occurred during this time that marked the true beginning of the field of e-discovery and that continue to form the foundation of how the process is handled today. In *Zubulake v. UBS Warburg*, Judge Shira A. Scheindlin articulated major principles and themes regarding e-discovery, including the responsibilities of lawyers and clients, sanctions for spoliation of evidence and what constitutes accessible versus inaccessible data [1]. In 2006, the Federal Rules of Civil Procedure were amended to incorporate Judge Scheindlin's rulings and to establish the discoverability of Electronically Stored Information (ESI) as an umbrella term intended to encompass both current and future technology and the data that it generates. E-discovery is something that impacts everyone, whether they know it or not, because it deals with the proper collection, preservation, analysis and production of evidence in digital form. To put it bluntly, if you are sued, the opposing party's lawyer will request nearly every piece of digital evidence in any format that might be relevant to the case, including email and text messages and social media. Anyone can find himself/herself needing to comply with requests for potentially relevant evidence – in electronic or paper/hard copy form.

One of the special concerns with e-discovery is that a faulty and incomplete process, particularly during the review stage, can result in sanctions and waive the attorney-client privilege or other confidentiality doctrine if ESI that could and should have been protected is inadvertently produced to the opposing party. Such failures, especially for breaches in confidentiality, can result in disciplinary action being taken against the lawyer in the states where he/she is licensed [2]. The Federal Rules of Civil Procedure were amended again in December 2015 to shorten the timeframes for various stages in an e-discovery process, to place a greater emphasis on proportionality and to provide clarity on when and what kinds of sanctions the court can impose for spoliation of evidence [3]. Courts are already applying the amended version of the Federal Rules of Civil Procedure, which means that an e-discovery process must now be completed in a significantly reduced period of time and with greater specificity required for requests and objections [4]. The e-discovery process becomes increasingly complex as lawyers and clients deal with wearable devices and the Internet of Things, which create and store even more potentially relevant electronic evidence in a wider variety of files and formats.

There have been many attempts to use technology to address the complexity, time commitment, risk and expense of an e-discovery process. In the past ten years, vendors have improved and enhanced the services and software they offer for e-discovery, digital forensics, litigation support and information governance [5]. One of the services that some vendors provide is predictive coding, which is often referred to or included as part of Technology-Assisted Review (TAR). Predictive coding combines computer speed with human reasoning in the form of artificial intelligence and allows the computer to learn and make decisions [6]. Initially, such tools were looked at with considerable suspicion, even though information retrieval, indexing, machine learning

and data analytics have been used in other disciplines for many years [7]. Fortunately, the reticence to use these types of systems in litigation has faded somewhat, illustrated by a long line of cases that start with strong support of computer-assisted review articulated in *Da Silva Moore v. Publicis Groupe*, which is described as the first published opinion recognizing TAR as “an acceptable way to search for relevant ESI in appropriate cases.” [8–10]

Summaries of recent cases about predictive coding/TAR can be found in The Sedona Conference’s new publication, *TAR Case Law Primer* [11]. However, concerns remain about whether predictive coding systems are as effective as they could be, although the court in the *Dynamo Holdings* case noted that the gold standard of human review is a myth as is the expectation that e-discovery requires a perfect response [12, 13]. Other courts have held that while the use of predictive coding is permissible, a responding party cannot be forced to use it [14].

Predictive coding can be used throughout an e-discovery process, including early case assessment, reviewing ESI before production, prioritizing pre-production review, sorting ESI for privilege, for overall quality control, such as comparing human review with predictive coding results, and in reviewing production from the opposing party as well as during other stages of litigation, such as preparing for depositions, responding to summary judgment motions and working with expert witnesses. Some common features of predictive coding/TAR systems are concept searching, contextual searching, metadata searching (ESI must usually be produced in native format with the metadata intact), relevance probability and ranking, clustering, and sorting ESI by the issues and arguments in a case.

3 Methods

In order to design the interface, the first step was to conceptualize our ideas based on factual user data rather than grounding it only on technical assumptions. Although several vendors provide predictive coding/Technology Assisted Review (TAR) software as part of their overall electronic discovery services, we were especially concerned about developing a predictive coding system that is user-friendly and in keeping with proper usability principles for interface design. In setting our strategy, it was essential that our preliminary design be based on clear, unbiased feedback obtained from a passionate specialist within the field of e-discovery. Given that there are similar kinds of software already in the legal technology marketplace, we started our project by conducting research on several existing exemplary models for predictive coding, comparing their features and functionality and noting gaps in what users might require. From here, we identified our problem space.

4 The Problem Space

A typical predictive coding system uses analytics to assist lawyers in collecting, preserving, reviewing and analyzing the relevancy of the Electronically Stored Information (ESI) in a particular legal case. It is important to note that ESI can encompass a

wide range of documents and file types. In a manual system, the lawyer reviews all potentially relevant ESI by hand, classifies the ESI as relevant or not relevant, and indicates whether the ESI can be protected under one of the confidentiality doctrines. In predictive coding system, the lawyer reviews and develops a model with a preliminary set of documents that train the system. The machine routines this analysis model as its algorithm and categorizes the documents appropriately, thus drastically reducing the overall cost and time required to perform these phases of an e-discovery process, hopefully ensuring its accuracy and avoiding either failing to include potentially relevant ESI or inadvertently producing ESI that could and should be protected. One issue with existing predictive coding services is the user interface. These predictive coding services tend to focus more on yielding high throughput, but seem less concerned about the ease with which users interact with the system.

5 The Overall Framework

We can first look at the overall system architecture that we conceived of to acquire a deep understanding of the purpose of the user interface (Fig. 1).

The model accepts multimodal input, henceforth referred to as Electronically Stored Information (ESI) of any data format that can be stored in the repository by using the Hadoop Distributed File System (HDFS). The natural language processing component incorporates the supervised machine learning algorithms that support conceptual search. Conceptual search is an automated information retrieval method to search electronically stored, unstructured text that is conceptually similar to the

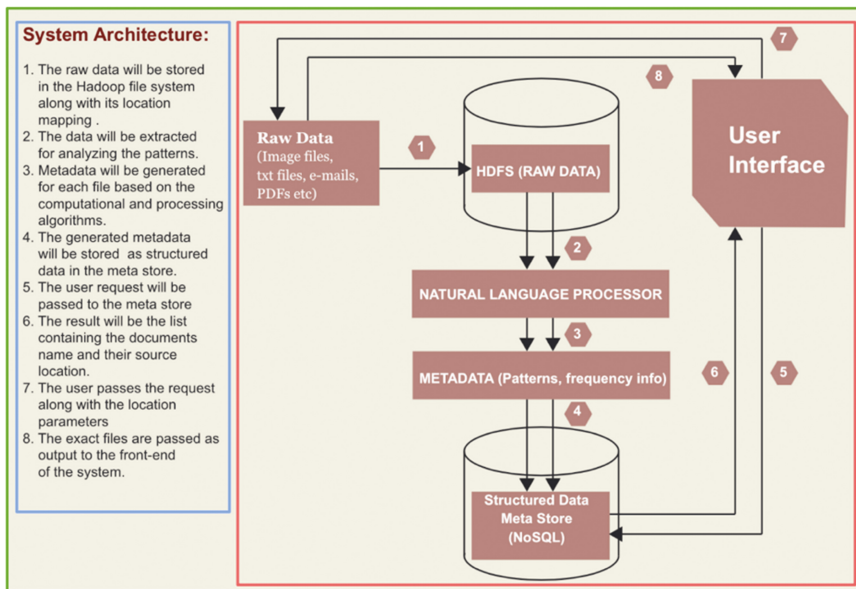


Fig. 1. Architecture of the system

information provided in a search query. A supervised learning algorithm promotes not only passive analysis of data, but also accepts a lawyer’s periodic input to enhance the accuracy and efficiency of the process. The robust algorithms for the predictive analysis are applied to this huge “raw data store.” The resulting metadata informs the frequency of a particular keyword for a file and is stored as structured metadata in a “meta store.” Communications between the “metadata store – interface” and the “interface – raw data store” are parameterized to support more reliable storage and retrieval of data from different servers since data in a law firm may be distributed and stored in multiple servers or even in the cloud.

A “Weka tool” generates a sample report that shows the recall and precision rate of passing an absolute training set to a model that uses classifiers to group files based on most frequently occurring common words in the files (Fig. 2).

An initial interface was developed only to verify that the basic functional needs for testing the model were met, rather than incorporating usability considerations (Figs. 3 and 4).

As outlined in the Background section above, even though there are many benefits to using predictive coding/Technology Assisted Review(TAR), lawyers still hesitate to use it because of their unfamiliarity with the technology. Thus, an optimal solution to this problem must focus on hiding technical complexities from users by designing an intuitive, attractive, easy-to-navigate interface that will give users comfort in having control over the system. To understand the key requirements of the system, one of the authors, who has considerable experience with e-discovery, provided extensive insights and feedback on the design and development process of the system over several months. In addition to the findings from our review of the literature and existing predictive coding services, we relied on books by Levy and Nunnally and Farkas as well as the Nielson Norman Group’s 10 Usability Heuristics for User Interface

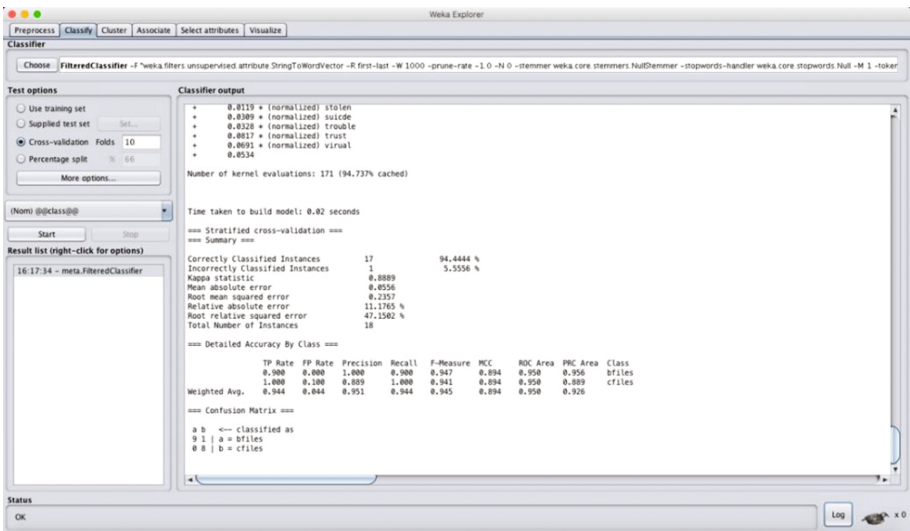


Fig. 2. Weka tool report



Fig. 3. Initial interface

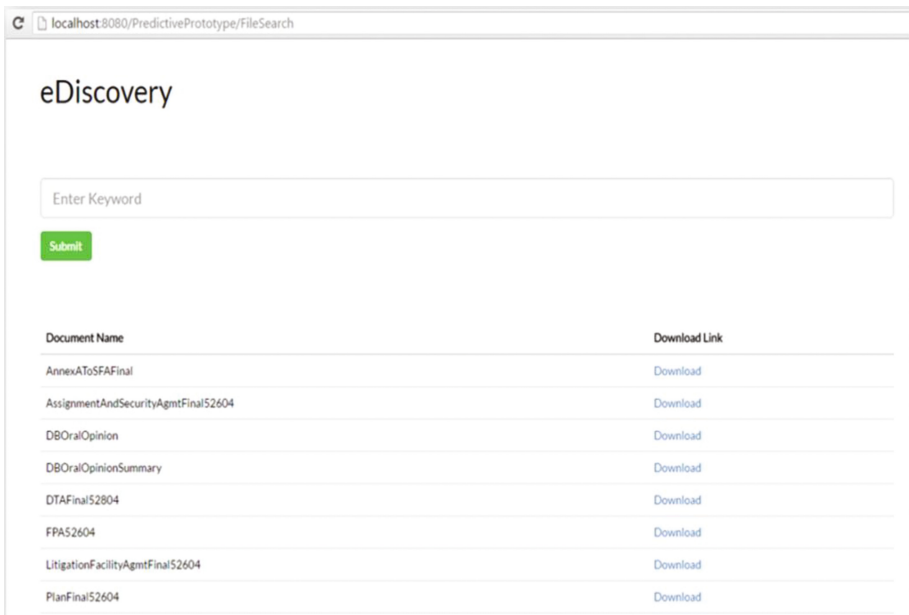


Fig. 4. Initial interface

Design [15–17] During the first phases of the development of our system, we were fortunate to be able to present our work at two local conferences to audiences with expertise in data mining, interaction design and usability testing.

The interface encompasses the following modules:

- Search
- Upload
- List
- Tag
- Inline tooltip

6 Persona

To understand the behavior of our prospective users and their mental models in interacting with each of these modules, we developed a persona of a law firm employee who would be working in e-discovery and followed this persona through a typical e-discovery scenario.

Marie is paralegal working for a law firm that concentrates its practice on general business law, bankruptcy and creditors' rights. She has a certificate in Paralegal Studies and several years of experience. She facilitates and manages e-discovery processes, under the guidance and supervision of the lawyers who are responsible for the cases she works on. Her roles may encompass educating clients on e-discovery policies, drafting litigation hold procedures for the lawyer's review and communication to clients, working closely with e-discovery teams to assess a client's electronically stored information (ESI), and helping to ensure compliance with federal and state court rules and the law firm's protocols for e-discovery. Her motivation is to use a predictive coding system to collect, preserve, analyze and review potentially relevant evidence from clients that has been generated by a variety of software and is stored in different file formats.

7 Scenario

Marie needs to help collect, preserve, analyze and review potentially relevant evidence for a bankruptcy case. She must identify a set of relevant files such as client's bank statements and details about the client's assets and liabilities, including real and personal property. She needs to retrieve a large set of the client's email transactions and social media information to increase the depth of the search. She will identify the most frequently used keywords from similar cases to accelerate the search process. There is a very short timeline for an e-discovery process, particularly because of the ambitious deadlines for each step in the process articulated in the 2015 amendments to the Federal Rules of Civil Procedure [3]. Marie also needs to consolidate and maintain a list of files and documents so that in future recalling this collection will not be a tedious process. The law firm must avoid transmitting certain documents and files to the opposing party because they may be protectable under attorney-client privilege, as attorney work-product or because of another confidentiality doctrine. Thus, downloading those documents should be restricted in order to avoid inadvertently including them in the set of ESI to share with the opposing party's lawyer.

Based on the presented scenario, we have identified design goals for each module that may be useful as part of Marie's efforts.

8 Discussion

We discuss the design goals for each module as follows. At this point in the design of the system, we relied extensively on Nielson Norman Group's 10 Usability Heuristics for User Interface Design [17].

8.1 Search Module

This module addresses the main purpose of the system, to search a file or files of the listed category. An “autocomplete suggestion widget” is a good choice for promoting usability because it guides the user in building the search query. To make it manageable, special attention is required in designing the autocomplete widget. Avoiding the scroll bars for the autocomplete search box promotes an optimized search. Including scroll bars in our system would make the search process more cumbersome, since it would require continuous manual scrolling on a user’s computer that most users dislike.

We plan to offer two types of categories in the search bar. These two options are machine-generated and user-generated categories. Differentiating the categories with different colors or styles aids the user in identifying the more prominent category (machine-generated category). Not only do the different types of categories require a color or style variation, but also the search key and the category need to be distinguished. The suggested categories need more focus than the user-typed search key, because users are already aware of the words they typed in the bar and thus it needs no highlighting. This reduces the cognitive overload for users. These variations provide a clear perception of the search suggestion by converging the user’s focus on the categories rather than the search key. Finally, the more important component is the widget label. It communicates the type of suggestion that the interface is providing to the user, averting unnecessary confusion for users in understanding autocomplete suggestions. The widget’s label should clearly mention the purpose of the user’s search.

8.2 Upload Module

This module has three options for users. They can upload a document or file, a selected list of documents and files or an entire folder of documents and files. To achieve optimal usability, users are provided with options based on their goals. In our scenario, one of the tasks for our user is uploading huge volumes of files and documents, selectively uploading a list of files and documents or bulk uploading of all of this material through folders. All three methods for uploading potentially relevant ESI are supported by allowing users to either select the files and documents from directories by meticulous clickable options or to “drag and drop” to upload the materials. In this way, we promote mapping real-world actions to the system and reduce the number of operations involved in uploading content to the repository.

Another feature of our system is the progress bar. The progress bar indicates the status of the loading process. It will not only indicate the status of the uploading process itself, but also the status of different procedures that are happening within the backend of the system while a file or document is being uploaded. The different stages that a file passes as it is entered into the repository are visible to users as icons with a green checkmarks over it to indicate that the file is moving from one stage to another. These icons help a user clearly understand the process that is happening behind the interface, thus encouraging users to trust the system and giving the user a greater sense of control. The four stages represented by the icons correspond to the backend

processes that we developed for the system. These designations are for ESI transferred across the security layers, ESI stored in the database, ESI categorized based on the computed model, and ESI encrypted and stored in the repository.

8.3 Tag Module

This module addresses the problems with organizing files and documents based on a user's preferences. By default, the system is programmed to be able to organize files and documents based on a training or seed set of files that have been input into the system through the initial model improvement progression. However, the system needs frequent interventions by the manual reviewers (lawyers or their paralegals) to calibrate the precision and recall rate of the model. The tag module takes care of this calibration process over time. It takes the input for calibration from all the users of the system who are collaborating on the same case, but at different times during the case. Users are adding files and documents to existing established categorization, increasing the precision rate, or generating a new category for the files, increasing the recall rate.

The "list" module has the tag module embedded within it. Users may select and tag an unlimited number of files and documents for future reference. To facilitate usability, users can either add a new tag or attach an existing tag to the file or document selected. The tag option is available for a file when it is selected using a checkmark. This supports "multi-file" tagging and "select file" tagging. Thus, the system spares users from redundant iterative monotonous tagging activity. A label's success depends on using it properly, so designing the tag module needs considerable care. Users will note the significance of the tag because tags will be displayed by the frequency of use or its latest use information. Both kinds of data inform the user how that tag is being interpreted and applied by other users, which influences the user's decision in choosing a tag for his/her selected files and documents. The more often a tag is used, the more chances for the system to learn that category and prepare the search hierarchy accordingly. If a user believes that he/she has selected or added a tag in error, an option to delete a tag allows the user to correct his/her error.

8.4 List Module

The list module has all the files and documents listed in order of choice preferred by the user. To achieve additional usability, users are given the freedom to order files and documents according to their needs in a case. Users should be able to sort based on relevancy, frequency of use, most recent and alphabetical order. Users might also require more sophisticated filtering options. They should be able to filter the files and documents based on their file structure, file type or protected or unprotected as confidential, as attorney-client privilege or as attorney work-product. The view of the list bar must be scalable. Users can be provided with the option to vary the number of files and documents that could be listed for a page. The system will provide a download option. Users can select files or documents to either download or tag. Thus, the system needs to provide users with the functionality to easily select files and documents. To facilitate this function, we included an option to select and de-select files and documents using checkboxes.

8.5 Inline Tooltip Module

This module ensures that the users are aware of every possible interaction that they could execute to attain their search and retrieval goals as seamlessly and effortlessly as possible with minimal need to consult system documentation. To achieve usability, note that although one aspect of good practice when designing an interface is that it must be self-explanatory, it is also good practice to provide users with proper documentation and training materials on how to use the system. Instead of confronting users with non-intuitive, immense documentation on how to use the features and functionality of a system, introducing inline tool tips motivates a new user to get started with the system without any struggles. The inline tooltip feature can be turned off, so that experienced users will not perceive it as an inconvenient element.

9 Results

After identifying the design requirements and features of the system, we began the initial design phase of the project by outlining a basic layout.

9.1 The Conceptualization

A user can begin with “suggestion widget” as a search bar to type in the keywords (terms, party names, dates) of the files or documents that he/she requires. The “suggestion bar” is activated when it senses that there is text in the text field of the “search bar.” The user chooses the particular category from the widget and clicks on the search button. The “list” module lists the files that are relevant or related conceptually to the keywords searched. Users can modify the list based on two sorting criteria – the relevancy of the keyword (more frequently used) and ascending/descending order of the file. Users can filter files and documents based on such attributes as whether an item is protected/privileged or not and file and format types. They have the option to mark a file as protected. They can click on the lock icon to protect it. The lock icon indicates the status of the file or document as either protected or not protected. The download option is available for all files except protected materials.

Tagging a file is permissible for all types of files and documents. The inline tool tips are activated when a user interacts with the system for the first time at the following functional points (Figs. 5 and 6).

1. When a user logs into the system, the tooltip greets the user and requests that the user type a keyword into the text field of the search bar.
2. It shows the user how to upload a file or document when the user clicks on the upload option.
3. Similarly, an inline tooltip is activated when the user hovers over the protected, sort, tag or download user interface (UI) controls.

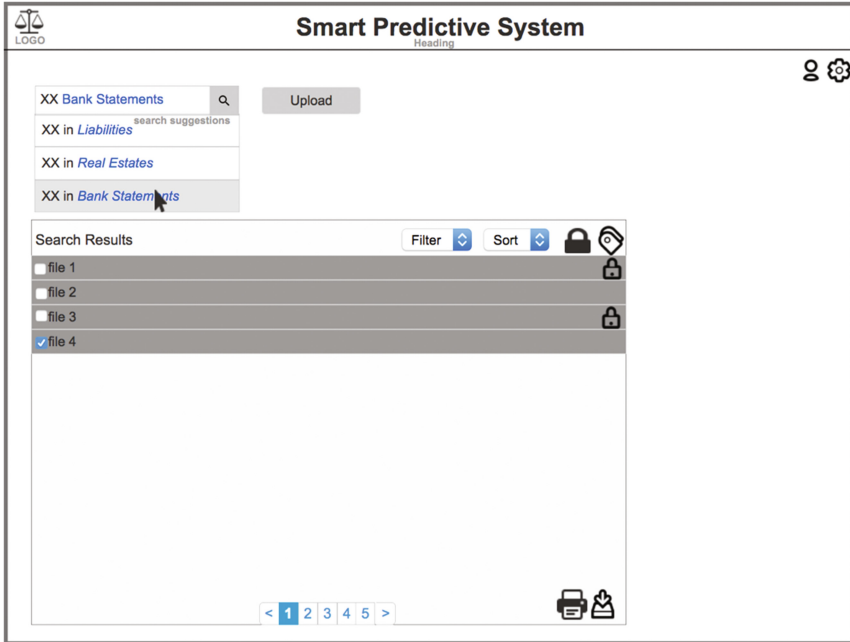


Fig. 5. Conceptual design of the “Smart Predictive System”

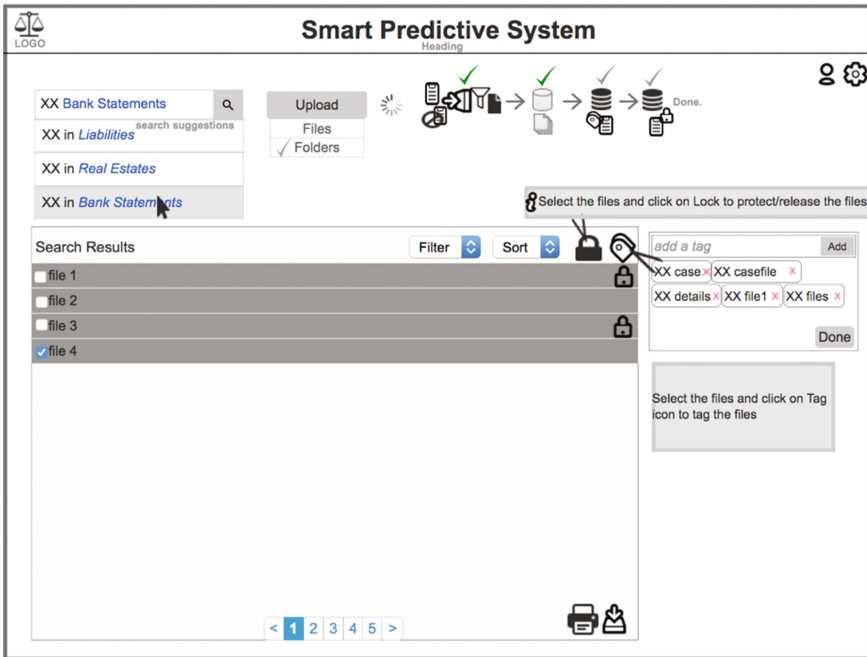


Fig. 6. Conceptual design of the “Smart Predictive System”

10 Future Work

Up to this point in the project, we have been working with a limited set of files and documents that we created, primarily files and documents that relate to bankruptcy law. In addition to testing and refining the system, particularly the backend processes, we need to gather a broader range of ESI, including files and documents from other areas of the law and materials that can be clearly identified as being those that need special protection due to confidentiality concerns. We also need to test a range of non-text-based items, because dealing with increasingly complex formats is another challenge to the e-discovery process and to predictive coding/Technology Assisted Review (TAR) services. Another step is to better integrate all of the modules and parts of our system, such as the database, the logic and algorithms, natural language processing and conceptual searching. Once the system is fully developed, we will test it again with a group of lawyers who practice bankruptcy law.

Some specific activities that will be part of our future work include the following.

10.1 Initial Evaluative User Testing

Even though the basic requirements of the system have been identified and conceptualized, in order to make the system more effective and efficient, additional insights into user preferences and the rationale for final design choices need to be validated before developing the low-fidelity prototype and testing the flow of a user's interactions with the system. To obtain feedback about a user's preferences for the modules, two variations of key modules will be shared with a group of users. This feedback will also reveal any features and functionality that we may have overlooked in our initial planning and design approaches. A survey will be created and follow-up one-on-one interview sessions with potential users will be conducted. The questions in the survey and interviews will attempt to understand a user's choice of one type of module over the other and the rationale behind it, except for the "list module." The design of the list module will be demonstrated to users to verify the accuracy of the reflections obtained as the result of the interview data synthesis phase (Fig. 7).

10.2 The Two Variations of the Autosuggestion Widget

The first variation has categories presented based on the most frequently used order. The second variation has the categories listed bas on alphabetical order.

10.3 The Two Variations of the Tab Module

The first variation lists the previously used tags with a number appended to it. The number indicates how often people have used that tag. The second variation lists the previously used tags with color intensity variation. The more frequently used tags have more intensity when compared with other less used tags. The color intensity is directly proportional to frequency of usage of that respective tag (Fig. 8).



(a)



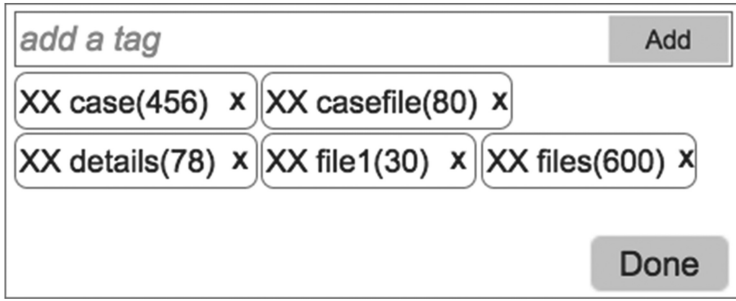
(b)

Fig. 7. (a) and (b) are two Variations of the Autosuggestion Widget

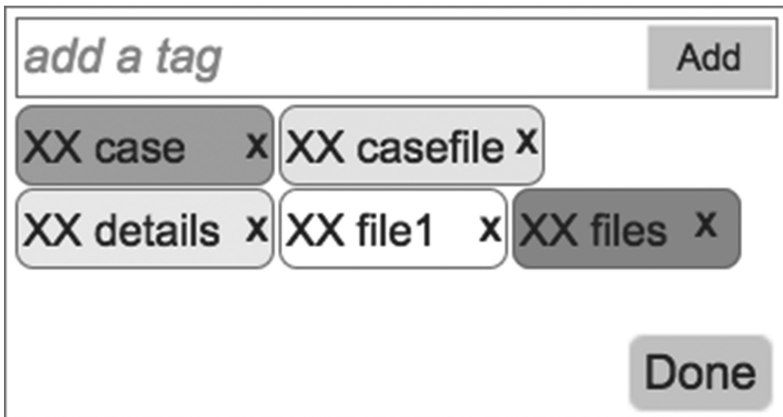
10.4 The Two Variations of the Upload Module

The first variation of the upload module has a normal progress bar indicating the status of the upload process whereas the second variation has different icons each indicating the progress, acknowledging to users the completion of each stage of the upload process that the files and documents pass through that is happening behind the interface of the system (Fig. 9).

After the final version of each module is determined, we will develop a low-fidelity prototype and test the flow of the interaction. After that, a high-fidelity prototype will be designed and a “think-aloud session” will be conducted to obtain direct input on how users actually perceive the system.



(a)

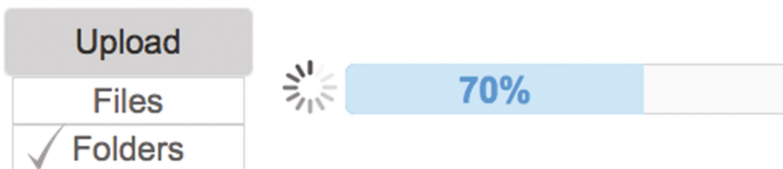


(b)

Fig. 8. (a) and (b) are two variations of the tab module



(a)



(b)

Fig. 9. (a) and (b) are two variations of the upload module

11 Conclusion

This paper presents the initial results of a pilot project to design a full-featured predictive coding system to collect, preserve, analyze and review potentially relevant evidence as part of an electronic discovery (e-discovery) process. Predictive coding/TAR systems have the potential to greatly streamline an e-discovery process, reduce the time and expense of the process and prevent the kinds of errors and omissions that could result in ethical violations for the lawyer. As part of our project, we have endeavored to use readily available software tools, natural language processing and other information search and retrieval capabilities, and best practices in interface design and usability. Future work includes refinement of the system and its individual modules based on user feedback and testing with a broader set of documents and file formats.

References

1. *Zubulake v. UBS Warburg*, 217 F.R.D. 309 (S.D.N.Y. 2003); *Zubulake v. UBS Warburg*, 216 F.R.D. 280 (S.D.N.Y. 2003); *Zubulake v. UBS Warburg*, 220 F.R.D. 212 (S.D.N.Y. 2003); *Zubulake v. UBS Warburg*, 229 F.R.D. 422 (S.D.N.Y. 2004)
2. ABA Model Rules of Professional Conduct. http://www.americanbar.org/groups/professional_responsibility/publications/model_rules_of_professional_conduct/model_rules_of_professional_conduct_table_of_contents.html, Accessed 8 Feb 2017
3. Federal Rules of Civil Procedure, Legal Information Institute, Cornell University Law School, 1 December 2015. <https://www.law.cornell.edu/rules/frcp>, Accessed 8 Feb 2017
4. Fuchs, J.L., McLean, C.G., Fiorentinos, I.S., FitzGerald, J.W., Heintz, A.F., Marino, E.J.: Noteworthy trends from cases decided under the recently amended federal rules of civil procedure, Jones Day, September 2016. <http://www.jonesday.com/noteworthy-trends-from-cases-decided-under-the-recently-amended-federal-rules-of-civil-procedure-09-06-2016/>, Accessed 8 Feb 2017
5. Hernandez, A.: Common problems with e-discovery – and their solutions. *The Federal Lawyer* **63**, 63–68 (2016)
6. Phillips, A., Godfrey, R., Steuart, C., Brown, C.: *E-Discovery: An Introduction to Digital Evidence*. Cengage Learning, Boston (2014)
7. Bethea, D.: Predictive coding: Revolutionizing review or still gaining momentum? In: White paper, *Litigation and Practice Support*, International Legal Technology Association, pp. 37–42 (2014)
8. *Da Silva Moore v. Publicis Groupe*, 287 F.R.D. 182, 183 (S.D.N.Y. 2012)
9. Hampton, W.M.: Predictive coding: it's here to stay. *E-Discovery Bulletin*, June/July 2014. https://www.skadden.com/sites/default/files/publications/LIT_JuneJuly14_EDiscoveryBulletin.pdf, Accessed 8 Feb 2017
10. Sutton, A.J.: Discovering discovery technology: a model order and pilot program for implementing predictive coding and other new technologies in document review. *AIPLA Q. J.* **42**, 459–492 (2014)
11. The Sedona Conference. *TAR Case Law Primer*, January 2017, <https://thesedonaconference.org/publications>. Accessed 8 Feb 2017

12. *Dynamo Holdings Limited Partnership v. Commissioner of Internal Revenue*, 2016 WL 4204067 (T.C. 13 July 2016)
13. Grant, D.: Seeing is believing: using visual analytics to take predictive coding out of the black box. *LegalTech News*, pp. SS1–SS4, December 2015
14. *Hyles v. New York City*, 10 Civ. 3119 (S.D.N.Y. 1 August 2016)
15. Levy, J.: *UX Strategy: How to Devise Innovative Digital Products That People Want*. O'Reilly Media, Sebastopol (2015)
16. Nunnally, B., Farkas, D.: *UX Research: Practical Techniques for Designing Better Products*. O'Reilly Media, Sebastopol (2016)
17. Nielsen, J.: 10 Usability Heuristics for User Interface Design. Nielsen Norman Group, 1 January 1995. <https://www.nngroup.com/articles/ten-usability-heuristics/>, Accessed 8 Feb 2017