Analyzing the Extraction of Relevant Legal Judgments using Paragraph-level and Citation Information

K. Raghav [1] and **P. Krishna Reddy**[2] and **V. Balakista Reddy** [3]

Abstract. Building efficient search systems to extract relevant information from a huge volume of legal judgments is a research issue. In the literature, efforts are being made to build efficient search systems in the legal domain by extending information retrieval approaches. We are making efforts to investigate improved approaches to extract relevant legal judgments for a given input judgment by exploiting text and citation information of legal judgments. Typically, legal judgments are very large text documents and contain several intricate legal concepts. In this paper, we analyze how the paragraph-level and citation information of the judgments could be exploited for retrieving relevant legal judgments for the given judgment. In this paper, we have proposed improved ranking approach to find the relevant legal judgments of a given judgment based on the similarity between the paragraphs of the judgments by employing Okapi retrieval model and citation information. The user evaluation study on legal judgments data set delivered by Supreme Court of India shows that the proposed approach improves the ranking performance over the baseline approach. Overall, the analysis shows that there is a scope to exploit the paragraph-level and citation information of the judgments to improve the search performance.

1 Introduction

In recent years, there has been a significant increase in the availability of data in digital domain which has led to data explosion in different domains. The legal domain is one such domain with a huge amount of data generated on a regular basis which has led to the problem of information overload. Certain rules and regulations are incorporated by the state in order to protect the interests of the people and it can be observed in the form of constitutional rules and regulations or acts. In the modern world, legal issues are prevalent across the globe and a huge number of cases are registered every day across the world. Developing efficient retrieval approaches to improve the accuracy of search systems in legal domain is a research issue.

Civil law and common law are the two main systems of law. The major difference between civil law and common law lies in the main source of law. In civil law systems, codes and statutes are provided based on which decisions in the court are delivered and previous legal decisions have little role to play. Unlike civil law, in common law systems, previous decisions are regarded as the most important source of law. The statutes and the judgments (or precedents) play an important role in determining the outcome of a proceeding in the common law based legal systems. Precedent or previous judgment is an important concept in the common law based systems where subsequent cases with similar issues or facts use the precedents.

In the web domain, efforts have been made for efficient search and retrieval of relevant web pages. Text and link-based (hyperlinks) information of the web pages have been exploited effectively to develop better search and retrieval approaches. In the literature, efforts [6, 7, 22] have been made to exploit the text of the judgments for query-based retrieval of legal judgments. In addition, link-based information in the legal data has also been exploited for similarity analysis of legal judgments.

We attempt to solve the problem of ranked retrieval of relevant legal judgments for a given input judgment in the legal domain. The motivation is as follows. Common law system is largely based on the doctrine which is implicit in court decisions, customs, and usages, rather than on codified written rules. This reliance on the precedence by the legal system makes it critical for legal practitioners for studying older cases to analyze how the issues related to the current task were discussed and ruled in preceding cases [27]. Therefore, it is required for legal practitioners to get the updates about the latest 'legal concepts' which may help to prepare the arguments. In general, in order to explore a 'legal concept', a legal practitioner starts browsing legal database using her/his knowledge and experience. After retrieving one or more seed cases, she/he starts looking for more judgments similar to those seed judgments for a detailed analysis of the 'applied legal concept' in those judgments. (An applied legal concept refers to a specific legal interpretation, accepted under 'facts' present in a case.)

A legal judgment is a text document which contains the formal choice made by the court. In addition to the text-based information, the legal judgments delivered under common law system contain links to other judgments known as citations. Citations are similar to the references in research articles. Essentially, legal judgments are large and complex text documents embedded with various legal concepts at a granular level. When comparing the legal judgment as an entire text document for retrieval, relevance cannot be captured effectively between the judgments due to the huge length of the judgments.

In this paper, we analyze how paragraph-level and citation information of the judgments could be exploited for the extraction of relevant legal judgments of a given judgment. We have proposed improved ranking approach to find the relevant judgments of a given judgment based on the similarity between the paragraphs of the judgments by employing Okapi retrieval model and citation information. The experimental results on the real world dataset containing judgments delivered by Supreme Court of India show that by exploiting the paragraph and citation-based information in the judgments im-

[1] Kohli Center on Intelligent Systems (KCIS), IIIT-Hyderabad. email: raghav.k@research.iiit.ac.in
[2] Kohli Center on Intelligent Systems (KCIS), IIIT-Hyderabad. email: pkreddy@iiit.ac.in
[3] NALSAR University of law, Hyderabad. email: balakista@gmail.com

proves the performance of retrieving relevant legal judgments.

The rest of the paper is organized as follows. In Section 2, we present the related work. In Section 3, we explain about the structure of the judgment. In Section 4, we provide an overview of the background approaches. In Section 5, we present the proposed ranking approaches. In Section 6, we present the experimental results. Section 7 contains the discussion. In the last Section, we present summary and conclusions.

2 Related Work

In the legal domain, research efforts have been carried out in the following categories: query-based retrieval of relevant legal judgments, updating and summarizing of legal judgments, citation-based analysis. We discuss the related approaches in legal domain after discussing the related approaches in the text domain.

Vector space model [19] is a popular approach to model documents and cosine similarity method is employed to compute the similarity. Traditional methods to compare two documents treat documents as bag-of-words where each term is weighted according to TF-IDF score [25]. Okapi retrieval model is a probabilistic retrieval model introduced in [18]. Okapi BM25 is a weighting scheme which includes term weighting factors and a normalization factor together. Salton et al. [20] modeled the document as a collection of concepts and improved the search performance by considering the paragraph level information of the documents.

Link information in the documents has been exploited in the literature to improve search performance. Link information in the web pages is exploited in [4, 10] to propose better ranking approaches. Efforts have been made in [9] for comparing scientific articles by using the link information. Senthilkumar et al. [23] have proposed a method that can quantify the quality of citations based on the assumption that each research article would have different quality of citations.

In the legal domain, Turtle et al. [26] survey various procedures for retrieving legal documents according to their relevance to the query. They present retrieval models which consider the text information in the judgment documents and find matching between user query and judgments in the corpus. In their survey, they present existing retrieval models based on text information in the judgments. Boolean models, vector space models, and probabilistic models are the some extensively used models for computation of relevance between the user query and judgments to be retrieved. Chen et al. [6] have introduced an approach to assist the general public in retrieving the most relevant judgments using ordinary terminology or statements as queries which could be more useful for a non-domain user. Silveira et al [24], through a case study, have attempted to improve the ranking of search results by matching and combining the results of simple queries to concepts in a judicial domain-specific thesaurus. The work aimed at query understanding and extension by using the concepts in the thesaurus. Paulo et al. [16] have proposed an architecture for building knowledge bases on Portuguese legal data by developing an NLP-based legal ontology. Huu et al. [7] have attempted to address the retrieval of legal judgments for input user query by utilizing ontology-based frameworks. Maxwell et al. [14] have supported that legal concepts, issues, and factors play an important role in legal information retrieval system using NLP-based techniques. In addition, efforts have also been made for clustering legal judgments [13] in order to group judgments into sets of similar judgments. By considering the scenario of Indian legal judgments, Saravanan et al. [22] tried to retrieve legal judgments for input user query based on ontological frameworks. Saravanan et al. [21] also proposed summarization approach identifying rhetorical roles in the judgments and using graphical models. N B Bilgi et al. [3] have made efforts to build legal knowledge based expert systems using rule-based graphical design approaches.

Padmanabhan et al. [15] have proposed a ranking algorithm in which weights are assigned to edges of the clustered document graphs to capture the semantic information of the documents. In their approach, they try to utilize the link information to find important legal entities and documents. Agosti et al. [2] have introduced an information retrieval model for extracting relevant legal judgments for input query by using a two-level architecture. They focus on combining hypertext and IR approaches for extracting relevant documents and exploring those documents using a browsing technique. Casellas et al. [5] have proposed a method for constructing ontological graphs in which the nodes representing the sentences of the legal documents for the query and FAQs, and compares these graphs to perform retrieval of FAQs for search system and case law browser for the Spanish judges.

Efforts have been made to exploit the link information for analyzing similarity between sample pairs of judgments in [11, 12]. An effort has been made in [17] to find similar legal judgments through cluster analysis by exploiting the text and citation-based information.

The preceding approaches were aimed at developing systems to retrieve relevant legal judgments using information retrieval approaches. It can be noted that their main focus was to retrieve legal judgments based on the input user query. The existing ontological frameworks have adopted various keyword-based search mechanisms for retrieving relevant documents based on the user input query. The paragraph-level and citation-based approaches have been proposed to compute the similarity among the documents and cluster the documents. In this paper, we have made an effort to analyze how paragraph-level and citation information could be used to extract ranked list of relevant legal judgments for a given input legal judgment.

3 Structure of the Judgment

In this section, we explain the components and citation information of judgments by considering the example of Indian legal judgments.

3.1 Components of a Judgment

A legal judgment is a text document which contains the formal decision made by the court. A sample judgment delivered by Supreme Court of India is shown in Figure 1. The following are some of the important components of a legal judgment [12].

- *Judgment Title* contains the Petitioner and Respondent of the judgment along with the date of the decision.
- *Petitioner* is a person or an organization presenting a petition to the court.
- *Respondent* is an entity against which/whom an appeal has been made to the court.
- *Unique Id* is the unique reference given to the legal judgment by a reporter (law report) for further reference. Different law reporters provide different IDs to the judgments. Supreme Courts Reporter (SCR), Supreme Court Citations (SCC) and All India Reporter (AIR) are some of the law reporters which provide unique ids for judgments delivered by Supreme Court of India.
- *Citator Info* is the information regarding the judgments which refers the current judgment. Here the Unique Ids of the judgments

which refer to the current judgment in their judgment text are provided.

- *Act* indicates the category of the judgment. In this section, the basic issue discussed in the judgment is categorized in a legal point of view. It provides information regarding the acts under which the current legal dispute falls into.
- *Headnote* section of a judgment contains a brief summary of the judgment. Legal judgments are long text documents with a large amount of text and legal information and hence the headnote section of the judgments provides a summary of the most important concepts in the legal judgment for providing better readability to the users.

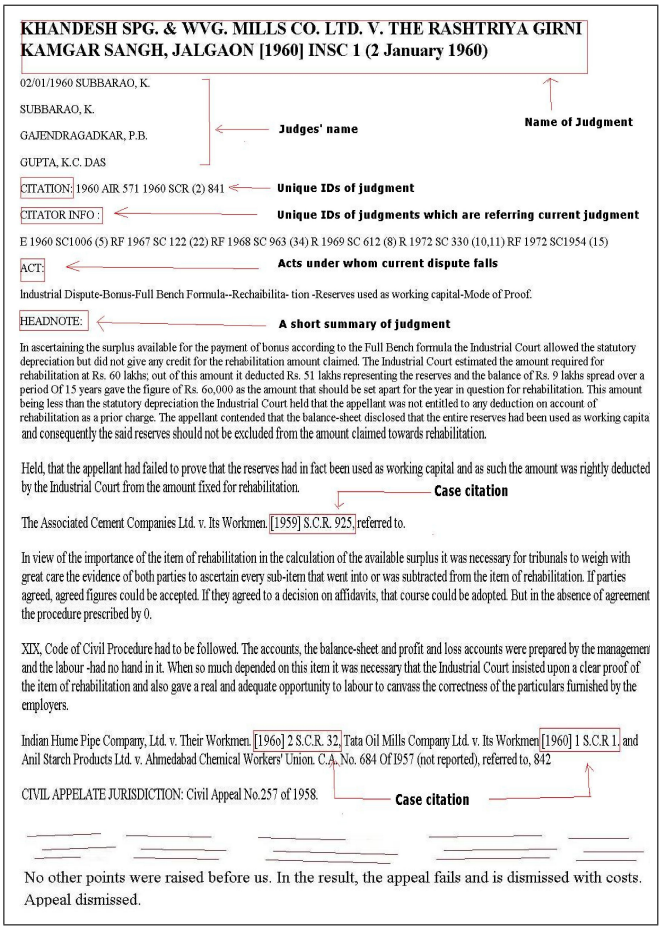


Figure 1: Sample judgment delivered by Supreme Court of India

- *Citations* are the external references made to other judgments to deliver the current judgment. The text section of the judgment contains the citation information where precedent legal judgments which discuss the topics of current legal judgment are referred by using the unique ids of the judgments. For example, in the Figure 1, "[1960] 2 S.C.R 32" is an example of a citation.

3.2 About the Citation Information

A citation network can be formed by considering the judgments as nodes and the linkage between the judgments as edges. Suppose a judgment J_m refers to another judgment J_n then judgment J_m is referred as an out-citation to J_n and J_n is referred as an in-citation from J_m. An instance of citation network is shown in Figure 2. The sample network shows a directed graph of four judgments, judgment J_1 has J_3 and J_4 as out-citations and judgment J_4 has J_1 and J_2 as in-citations.

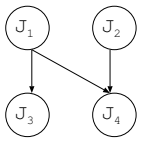


Figure 2: Depiction of citations

4 Background Approaches

Modern search engines take a user query as input and provide a ranked list of relevant documents by searching among a huge collection of documents. Relevant documents are retrieved for an input query by using different types of popular similarity measures like vector space and probabilistic models.

In the literature, several efforts have been made for defining the similarity measures using various term weighting approaches. TF-IDF based weighting and document length normalization are the standard practices in computing similarity using the term vectors. Traditionally, TF-IDF based term weighting and cosine distance measure were the most widely used approaches which do not involve document length normalization. With the advent of large data sets with highly varied document lengths, document length normalization was found to be quite important and hence the TF-IDF weighting represented by the pivoted normalization and Okapi are more increasingly used in current retrieval approaches. Using both the approaches of pivoted normalization and Okapi, the dot product has been found to be more appropriate than the cosine measure [8, 25]. We briefly explain the pivoted and okapi retrieval formula.

4.1 Pivoted normalization

The pivoted normalization retrieval formula [25] is a standard query based document retrieval formula. The pivoted normalization retrieval formula as given in [25] is as follows:

$$Sim(Q,D) = \frac{1}{(1-s)+s\times\frac{|D|}{AvgDl}\times|Q|} \times \sum_{t\in Q,D}(1+\log_e f(D,t)) \times \log_e(1+\frac{N}{N_t}) \quad (1)$$

where, Q is the input query, D is a document in the corpus, $|D|$ is the document length, $AvgDl$ is the average length of all the documents in the corpus, $|Q|$ is the length of input query, s is the slope parameter, t is a term in the document and query, $f(D,t)$ is the frequency of term t in document D, N is the number of documents in the corpus, N_t is the number of documents in which term t occurs.

4.2 Okapi

Probabilistic information retrieval approaches are based on the concept of documents should be ranked based on the probability of relevance to input user query. Okapi BM25 probabilistic retrieval model formula was proposed in [18] as follows:

$$Sim(Q,D) = \sum_{t\in Q} \frac{f(D,t).(k_1+1)}{f(D,t)+k_1(1-b+b\frac{|D|}{AvgDl})} \times log\frac{N-N_t+0.5}{N_t+0.5} \quad (2)$$

where, Q is the input query, D is a document in the corpus, $|D|$ is the document length, $AvgDl$ is the average length of all the documents

in the corpus, $|Q|$ is the length of input query, t is a term in the document and query, $f(D,t)$ is the frequency of term t in document D, N is the number of documents in the corpus, N_t is the number of documents in which term t occurs. k_1 and b are free parameters whose values are determined experimentally.

5 Proposed Ranking Approaches

In this section, we explain the basic idea and proposed ranking approaches.

5.1 Basic Idea

The problem is to find relevant legal judgments of given legal judgment. Generally, legal judgments are very large and complex text documents assembled with various sub-sections and links. Due to these inherent characteristics of large size and complex nature, there are a lot of intricate legal concepts at a granular level of the legal judgment which can be exploited to improve the performance of retrieving relevant legal judgments for a given input judgment. Essentially, we can observe that a legal judgment contains a large number of distinct legal concepts at a paragraph level. It will be very useful to a legal practitioner if judgments that capture the distinct legal concepts of the current proceeding are retrieved. To address this concern, we introduce a retrieval approach by exploiting the notion of paragraph-level granularity and make an effort to capture the legal concepts discussed at the paragraph-level and provide a ranked list of relevant legal judgments for a given input judgment. For example, as shown in the Figure 3 the first paragraph the concept of deprivation of property is discussed, in the second paragraph the concept of draft scheme published by the state transport department is being discussed, the two paragraphs in the same judgment capture two different legal concepts.

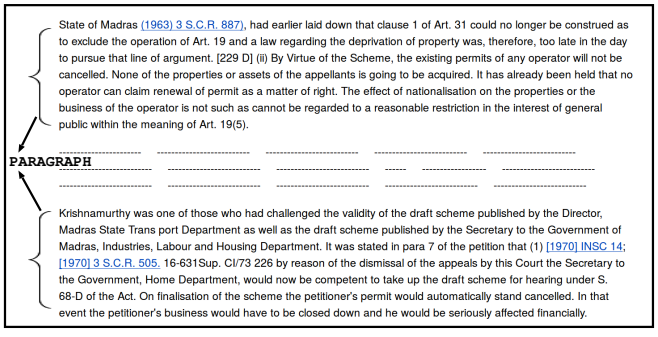


Figure 3: Paragraphs in the judgment

The process of extracting paragraph information and citation information from the corpus of Indian legal judgments is provided in the Algorithm 1. The input to the algorithm is the judgment corpus and the output of the algorithm is the extracted paragraphs of the judgments and citation information of the judgments. In the procedure *ExtractParagraphs*, we extract the text content of the legal judgments between the paragraph tags in the judgment. In the procedure *ExtractCitations*, we extract the unique ids references given to the legal judgments by using regular expressions for the format used by the reporters.

5.2 Paragraph-based Retrieval Approach (PA-rank)

In general, the input to the relevant document retrieval system is an input document. The input legal judgments are large and complex legal documents containing a lot of information in the form of legal concepts at minute levels of the document. Due to the large size of the legal documents and inherent complexity in the issues discussed by comparing the document overall by utilizing the text content or by using semantic-based approaches, the relevancy of the input judgment with the considered judgment may not be completely captured. In order to find relevant legal judgments for an input legal judgment, we try to capture the relevance of the input legal judgment with the judgments in the corpus by comparing the paragraphs of the query judgment with the paragraphs of the judgments in the corpus. We identify the most similar paragraphs for input paragraphs by comparing with the paragraphs of the other judgment and consider the aggregate score of matching between the top few similar paragraphs.

Algorithm 1 Extraction of Paragraphs and Citations from Legal Judgments

Input: Judgment Corpus J
Output: Paragraphs(P_J), Citations (C_J)

procedure EXTRACTPARAGRAPHS&CITATIONS(*J*)
  $P_J \leftarrow \{\}$ ▷ list containing list of paragraphs of each judgment in the corpus
  $C_J \leftarrow \{\}$ ▷ list containing list of paragraphs of each judgment in the corpus
  for all $J\ as\ J_i$ **do**
    $P_J[J_i] \leftarrow ExtractPagragraphs(J_i)$
    $C_J[J_i] \leftarrow ExtractCitations(J_i)$
  end for
  return P_J, C_J
end procedure

procedure EXTRACTPARAGRAPHS(*J*)
  $L_J \leftarrow \{\}$ ▷ list containing the paragraphs of J
  $rbegin \leftarrow$ "HELD:"
  $rend \leftarrow$ "Reproduced in accordance with..."
  $T_J \leftarrow extract(J, rbegin, rend)$ ▷ text content between begin and end tags in J
  $ptag \leftarrow$ '<p>' ▷ paragraph HTML tag
  for all $T_J.split(ptag)\ as\ p$ **do**
    $L_J.append(p)$ ▷ add p to list of paragraphs
  end for
  return L_J
end procedure

procedure EXTRACTCITATIONS(*J*)
  $L_J \leftarrow \{\}$ ▷ list containing the citations of J
  $rx \leftarrow$ regular expressions ▷ for format used by legal reporters SCR,SCC and AIR
  $L_J \leftarrow$ extract(T_J, rx) ▷ extract all expressions in the format specified by regular expression
  return L_J
end procedure

We consider three key factors i.e., term frequency, paragraph frequency and paragraph length to extract the importance score of a judgment. Okapi BM25 [18] is a retrieval model used by search engines to rank the matching documents according to their relevance to a given search query. In this paper, we rephrase the notion of original Okapi formula to extract the paragraph-level scores of relevant legal judgments for a given input judgment. In the proposed *PA-rank* approach, we consider paragraphs in the input judgment as the query and calculate the paragraph-level aggregate score for the input judgment with other judgments in the corpus. The retrieved judgments are ranked based on the normalized paragraph aggregate scores. The *PA-rank* formula is defined as follows.

$$ParaScore(P,\ QP) = \sum_{t=1}^{n} \frac{F(QP_t,\ P).(k_1+1)}{F(QP_t,P)+k_1(1-b+b\frac{|P|}{avgPL})}.idf(QP_t) \tag{3}$$

$$idf(QP_t) = log\frac{N - f(QP_t) + 0.5}{f(QP_t) + 0.5} \tag{4}$$

where QP is the query paragraph, P is a paragraph, QP_t represents each term t in QP, $F(QP_t, P)$ is the term frequency of QP_t in P, $|P|$ is the length of paragraph P, $avgPL$ is the average length of a paragraph in the corpus, N is the total number of paragraphs and $f(QP_t)$ is the number of paragraphs containing QP_t. k_1 and b are free parameters whose values are determined experimentally.

$$AggScore(J,\ QJ) = \frac{1}{|J|}\sum_{\substack{P\in J\\ QP\in QJ}} Max^m(ParaScore(P,\ QP)) \tag{5}$$

where QJ is the input query judgment, J is a judgment in the corpus, P is a paragraph in J, QP is a paragraph in $H(QJ)$ and $Max^m(ParaScore(P,\ QP))$ is a function which considers m maximum $ParaScore$ values to compute the $AggScore$. The parameter m is experimentally determined based on the statistics of the dataset considered. $AggScore$ is the aggregate retrieval score generated for a judgment with respect to the input query judgment.

The pseudocode for the procedure of *PA-rank* is provided in Algorithm 2. The input to the procedure is the query judgment and the corpus of judgments to find the relevant legal judgments for the given input judgment. The output of the algorithm is the ranked list of relevant legal judgments for the input judgment. Initially, we extract the paragraphs of all the judgments in the corpus. For each judgment in the corpus, we compute the *ParaScores* for each paragraph of the input legal judgment with the paragraphs of the judgments in the corpus and find the *AggScore* of the input judgment with the judgments in the corpus. The list of relevant legal judgments are extracted for the given input judgment and returned based on the AggScores of the comparison.

5.3 Citation-based Retrieval Approach (C-rank)

In this approach, we consider only the citation or the link content in the legal judgment for finding relevant legal judgments. The inherently connected nature of the judgments dataset provides an opportunity for modeling the judgments collection as a citation network and utilizing the connectivity information of the judgments to find relevant legal judgments.

Bibliographic coupling [9] between two judgments is defined as the number of common out-citations between two judgments. A legal practitioner cites another judgment if there is a significant similarity in the legal concepts discussed in the judgment. Citation between the two judgments represents a significant relationship between the two judgments. Hence, the existence of common out citations between two legal judgments denotes a significant similarity between the judgments. In this approach, for a given input judgment we extract the relevant legal judgments on the basis of BC similarity score which is provided in Equation 6 as follows.

$$BC(J,\ QJ) = CS(J) \cap CS(QJ) \tag{6}$$

where QJ is the input query judgment, J is a judgment in the corpus, $CS(X)$ denotes the citation set containing the citations of Judgment X, $BC(J,\ QJ)$ is the bibliographic coupling score between the two judgments.

Algorithm 2 Paragraph-based Retrieval (PA-Rank) Algorithm

Input: Judgments Corpus J, Query Judgment QJ
Output: Judgments Ranked list of relevant judgments for QJ RJ

procedure PA-RANK(J, QJ, m)
  for all $J\ as\ J_i$ **do**
    $P_J[J_i] \leftarrow ExtractPagragraphs(J_i)$
  end for
  $QPJ \leftarrow ExtractPagragraphs(QJ)$
  $L_J \leftarrow \{\}$ ▷ set containing relevance score of each judgment with QJ
  for all $J\ as\ J_i$ **do**
    $PScores \leftarrow []$ ▷ list of paragraph scores
    for all $PJ_i\ as\ P$ **do**
      for all $QPJ\ as\ QP$ **do**
        $Scores.append(ParaScore(P, QP))$ ▷ insert ParaScore(P,QP) into Scores
      end for
    end for
    $PScores \leftarrow Sorted(PScores)$ ▷ sorted in descending order
    $AggScore(J_i, QJ) = 0$
    $loopcounter = 0$
    for all $PScores$ as s **do**
      $AggScore(J_i, QJ) = AggScore(J_i, QJ) + s$
      if $loopcounter$ = m **then**// break
      end if
      $loopcounter$ = $loopcounter + 1$
    end for
    $RJ[J_i] \leftarrow AggScore(J_i, QJ)$
  end for
  $RJ \leftarrow Sorted(RJ)$ ▷ Sort RJ based on AggScore values in descending order
  return RJ
end procedure

5.4 Paragraph and Citation-based Retrieval Approach (PC-rank)

In addition to the text, legal judgments also contain citation-based information which can be exploited to improve the retrieval of relevant legal judgments. The inherently connected nature of the judgments provides an opportunity to model and exploit the connectivity information of the judgments. Citations information in the legal judgments provides emphasis on the commonality of two legal judgments

which can be exploited in addition to the text-based relevance. Sometimes two legal judgments may be similar but may not contain similar keywords. By exploiting citation-based information along with the text-based information, we can improve the retrieval performance of relevant legal judgments.

In PC-Rank approach, we capture both the aspects of the judgment by combining the citation based information with the score generated by PA-rank approach. We propose a linear extension of the *PA-rank* to incorporate the citation based information in the retrieval scores for extracting relevant legal judgments to the input query judgment. We define a combined score between two judgments by elevating the $AggScore$ of the judgments if the bibliographic coupling of the two judgments is above a certain threshold. The *PC-rank* formula is defined as follows.

$$CombinedScore(J,\ QJ) = \begin{cases} (1+\alpha) \times AggScore(J,\ QJ) & \text{if } BC(J,\ QJ) \geq \theta \\ AggScore(J,\ QJ) & \text{Otherwise} \end{cases} \quad (7)$$

where QJ is the input query judgment, J is a judgment in the corpus, $BC(J,\ QJ)$ is the bibliographic coupling score between the two judgments, α is a constant threshold and $AggScore(J,\ QJ)$ is aggregate retrieval score defined in the Equation 5.

6 Experiments

In this section, we explain about the dataset and analysis, performance metrics, approaches, and results.

6.1 Dataset and Analysis

We compare the performance of the retrieval approaches by considering the judgments dataset of Supreme Court of India [1] delivered over a period of 24 years $(1970\ to\ 1993)$.

Table 1 provide the summary, Figure 4a provide the degree distribution, and Figure 4b show the log-log degree distribution of citation details. The long tail of degree distribution curve and approximate linear nature of the log-log distribution function allows us to infer the power law distribution behavior of the legal citation network. The citation graph is very sparse as most of the legal judgments will have a few citations or none at all and very few judgments will have a large number of citations.

Table 1: Properties of citation information

Parameter	Value
Number of judgments	3738
Total number of citations	6117
Average degree of the citation network	3.28
Density of the citation network	0.00043
Range of Out-citations in a judgment	$1-41$
Range of In-citations in a judgment	$1-74$

6.2 Performance Metrics

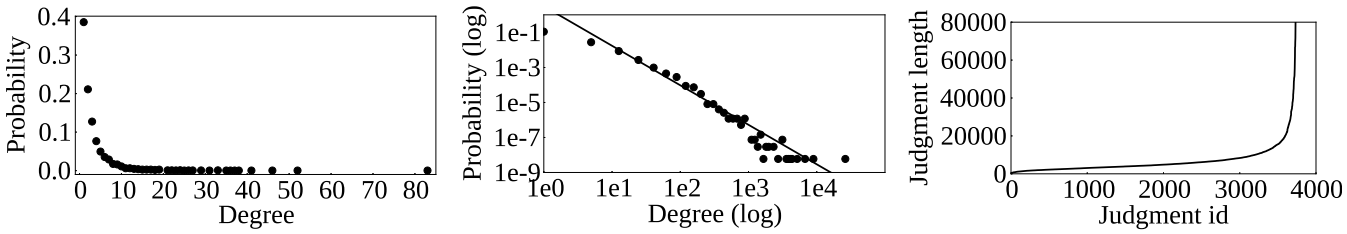
In the absence of standard results, we utilize a human expert score-based evaluation of the retrieval approaches. In the experiments, we consider two judgments as relevant to other if they are similar and the similarity indicates the commonality between the issues discussed in the judgments. We provided sample judgment pairs at random to each one of the 5 legal domain experts without informing each other of the computed similarity values. The legal experts provided a score between 0 to 10 based on the similarity between the pair of judgments where 10 indicates high similarity between the pair of judgments. We performed the analysis by averaging the score given by experts and evaluation was carried out by using the hit metrics by using the judgment pairs evaluated by the experts. The judgment J_1 is given as input judgment and a set of judgments were retrieved in ranked order for J_1. We then find the rank (k) of the judgment J_2 with respect to J_1 from the retrieved results and find the hit metrics using the following formula.

$$HitMetric(J_1, J_2) = \begin{cases} TruePositive(TP) & \text{if}(k \leq \theta_1\ \&\ es \geq \theta_2) \\ TrueNegative(TN) & \text{if}(k \geq \theta_1\ \&\ es \leq \theta_2) \\ FalsePositive(FP) & \text{if}(k \leq \theta_1\ \&\ es \leq \theta_2) \\ FalseNegative(FN) & \text{if}(k \geq \theta_1\ \&\ es \geq \theta_2) \end{cases}$$

here k is the rank obtained for J_2 with respect to J_1 and θ_1 is the threshold. Also, es is the expert score (es) for J_2 to be relevant to J_1 and θ_2 is the corresponding threshold value. Suppose, we consider all the judgments below rank five are relevant to J_1 and J_2 is returned as the third rank and es between J_1 and J_2 is 7 and we consider threshold of expert score, θ_2, is 5. Here k for J_2 is 3, es is 7 and θ_1 is also equal to 5. As a result, the Hit Metric($J_1,\ J_2$) is True Positive. In the experiments we extracted the ranked list of relevant legal judgments for a given input judgment among the judgments which are a part of the 100 evaluation pairs. We report the effectiveness of the retrieval approaches using binary classification measures of Precision as $\frac{TP}{TP+FP}$, Recall as $\frac{TP}{TP+FN}$, Accuracy as $\frac{TP+TN}{TP+TN+FP+FN}$, F1-score as $\frac{2\times TP}{(2\times TP)+FP+FN}$.

6.3 Approaches

- **Pivoted Cosine Approach:** Pivoted cosine normalization [25] is the standard vector normalization approach used in the case of varied length distributions. As the variance of the legal judgments length distribution is very high in the dataset as shown in Figure 4c, we consider pivoted cosine normalization as our baseline approach to compare the results of the proposed approaches. We apply the pre-processing steps of stop word removal and stemming and carry out the retrieval by using the pivoted normalization approach under the parameter setting ($slope$ and $pivot$ to 0.2 and average length normalization respectively) and the formula presented in [25].
- **PA-rank Approach:** For each judgment, we consider a paragraph as the text between two consecutive paragraph html tags. We extract all the paragraphs in the headnote section of judgment. From each extracted paragraph, we remove stop words and apply stemming. We consider the top *m* relevant paragraphs for calculating the paragraph aggregate scores as explained in the Equation 5 with the standard parameter values of $b = 0.75$ and $k_1 = 1.2$. The value of *m* is experimentally determined to be 35. With the set parameters, we carry out the retrieval using the proposed *PA-rank* formula mentioned in the Equation 5.
- **C-rank Approach:** For each judgment, we consider only the citation information in the legal judgments extracted by using the Algorithm 1. We use the Bibliographic coupling score [9] which is the number of common out-citations to retrieve relevant legal judgments for a given input legal judgment as shown in Equation 6.
- **PC-rank Approach:** In this approach, we consider both the paragraph level and citation information in the judgments. We consider the bibliographic coupling threshold for relevance as $\theta = 3$, which



(a) Degree distribution (b) Degree distribution(log) (c) Length distribution

Figure 4: Analysis on the dataset

has been defined as a reasonable estimate in the literature [17]. We set α as 0.5 experimentally. We carry out the proposed *PC-rank* retrieval using the formula mentioned in the Equation 7.

6.4 Results

Tables 2 and 3 summarize the *top-k* results for pivoted cosine, *PA-rank*, *C-rank* and *PC-rank* approaches in second, third, fourth and fifth columns respectively against the metrics defined in the first column. Table 2 provides the performance results of the three approaches for k = 5.

From Table 2, we could observe that the evaluation metrics for the proposed *PA-rank* and *PC-rank* approaches yield an improvement over the pivoted cosine approach and *C-rank* approach yields comparable results to the pivoted cosine approach. The *F1-score* metric obtained for *PA-rank* is 0.85. The *PA-rank* approach shows an improvement over the standard pivoted cosine approach. This observation supports our hypothesis that *PA-rank* approach captures the intricate legal concepts discussed at the paragraph level in the judgments and can be exploited for improving the retrieval performance. We can also observe that the *F1-score* metric obtained using *C-rank* is 0.78 which is comparable to that obtained using pivoted cosine approach. It can be observed that the *F1-score* metric obtained for *PC-rank* approach is 0.87 and the retrieval performance of the *PC-rank* approach is better than the *PA-rank* and *C-rank* approaches. It can be observed that the precision values of *C-rank* approach are considerably better than all other approaches.

Table 2: Retrieval performance (k = 5)

Parameter	Pivoted Cosine	PA-rank	C-rank	PC-rank
Precision	0.77	0.86	0.93	0.85
Recall	0.78	0.84	0.66	0.87
Accuracy	0.69	0.79	0.68	0.81
F1-Score	0.78	0.85	0.78	0.87

Table 3: Retrieval performance (k = 10)

Parameter	Pivoted Cosine	PA-rank	C-rank	PC-rank
Precision	0.71	0.75	0.93	0.75
Recall	0.90	0.91	0.66	0.91
Accuracy	0.69	0.72	0.68	0.72
F1-Score	0.79	0.82	0.78	0.82

From Table 3, we could observe that the performance of *PA-rank*, *C-rank* and *PC-rank* approaches are comparable to the pivoted cosine approach. The *F1-score* metric obtained for the pivoted cosine, *PA-rank*, *C-rank* and *PC-rank* are 0.79, 0.82, 0.78 and 0.82 respectively. This observation supports that citation based retrieval approach captures the relevancy between the legal judgments and can be used for the retrieval of relevant legal judgments. This study also suggests that exploiting paragraph-based information and citation-based can be useful in retrieving relevant legal judgments for a given input legal judgment.

We could observe from the results that the *PA-rank*, *C-rank* and *PC-rank* approaches perform slightly better when compared with the pivoted cosine approach. We analyze the $Recall$ performance by varying k from 3 to 15. We notice that the retrieval performance of the proposed *PA-rank* and *PC-rank* approaches is substantially higher when k is between 3 and 7. This observation suggests that the proposed *PA-rank* and *PC-rank* approaches have led to improving the retrieval performance substantially in retrieving highly relevant legal judgments (k = 3 to 7). As in the legal domain, given the time and effort, a relevant document of greater rank is of lesser importance to the legal practitioner in preparing the legal proceeding. This study demonstrates the ability of the proposed *PA-rank* and *PC-rank* approaches to rank the highly relevant documents toward the top of retrieval results when compared with the pivoted cosine approach.

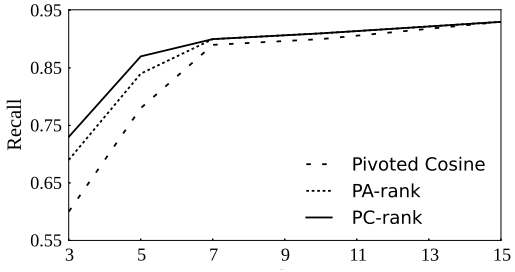


Figure 5: Recall performance

7 Discussion

In this paper, we analyze the advantages of using paragraph-based information and citation information and try to address the problem of associative legal judgment retrieval for finding a ranked list of relevant legal judgments for a given input judgment. Due to the lack of standard results on Indian legal judgments delivered by Supreme Court of India for retrieval of relevant legal judgments, we use expert score based evaluation of the proposed approaches. The bias is minimized in the expert score based evaluation by taking average expert score across multiple experts and all of them having similar aspects for the relevance of a judgment with respect to another judgment. The citation network on Indian legal judgments is sparse and very low in number. But the importance of the link information between the judgments prompts for building approaches which utilize the citation information to extract relevant legal judgments for a given input judgment. Incidentally, in the evaluation dataset judgments which are similar based on citation-based similarity are also similar based on paragraph-level similarity, as a result, the distinct performance

difference of citation-based similarity could not be separated prominently from the paragraph based approach using this dataset. This aspect will be investigated as a part of future work.

8 Summary and Conclusions

In this paper, we have analyzed how paragraph level and citation information can be exploited to extract relevant legal judgments. By employing Okapi retrieval model, we proposed approaches to computing relevant judgments by exploiting paragraph level information. The user evaluation study on legal judgments data set delivered by Supreme Court of India shows that the proposed approach improves the ranking performance over the baseline approach. Overall, the analysis shows that there is a scope to exploit the paragraph-level and citation information of the judgments to improve the search performance. As a part of future work, we would like to exploit the role of act-based information of the judgments in improving the retrieval performance. In addition, we would like to explore hybrid techniques using weighted PA-rank and weighted PC-rank.

REFERENCES

[1] The Supreme Court of India Judgments. `http://www.liiofindia.org/in/cases/cen/INSC/`.

[2] Maristella Agosti, Roberto Colotti, and Girolamo Gradenigo, 'A two-level hypertext retrieval model for legal data', in *Proceedings of the 14th annual international ACM SIGIR*, pp. 316–325. ACM, (1991).

[3] Nitin Bilgi and RV Kulkarni, 'A investigative survey of application of knowledge based system in legal domain', *IJITKM*, 517–525, (2008).

[4] Sergey Brin and Lawrence Page, 'The anatomy of a large-scale hypertextual web search engine', *Computer Networks*, **30**(1-7), 107–117, (1998).

[5] Núria Casellas, Pompeu Casanovas, Joan-Josep Vallbé, Marta Poblet, Mercedes Blázquez, Jesús Contreras, José-Manuel López-Cobo, and V Richard Benjamins, 'Semantic enhancement for legal information retrieval: Iuriservice performance', in *Proceedings of the 11th ICAIL*, pp. 49–57. ACM, (2007).

[6] Yen-Liang Chen, Yi-Hung Liu, and Wu-Liang Ho, 'A text mining approach to assist the general public in the retrieval of legal documents', *JASIST*, **64**(2), 280–290, (2013).

[7] Huu-Thanh Duong and Bao-Quoc Ho, 'A vietnamese question answering system in vietnam's legal documents', in *CISIM*, pp. 186–197. Springer, (2014).

[8] Hui Fang, Tao Tao, and ChengXiang Zhai, 'A formal study of information retrieval heuristics', in *Proceedings of the 27th Annual International ACM SIGIR*, SIGIR '04, pp. 49–56. ACM, (2004).

[9] M. M. Kessler, 'Bibliographic coupling between scientific papers', *American Documentation*, **14**(1), 10–25, (1963).

[10] Jon M. Kleinberg, 'Authoritative sources in a hyperlinked environment', *J. ACM*, **46**(5), 604–632, (1999).

[11] Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Aditya Singh, 'Similarity analysis of legal judgments', in *Proc. 4th Annual COMPUTE*, pp. 17:1–17:4. ACM, (2011).

[12] Sushanta Kumar, P.Krishna Reddy, V.Balakista Reddy, and Malti Suri, 'Finding similar legal judgements under common law system', in *DNIS*, pp. 103–116. Springer, (2013).

[13] Qiang Lu, Jack G Conrad, Khalid Al-Kofahi, and William Keenan, 'Legal document clustering with built-in topic segmentation', in *Proceedings of the 20th ACM international conference on Information and knowledge management*, pp. 383–392. ACM, (2011).

[14] K Tamsin Maxwell and Burkhard Schafer, 'Concept and context in legal information retrieval.', in *JURIX*, pp. 63–72, (2008).

[15] Divya Padmanabhan, Prasanna Kumar Desikan, Jaideep Srivastava, and Kashif Riaz, 'WICER: A weighted inter-cluster edge ranking for clustered graphs', in *International Conference on Web Intelligence*, pp. 522–528, (2005).

[16] Paulo Quaresma and Irene Pimenta Rodrigues, 'A question-answering system for portuguese juridical documents', in *ICAIL*, pp. 256–257. ACM, (2005).

[17] K. Raghav, Pailla Balakrishna Reddy, V. Balakista Reddy, and Polepalli Krishna Reddy, 'Text and citations based cluster analysis of legal judgments', in *MIKE*, pp. 449–459. Springer, (2015).

[18] Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford, 'Okapi at TREC-3', in *TREC*, pp. 109–126. National Institute of Standards and Technology, (1994).

[19] G. Salton, A. Wong, and C. S. Yang, 'A vector space model for automatic indexing', *Commun. ACM*, **18**(11), pp. 613–620, (1975).

[20] Gerard Salton, James Allan, Chris Buckley, and Amit Singhal, 'Readings in information visualization', chapter Automatic Analysis, Theme Generation, and Summarization of Machine-readable Texts, 413–418, Morgan Kaufmann Publishers Inc., (1999).

[21] M. Saravanan, Balaraman Ravindran, and S. Raman, 'Improving legal document summarization using graphical models', in *JURIX*, volume 152 of *Frontiers in Artificial Intelligence and Applications*, pp. 51–60. IOS Press, (2006).

[22] M. Saravanan, Balaraman Ravindran, and S. Raman, 'Improving legal information retrieval using an ontological framework', *Artif. Intell. Law*, **17**(2), 101–124, (2009).

[23] S Sendhilkumar, E Elakkiya, and GS Mahalakshmi, 'Citation semantic based approaches to identify article quality', in *Proceedings of International conference ICCSEA*, pp. 411–420, (2013).

[24] Maria De Lourdes Da Silveira and Berthier A. Ribeiro-Neto, 'Concept-based ranking: a case study in the juridical domain', *Inf. Process. Manage.*, **40**(5), 791–805, (2004).

[25] Amit Singhal, Chris Buckley, and Mandar Mitra, 'Pivoted document length normalization', in *Proceedings of the 19th annual SIGIR*, pp. 21–29. ACM, (1996).

[26] Howard Turtle, 'Text retrieval in the legal world', *Artificial Intelligence and Law*, **3**(1-2), 5–54, (1995).

[27] Paul Zhang and Lavanya Koppaka, 'Semantics-based legal citation network', in *The Eleventh International Conference on Artificial Intelligence and Law, Proceedings of the Conference, June 4-8, 2007, Stanford Law School, Stanford, California, USA*, pp. 123–130, (2007).