IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661, p-ISSN: 2278-8727, Volume 27, Issue 5, Ser. 3 (Sept. – Oct. 2025), PP 01-06
www.iosrjournals.org

Indexing The Enormous Legal Documents To The Aid Of Tech-Savvy Lawyers

Dr. V. Annapoorani
Professor/MCA
Paavai Engineering College, Namakkal, Tamilnadu.

Abstract
This research concentrates on indexing the enormous legal documents to the aid of Tech-Savvy lawyers. Why corporate lawyers are ready for technological innovation? All the budding lawyers are moving towards technology or in other words Tech-Savvy. Undeniably, technology has revolutionized the business world, rapidly changing and expanding in every field imaginable. When it comes to the legal services industry, technological innovation is no exception. Technological innovation as a means of creating more efficiency has been steadily emerging across many areas of the law. The indexing process takes a group of document files and produces a new index. Document clustering is one of the imperative techniques for organizing documents in an unsupervised manner. So that, in this paper focus an automatic indexing method for providing an indexing in an alphabetical order from a enormous legal documents in clock structure to the aid of Tech-Savvy lawyers.
Keywords: *SSARC, PDDP, SLIA, TFIDF, LSI, NLU, HTML*

Date of Submission: 01-10-2025 Date of Acceptance: 11-10-2025

I. Introduction

Over ten years later, if a question arise on most likely change agent in the legal market, technological innovation was chosen by nearly 50% of the law firm leaders. On proceedings side, up-to-date innovation has been focused that is in case of eDiscovery tools and software, while reviewing emails and other digital records, time and cost savings were enabled. But when it comes to transactional work, the level of innovation has not been evenly distributed. The reason is that the binary analysis was obtained with the databases of email, coding responsive or non-responsive documents during the advance with eDiscovery verses due diligence. Based on the coding of document division by lawyer, Artificial Intelligence (AI) tools can be learnt and is applied to the remaining contents in that document. Corporate framework wind their way throughout a contract to be extracted and summarized for a vast number of highly varied documents due to its attentiveness, complex provisions.

Indexing

Automatically full-text index is created and for every document in a database, computer software ready every word. An inverted index of words was created with their locations in the database. With any words, the database is searched by the end-users where the computer finds every match between the search terms and the text of the documents. The document is located by the full-text searching and the users finds a high number of irrelevant items when users are not exactly sure what they need. Large amount of time retrieving documents are identified with previous paper-based system in databases searches, photocopying the documents, delivering the copies to attorneys and legal assistants and refiling the originals. The clerks also spent considerable time searching for misfield originals.

Legal Benefits

Litigation protection, in a law suit, records need to be produced very quickly. For lawsuit, an indexing system is used to identify and retrieve documents to pay for itself. Response to Rule 26. In a federal lawsuit, parties are required by a new law to identify and produce relevant records within 85 days of the beginning of the litigation. Also required a quick and accurate retrieval of records. Parties are not excused from acquiescence by the disorganization of records. To indexing system, new documents are added and all users accessed them immediately if documents are indexed and created. End-users can do their jobs better.

In this research work, we are sorting the civil case labels in alphabetical order by using spontaneous sorting clock method. This clock, collects all the legal documents from the web documents to sort and store the documents inside the clock division. Our clock's design is similar to the normal clock but using alphabets A to Z instead of the number 1 to 12. These Legal learning represents knowledge as logical rules that ordering and performs reasoning on these rules to search for proofs in civil cases. Proofs can be compiled into more complex

rules to solve problems with a small number of searches in the documents that they required. Figure 1 shows the Spontaneous Sorting Clock Method.

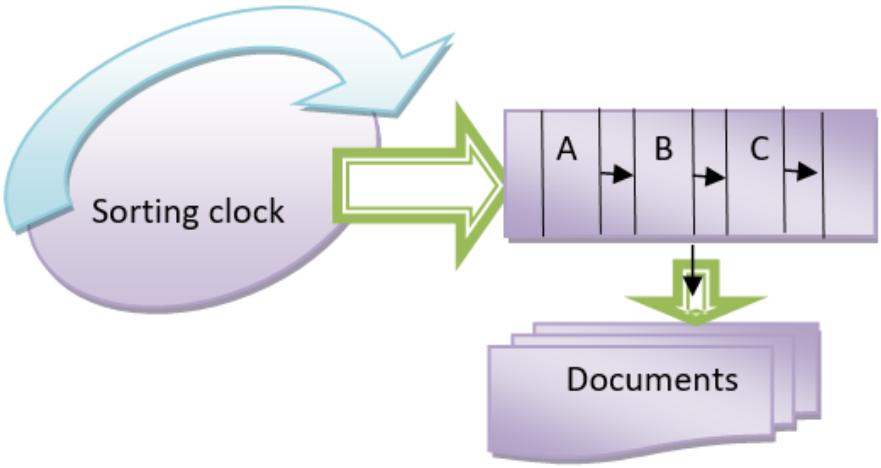

Figure 1 Spontaneous Sorting Clock Method

The sorting is based on the circular linked list in data structure in which each node is an alphabet present in the list and starting name of the legal documents the 'Aa' is collected by the first node. The index of civil cases are noted in this linked list of an each node for space saving. The link is present at each index that holds the entire details of particular case.

Partitioning document directive model

For cluster, each of the sorted document is voted so that the related data is ordered linked the list that has been assigned to continuous legal ruled data. In keyword extraction phase, based on cluster value a document analyses is obtained and is normalized on the sum of the weights of the chosen keywords.

Assuming correlating each document to one cluster is too strong and to smooth this, voting is done by arranging the clusters in descending order and associate the document to the first three clusters in this ranking. Thus off-line preprocessing data generative model is closed in civil case which partition the collected whole document to different subareas.

Hierarchal approach of document clustering

Hierarchical clustering (shown in figure 2) can further be classified as agglomerative or divisive. An agglomerative method starts with each document representing a single cluster. This is a bottom up approach to clustering. In divisive approach all observations start in one cluster and continue until each object is in a separate group. It is a top down approach to clustering.

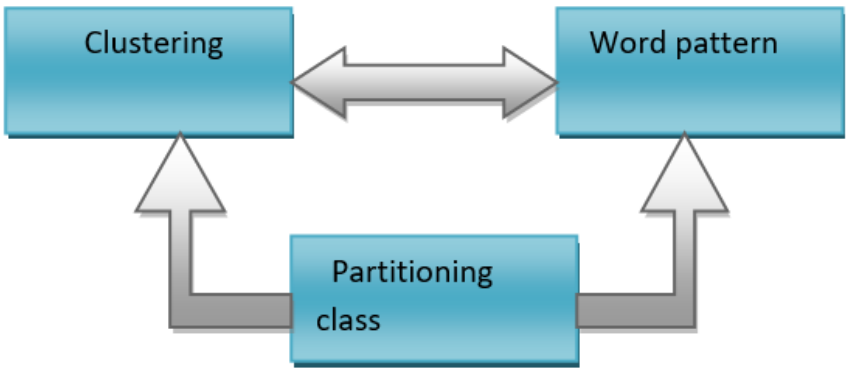

Figure 2 Clustering Diagram

PDDP is hierarchical clustering algorithm for clustering documents. The output of clustering is hierarchy comprising of similar documents in a groups or cluster. The method is based on principal component analysis and returns a tree with nodes of clusters. Each cluster comprises of similar documents. The input documents are represented as a matrix. The vector space model of information retrieval is used for representing

the input document corpus. The hierarchical clustering method returns a binary tree in which each node is either a leaf node or further splits into child nodes.

It is an agglomerative method of hierarchical clustering. In this method the distance between two clusters is the distance between two closest data points in the two clusters. The drawback associated is that it is sensitive to noise and may result in chaining effect. It merges the two clusters in each step with the smallest minimum pair-wise distance.

PDDP unsupervised learning

The PDDP algorithm is an unsupervised top-down clustering algorithm that has been shown to be useful for exploring large datasets such as web-based text document [3, 2]. Preprocessing: Preprocessing is the initial step to clustering. It includes:

- stop word elimination,
- stemming and Converting the documents into Term-document matrix.

Stop word elimination Stop words are words in a text document whose frequency is high but low semantic weight. Generally articles and prepositions fall in this category. Such words can be eliminated and is an essential step for large document corpus. Stemming It is the process that reduces the variant forms of a word to a common form, It is a process of linguistic normalization. Term document matrix One of the essential considerations in document clustering is the representation of documents in a form suitable for further processing or clustering process. It is required to encode the documents in some numeric form in order to apply computational methods. The solution is a term-document matrix. A vector space model solves the problem by representing text documents as a matrix. Clustering contact annotations based on personal issues on word comparison. To summarize the steps in the overall sorting clock, showing how the various methods just described are assembled into an overall sorting clock.

Word pruning criteria

Some of the issues that had to be addressed in order to apply the methods chosen to the data were: scaling the data, selection of stop words to be removed from the data, the stopping test to be used, and the word pruning criterion. Ambiguity of natural language is overcome by the PDDP.

Context of the discourse permits to select most appropriate text and well-known lexical taxonomy is used to proceed this approach where its extension to deal with domain categories, as a background knowledge. Figure 3 shows the architecture for document analyze model.

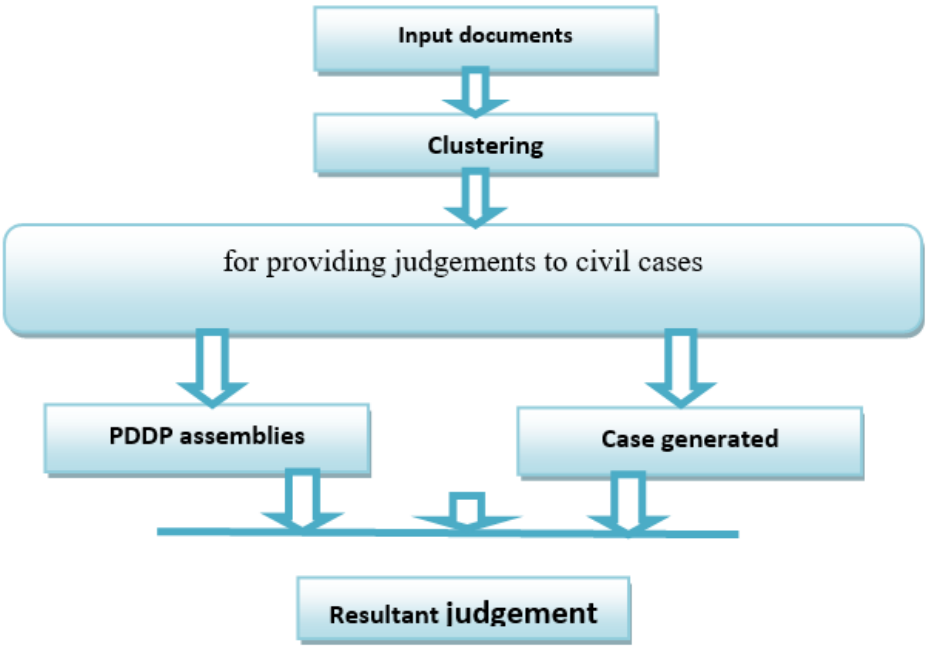

Fig3 Document Analyze Model Architecture

Algorithm states that entire set of documents are partitioned into two by its principle direction. These two partition is then again divided into two sub partitions by repeating same process recursively. The obtained result is arranged as “PDDP Tree”. Here the partition is in the form of a leaf node or forms two children with two sub partitions.

Frequency analyze model Document scaling.

Definition of a node in the binary PDDP tree for a cluster with P documents. The root node contains all m documents, each lengthen. The data could be uncalled, scaled with a TFIDF (Term Frequency and Inverse Document Frequency) scaling. Hence, to obtain such synsets, we need to compute for each document the prevalent domain. Past experience with PDDP has shown that TFIDF scaling does not add much more accuracy than the simpler document length scaling. Thus, each domain takes as weight the sum of all the weights of synsets associated to it, which results in a ranking of domains by decreasing weight.

Classical vector document pattern.

Ranking algorithms based solely on the words are used for the retrieval of a classic information. With one dimension per term, a high-dimensional vector space is safe. In vector space, vector is defined as the document or query. Entries occurred are positive and entries of terms not occurring in the document key part of the personalization process which is based on the observed patterns, and resulting probabilities. Figure 4 shows the user interface.

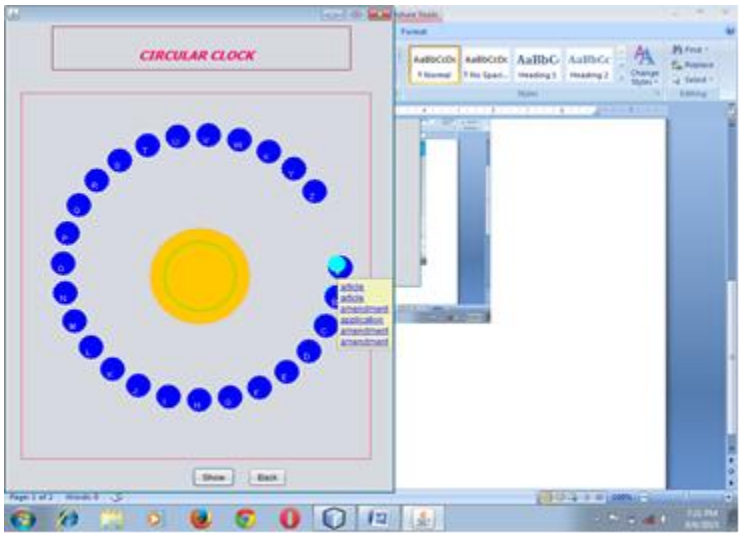

Figure 4 User interface

II. Experimental Results

Semantic lexicon is built for a real application for the beneficial purpose of proposed SSARC algorithm. Other classes represent dates (time), locations (location) and types of cases (case). Set of pit words are needed for proposed algorithm for each semantic class. Two common conditions are needed to choose seed words:

1. In domain, word is to be frequent and it ensures the occurrence of the word in the corpus.
2. To reduce the risk of identifying irrelevant contexts around the word, clear understand of a word is necessary.
3. Five seed words are defined for each eleven semantic classes which is shown in table present in figure 5.

To create index, proposed method is compared with PDDP, Semantic Lexicon algorithms and SSClock. Precision, recall and F-measure are used for measuring the results which are evaluated using the adherent equation 1.0.

Performance Measures

Two different performance measures are used to calculate the comparative quality of the clustering. These measures were scatter and entropy.

SCATTER

Scatter S_c of a cluster M_c is defined as shown in equation (1):

$$S_c = def \sum_{j \in c} (X_j - W_c)^2 = \| M_c - W_c eT \| f^2 \qquad (1)$$

W_c - means the cluster, e - mdimensional vector $[1,1,.....1]^T$ $\| \ \|_f$ is the frobenius form. For some nxm matrix A, the frobenius norm of A is $\| A \|_f = (\sum 1 <= i <= n, 1 <= j <= ma^2 i, j)^2$ where $a\ i, j$ the entry in the ith row and jth column of A. Good cluster quality is obtained when the scatter value is low.

Entropy

Entropy as a measure of goodness of the clusters and when each cluster contains exactly one document then attain the best entropy. Entropy value is 0.0 when a cluster contains documents from one class only. The values of entropy of the cluster is higher when cluster consists of documents from many different classes.

Total entropy is calculated as the weighted sum of entropies of the clusters. The entropy e_j of cluster j is defined by e_j def - $\sum_i c(i,j).\log(C(i,j)/\sum_i c(i,j)$ where $C(i,j)$ is the number of times label i occurs in cluster j. Summary of results for various methods are shown in figure 5.

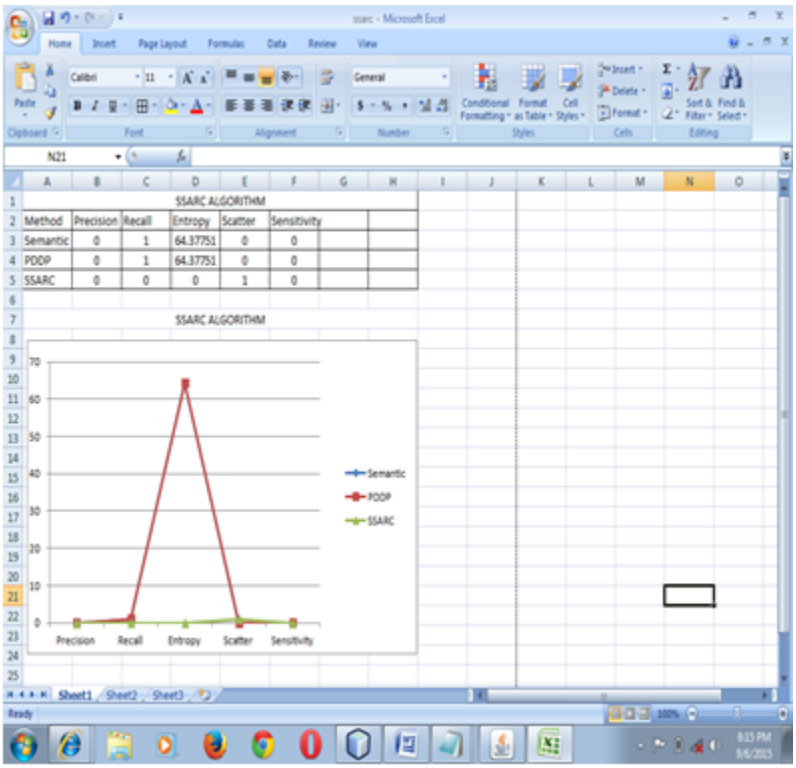

Figure 5 Summary of results for various methods

III. Conclusion And Future References

In this 21st century, we are surrounded by huge amounts of large scale raw data that is awaiting to be processed. The initial step in processing any data is Indexing and it makes handling huge data easier. Law and Order is one of the oldest departments in the history of mankind and it has huge amounts of data to be processed and Indexed. SSARC helps in indexing these enormous data swiftly and with ultimate ease. Indexing legal documents does not only help young attorneys but also helps the experienced practitioners who are moving forward towards technological developments. In experiment, Semantic, PDDP and SSARC method are used and among them entropy of Semantic and PDDP performs better than SSARC.

References

[1]. Koniaris, M., Anagnostopoulos, I., & Vassiliou, Y. (2016, November). Multi-Dimension Diversification In Legal Information Retrieval. In International Conference On Web Information Systems Engineering (Pp. 174-189). Springer International Publishing.
[2]. Boley, D. (1998). Principal Direction Divisive Partitioning. Data Mining And Knowledge Discovery, 2(4), 325-344.
[3]. Krallinger, M. (2015). Development, Application And Evaluation Of Text-Mining Methods For Biomedical Literature Processing: From Document Categorization To Gene Ranking.
[4]. Soderland, S., Fisher, D., Aseltine, J., & Lehnert, W. (1995). CRYSTAL: Inducing A Conceptual Dictionary. Arxiv Preprint Cmp-Lg/9505020.
[5]. Riloff, E. (1996). An Empirical Study Of Automated Dictionary Construction For Information Extraction In Three Domains. Artificial Intelligence, 85(1), 101-134.
[6]. O'Neill, J., Privault, C., Renders, J. M., Ciriza, V., & Bauduin, G. (2009, June). DISCO: Intelligent Help For Document Review. In Global E-Discovery/E-Disclosure Workshop–A Pre-Conference Workshop At The 12th International Conference On Artificial Intelligence And Law, Barcelona, Spain.
[7]. Vo, D. T., & Ock, C. Y. (2015). Learning To Classify Short Text From Scientific Documents Using Topic Models With Various Types Of Knowledge. Expert Systems With Applications, 42(3), 1684-1698.
[8]. Liu, Y., Mostafa, J., & Ke, W. (2007). A Fast Online Clustering Algorithm For Scatter/Gather Browsing. Chapel Hill, NC, USA: UNC School Of Information And Library Science., Tech. Rep. TR-2007-06.

[9]. Pletscher-Frankild, S., Pallejà, A., Tsafou, K., Binder, J. X., & Jensen, L. J. (2015). DISEASES: Text Mining And Data Integration Of Disease–Gene Associations. Methods, 74, 83-89.
[10]. Lehnert, W., Mccarthy, J., Soderland, S., Riloff, E., Cardie, C., Peterson, J., ... & Goldman, S. (1993, August). Umass/Hughes: Description Of The CIRCUS System Used For MUC-5. In Proceedings Of The 5th Conference On Message Understanding (Pp. 277-291). Association For Computational Linguistics.
[11]. Conrad, J. G., Al-Kofahi, K., Zhao, Y., & Karypis, G. (2005, June). Effective Document Clustering For Large Heterogeneous Law Firm Collections. In Proceedings Of The 10th International Conference On Artificial Intelligence And Law (Pp. 177-187). ACM.
[12]. Amoli, P. V., & Sh, O. S. (2015). Scientific Documents Clustering Based On Text Summarization. International Journal Of Electrical And Computer Engineering, 5(4), 782.
[13]. Miikkulainen, R. (2000). Text And Discourse Understanding. In Handbook Of Natural Language Processing. CRC Press.
[14]. Abbey, R., Diepenbrock, J., Langville, A., Meyer, C., Race, S., & Zhou, D. (2012). Principal Direction Gap Partitioning (PDGP).
[15]. Littau, D., & Boley, D. (2006). Clustering Very Large Data Sets With Principal Direction Divisive Partitioning. In Grouping Multidimensional Data (Pp. 99-126). Springer Berlin Heidelberg