Measuring Similarity among Legal Court Case Documents

Arpan Mandal
IIEST Shibpur
Howrah, West Bengal, India

Raktim Chaki
IIEST Shibpur
Howrah, West Bengal, India

Sarbajit Saha
IIEST Shibpur
Howrah, West Bengal, India

Kripabandhu Ghosh
IIT Kanpur
Kanpur, Uttar Pradesh, India

Arindam Pal
TCS Research
Kolkata, West Bengal, India

Saptarshi Ghosh
IIT Kharagpur
Kharagpur, West Bengal, India
IIEST Shibpur
Howrah, West Bengal, India

ABSTRACT

Computing the similarity between two legal documents is an important challenge in the Legal Information Retrieval domain. Efficient calculation of this similarity has useful applications in various tasks such as identifying relevant prior cases for a given case document. Prior works have proposed network-based and text-based methods for measuring similarity between legal documents. However, there are certain limitations in the prior methods. Network-based measures are not always meaningfully applicable since legal citation networks are usually very sparse. On the other hand, only primitive text-based similarity measures, such as TF-IDF based approaches, have been tried till date. In this work, we focus on improving text-based methodologies for computing the similarity between two legal documents. In addition to TF-IDF based measures, we use advanced similarity measures (such as topic modeling) and neural network models (such as word embeddings and document embeddings). We perform extensive experiments on a large dataset of Indian Supreme Court cases, and compare among various methodologies for measuring the textual similarity of legal documents. Our experiments show that embedding based approaches perform better than other approaches. We also demonstrate that the proposed embedding-based methodologies significantly outperforms a baseline hybrid methodology involving both network-based and text-based similarity.

CCS CONCEPTS

•Information systems →Information retrieval;

KEYWORDS

Legal Information Retrieval, Legal Document Similarity, Court Cases, Topic Modeling, Word Embeddings, Word2vec, Doc2vec

1 INTRODUCTION

The Common Law System is one of the most followed legal systems in the world and it is followed in numerous countries including Australia, India, United Kingdom, and the USA.[1] The pivotal characteristic of a Common Law System is that it accords great importance to *prior cases*. It follows a principle (called *Stare decisis*), where it mandates that similar facts or similar situation should yield similar results.[2] It means, if a related or comparable case has earlier been decided, then the court is bound to study and follow the interpretation of the related issue(s) made in the previous case. If a previous case is referred to be useful in the proceedings of the ongoing case, then it is called a prior case (or a precedent). Precedents are considered to be equally important as any other written law (a *statute*) in this legal system. So, legal professionals arguing a case frequently need access to favorable older cases which are relevant to the current case.

The advent of Information Technology has caused a rapid increase in the number of legal documents that are digitally available. With this increased size of the domain, it is getting intractable for legal practitioners to manually find relevant prior cases that would assist an ongoing case in their favor. It is hence essential for legal practitioners to have systems which automatically retrieve relevant prior cases. The problem of building an automatic prior case retrieval system can be modeled as a task of Information Retrieval. Traditional IR systems (e.g., Web search engines) are presented with queries that are within a range of a few words or sentences. In case of Legal IR systems, a legal practitioner should be able to provide the system with a description of the ongoing case. This description of the current case would then serve as the 'query' for which relevant prior cases are to be found.

To this end, *measuring the similarity between two court case documents* is a critical challenge that needs to be addressed for finding relevance between the current case description and a previously decided case. This task is especially challenging considering the fact that legal documents are generally long and complex in structure [3]. Additionally, a single legal case document might include discussions of many different legal issues. These complexities imply that manually understanding similarity between two documents needs expensive domain expert knowledge. Hence, automating the process of similarity measurement between two court case documents is of special importance.

ACM COMPUTE 2017, November 16–18, 2017, Bhopal, India

978-1-4503-5323-6/17/11...$15.00
DOI: 10.1145/3140107.3140119

[1] https://en.wikipedia.org/wiki/List_of_national_legal_systems/
[2] https://en.wikipedia.org/wiki/Common_law/

Few prior methodologies have been proposed for measuring similarity between legal documents; these methodologies can be broadly classified into network-based methods (which utilize the citation network among legal documents), and text-based methods (which utilize the textual content of legal documents); some hybrid methods have also been proposed (see Section 2 for a survey of these methods). However, the prior methodologies have certain limitations. On one hand, network-based methodologies are not always applicable since legal citation networks are usually very sparse [5]; hence the network-based measures cannot be meaningfully computed for most of the cases. On the other hand, to our knowledge, the text-based methods that have been used on legal documents are mostly primitive, such as TF-IDF based approaches [6]. These limitations of the prior methodologies motivated us to develop improved methodologies for measuring the similarity between legal documents.

In this paper, we explore various *text-based* methods to measure similarity between two legal documents. We do this by first considering different representations of a legal document, and then measuring similarities between these representations by various recent techniques. Among the different possible representations of a legal document, we consider (i) the whole document, (ii) the set of paragraphs of the document, (iii) summaries of the document, and (iv) the set of RFCs (the text surrounding a citation) in the document. These representations are attempts to separate out the meaningful / important units of the document.

The second step is to measure the similarity between the documents (or the document units). A popular methodology for this step is to form a vector representation of the document and then measure the cosine similarity between the vectors. As stated earlier, prior works [6] used TF-IDF vectors for this purpose. In this work, we apply advanced vector representations, such as topic modeling (Latent Dirichlet Allocation [2]) and neural network based approaches (e.g., word embeddings [8] and document embeddings [7]). These methods capture the *semantics* of the documents, and hence are likely to better formulate the similarity between documents even if different terms having the same semantic context are used in different documents. Comparing among the different methodologies, we found that the embeddings learned by the neural network approaches (Word2vec and Doc2vec) out-perform the other methodologies for the said task. To the best of our knowledge, this is the first attempt at using neural network based models to measure the similarity among legal case documents.

We demonstrate the efficacy of our methodologies through extensive experiments over a large dataset of Indian Supreme Court case documents. We consider as baseline, the prior work [6], which proposed a hybrid methodology for computing similarity between legal documents, involving both text-based and network-based techniques. The work [6] evaluated the methodology over 50 document-pairs, by correlating the similarity judgments of the methodology with that of legal experts. We reuse the same evaluation framework, and show that our proposed methodologies significantly outperform the baseline methods. In other words, the similarity scores inferred by the methods proposed in this work are much more correlated with similarity scores judged by legal experts, as compared to the baseline methods.

The rest of the paper is organized as follows. In Section 2, we briefly survey previous attempts in measuring similarity between legal documents. Section 3 discusses in detail the methods we have experimented with. The performance of the different methodologies is evaluated in Section 4, which also compares the proposed methodologies with the baseline. The last section contains the conclusions and future works.

2 RELATED WORK

The methods for measuring similarity between two legal documents can be broadly classified into text-based methods and network-based methods. Also, some hybrid approaches have used a combination of text-based and network-based methods. In this section, we briefly discuss some of the approaches proposed in the literature for measuring similarity between legal documents.

2.1 Network-based similarity measures

Network-based similarity measures consider a citation network of legal documents. A citation network of legal documents is a directed graph where, every node is a document, and each directed edge $A \rightarrow B$ between two documents signifies that the document A cites the document B. For instance, B might be a prior case that is cited as a precedent by a later case A.

Various similarity measures have been proposed based on the legal citation network. For instance, Kumar *et al.* [5], proposed two network-based similarity measures – *bibliographic coupling* and *co-citation*. Bibliographic coupling between two documents is the number of common 'out-citations' of these two documents. Whereas, co-citation is the number of common 'in-citations' between the two documents. It was shown that the two network-based methods are effective for finding similar judgments.

Two other network-based similarity measures were proposed in [9] – *dispersion* and *embeddedness*. Dispersion is an interesting measure that has previously been used on Facebook networks [1] to predict the link of romantic relationships within a small network. For a given pair of nodes, the Dispersion measure attempts to measure to what extent are the common neighbors of the nodes *not* themselves well connected. The other measure, Embeddedness, in simple terms means, how close the common neighbors of a node are to form a clique.

Although these network based methods perform well in finding similar documents, it is clear that the applicability of these methods depends on how well-connected the underlying citation graph is. Legal court case documents generally form a very sparse citation graph [5]. For instance, in the dataset that we use for this paper, the citation graph is very sparse, with more than 80% of the nodes being isolated nodes (i.e., not containing any citation). The spareness of legal citation graphs is primarily because lawyers / judges do not usually note down all possible cases relevant to a given case; rather, only a few of the most important prior cases are cited. In such circumstances, text-based similarity measures (or hybrid ones) are more useful than purely network-based measures.

2.2 Text-based and hybrid similarity measures

In addition to the network-based measures discussed earlier, Kumar et al. [5] also proposed two network-based measures – all-term cosine similarity, and legal-term Cosine Similarity. Both these measures are based on the TF-IDF scores of terms present in the documents. The documents are represented as vectors, of the size equal to the number of distinct words in the vocabulary of the entire corpus. So, each component / coordinate of the vectors would correspond to a distinct term in the vocabulary, and contains the TF-IDF score of the term. Once the vectors for both the documents are formed, cosine similarity was computed between the two vectors to obtain the similarity between the documents.

Kumar et al. [6] also presented a hybrid algorithm, that is based both on the text and on the citation network. For this, they introduced a new metric called *paragraph-links* or PLs. To calculate PLs, each document is first broken into their constituent paragraphs. Then, each paragraph is considered as a separate entity. Now a similarity score across all pairs of paragraphs is measured. Now, if the similarity between any two paragraphs is higher than a threshold then a *paragraph link* is added between the documents containing these two paragraphs. In other words, let $[A_1, A_2, A_3, ...A_i]$ be the set of paragraphs in document A and similarly let $[B_1, B_2, B_3, ...B_j]$ be the set of paragraphs in document B. So, if we measure similarities between the paragraphs of A and B, we have $i \times j$ similarity values. Now, a *paragraph-link* is inserted between A and B if any one of these $i \times j$ similarity values is greater than a threshold.

For measuring similarity between a pair of paragraphs, they first computed TF-IDF vectors for the paragraphs (considering the terms contained in the two paragraphs), and then computed the cosine distance between these two vectors. They showed that the PL method out-performed the bibliographic coupling measure from [5], over a restricted set of 3,866 Court Case Judgments from the Indian Supreme Court. We consider this study [6] as a baseline in the present work.

2.3 Present work

From the discussion above, it is evident that only primitive measures such as TF-IDF have been applied to the task of measuring similarity between legal case documents. In this work, we apply several more advanced techniques for measuring similarity between legal documents, including topic models, and neural network models such as word-level and document-level embeddings. We perform experiments over a dataset that is an order of magnitude larger than that of prior works [6], and show that the embedding-based approaches proposed in this work significantly out-perform the methods in prior works.

3 METHODOLOGIES FOR MEASURING DOCUMENT SIMILARITY

As stated earlier, in this work, we specifically consider text-based methodologies for computing the similarity of two legal documents. Measuring document similarity between the text of two legal documents involves two sub-tasks:

(1) **Finding an appropriate representation of a document** – most simply, the representation can be the whole document. However, a legal document is likely to discuss several different legal issues. Hence, an alternative approach can be to select appropriate subsets of the document (such that each subset captures a single legal issue), and then to measure the similarity between the subsets of the two documents.

(2) **Measuring the similarity between the representations of two given documents** – once the representations of the two documents have been selected, an appropriate methodology needs to be employed to measure the similarity between the representations.

In this section, we describe several methodologies for the two subtasks stated above. Different ways to select the representation of a legal document are described in Section 3.1, while Section 3.2 discusses various ways to measure similarity between the representations.

3.1 Selecting representations of a legal document

We apply simple techniques to either break the whole document into parts, or select specific important portions from it. The technique for measuring similarity (described later) depends on the partitioning / selection technique. We describe below the partitioning / selection techniques that we experimented with.

3.1.1 **Whole Document:** The first and simplest strategy of all is to consider the whole document. However, after consulting with legal experts, we understood that a single legal document might discuss multiple different legal issues. For instance, the same case document might discuss *validity of an evidence*, *the severity of a crime committed* and *the interpretation of a law*. When a document d_2 is cited as a prior case from the case document d_1, it is possible that d_1 and d_2 have just one legal issue in common (for which the citation was done), while the two documents are, in general, not very similar. To capture such partial similarities between two legal documents, we also consider other representations of a legal document, as described below.

3.1.2 **Summary of the document:** The summary of a document gives us an idea of the important points discussed in the document, while filtering out the redundant, less important parts. So, using summaries instead of the complete text can intuitively give us better representations of the important concepts that are discussed in a document.

We used the methodology proposed in the prior work [4] to generate the summary of a legal document. In this methodology, an *importance score* is first assigned to each sentence in the document, the sentences are then ranked in decreasing order of their scores, and the top 10% of the total number of sentences are selected to form the summary.[3] To score each sentence, an unsupervised technique called *MyScore* was used [4].

To score a sentence by this method, we first compute a score for each of the *terms* in the sentence and then, take an average of

[3] In cases where 10% of the sentences counted less than 5, a minimum of five sentences are selected.

these term-scores. Terms are considered to be those words in the sentence, which have a minimum term frequency of 2 (after stopword filtering, and stemming). For each term in the sentence, three different scores are calculated. The terms are ranked by separately sorting them using each of the three scores, and their final score is considered to be the average of these ranks. The three scores used in sorting the terms are – (i) TF-IDF score, (ii) frequency of the term in the document, (iii) the third score is obtained as follows. First we form a collection of sentences that contain at least one of the ten most frequent terms in the document. Let this collection be S. A set of candidate terms is then formed by taking all terms $t \in S$ that has a minimum term frequency 2. For each of the candidate terms t, we calculate a ratio $TF(t, S)/TF(t, Doc)$, where, $TF(t, S)$ is the number of times t has occurred in S, and $TF(t, Doc)$ is the number of times t has occurred in the Document.

Thus, a term-score is computed for each term (having frequency more than 2 in the document), and based on these term-scores, each sentence is assigned a score. The summary is constructed using the 10% top-scored sentences.

3.1.3 **Paragraphs of the document:** From our discussions with legal experts, we understood that a single paragraph in a legal case document is likely to discuss one specific legal issue. So, considering each paragraph in a document separately seems promising, since measuring the similarity between individual paragraphs in two documents might indicate the most similar legal issues discussed in the two cases.

Note that the method for identifying a paragraph-break in a document is specific to the corpus. In our case, we broke the text into paragraphs wherever there were two consecutive newline characters (i.e., wherever there was a blank line). Also, we discarded paragraphs that were shorter than 20 words in length. In this way, the average number of paragraphs in a document was found to be 26.7 and, the average number of words per paragraph was found to be 58.6.

It can be noted that measuring similarity for this method is different from the ones described above. Here, a document d is represented as a set of paragraphs $\{p_1, p_2, ..., p_n\}$, where n is the number of paragraphs in d. Given the two documents d_1 and d_2 between which the similarity has to be computed, let n_1 and n_2 be the number of paragraphs in d_1 and d_2 respectively. Then, we can compute a total of $n_1 \times n_2$ similarity scores between the paragraphs of the two documents. To get the final similarity score between the two documents, we consider the average (mean) of all these similarity scores.

3.1.4 **RFCs of the document:** Legal case documents often include citations to prior cases. The text surrounding such citations are called *Reason for citation* (abbreviated as *RFC*), since these text portions indicate why a particular case document cited another. Prior works [10] hypothesized that such RFCs include the important legal issues discussed in a document, which also provide the reasons why a future case should cite the current case.[4] Thus, focusing on the RFCs seems to be a promising way of identifying the important legal issues described in a document. To generate the RFCs from a given document, we consider the 40 preceding words and the 40 following words from the point where a citation was made in a document. Table 1 shows an example of RFCs in a sample case document from our dataset.

Using RFCs to measure the similarity between two legal documents is somewhat similar to the case of using paragraphs to represent a document (described above). If a document d has n number of citations (to prior cases), it can be represented as a set of RFCs $\{r_1, r_2, \ldots r_n\}$. Given the two documents d_1 and d_2 between which the similarity has to be computed, let n_1 and n_2 be the number of RFCs in d_1 and d_2 respectively. Similar to the case with paragraphs, we compute all possible $n_1 \times n_2$ similarity scores between the RFCs of the two documents, and then consider the average (mean) of all these similarity scores as the final similarity score between the two documents.

3.2 Measuring similarity between the representations

Once we have selected the meaningful parts (representations) from the text of the documents (between which similarity is to be measured), we then need to measure the similarity between the representations.

The problem of measuring the similarity between two text documents (representations) has received wide attention from the Information Retrieval and Data Mining research communities, and several well-known techniques exist in literature. The most popular techniques form a vector representation of the two documents, where the dimensions of the vector representation might be the terms contained in the documents, or latent topics or even semantic concepts. Once the vector representations of the two documents are obtained, the popular *cosine similarity* measure can be used to measure the similarity between the two vectors. Given two vectors A and B of dimensionality n each, the cosine similarity between the vectors is:

$$cos\_sim(A, B) = cos(\theta) = \frac{\sum_{i=1}^{n} A_i B_i}{\sqrt{\sum_{i=1}^{n} A_i^2}\sqrt{\sum_{i=1}^{n} B_i^2}}$$

where, A_i and B_i are the i^{th} components of the vectors A and B respectively, and θ is the angle between A and B in the n-dimensional space.

The question now remains as to what is a good way of converting the representation of a legal document to a vector representation, so that the similarity can be measured accurately. In this work, we experiment with the following four popular techniques to get the vectors for each selected text portion, and then compute the similarity between the vectors – (i) TF-IDF Vectorizer, (ii) Topic models, viz. Latent Dirichlet Allocation, (iii) Word embeddings, viz. Word2vec, and (iii) Document embeddings, viz. Doc2vec. In the rest of this section, we describe the four techniques for forming vectors out of a given text, how the corresponding models were formed and, how we use them to finally generate the embedding for each portion of text.

3.2.1 **Preprocessing the text:** We perform some basic preprocessing and normalizations over the text portions, before the

[4] LexisNexis, a popular legal retrieval system (https://www.lexisnexis.com/), is said to be developed based on this principle.

> On 14.7.1992 we passed a detailed order and so far as the admission rules fixing 50% of the marks to be obtained at the entrance examination as minimum qualifying marks for admission to the Post-Graduate medical courses are concerned the 700 same were held to be legal and it was further held that no exception can be taken to the same. It was however, contended on behalf of the petitioners that as a *result of the application of the aforesaid rule a large number of seats have remained vacant and in view of the observations made in Dr. Ambesh Kumar etc. etc., v. Principal, LLRM Medical College Meerut and Ors. etc. etc.* [[1986] INSC 274]; [[1987] 1 SCR 661] *such a situation must be avoided and the remaining seats should be filled up by applying different criteria, the cases were postponed for further hearing. We have heard learned cousel for the parties* and have thoroughly gone through the record. So far as the validity of the admission rules fixing 50% marks for the general category candidates and 40% marks for the SC/ST category candidates to be obtained at the entrance examination as minimum qualifying marks for being eligible for admission to the Post-Graduate medical courses, the same are not subject to any challenge as we have already held the same to be legal in our order dated 14.7.1922. *Learned counsel for the petitioners made strenuous effort to persuade us to take a different view, but they failed in the said attempt. It may be further mentioned that this Court in Ajay Kumar Agrawal and Others. v. State of U.P. and others* [[1990] INSC 359]; , [[1991] 1 SCC 636] *observed as under :- "It is not disputed that in Uttar Pradesh the prevailing practice was a 50 per cent for allowing Post Graduate Study to doctors with MBBS qualifications but taking their University examination as the base without any separate selection test*, it is not the case of any of the parties before us that the selection is bad for any other reason. We are of the view that it is in general interest that the 50 per cent cut-off base as has been adopted should be sustained."

Table 1: An excerpt from the court case 1992_INSC_182 showing the RFCs. The citations are underlined and, the RFCs are the italicized parts around the citations.

formation of vectors. These steps help to form better representations of a given text.

(1) Converting all alphabets to lowercase
(2) Tokenizing the text into words based on whitespaces
(3) Filtering out all words that are not alphabetic, except words containing the characters hyphen, dot and comma
(4) Removing standard English stopwords from the list of words.
(5) Performing stemming over all words, using the popular Porter Stemmer
(6) Filtering out all words that occur in less than three documents in the entire corpus. This step is a crude attempt to remove mis-spelled words and words that are very rare and might not add enough context to the final representation of the document.

3.2.2 TF-IDF Vectors:

This is a simple yet effective approach for converting a given text into a single vector. Each distinct term in the vocabulary of the whole corpus (document collection) is considered as a dimension for the vector space. A document d is represented as a vector of length equal to the number of distinct words in the vocabulary of the corpus. The i^{th} element of the vector corresponds to the i^{th} word in the vocabulary, and contains the TF-IDF score of the corresponding word, where the TF (term frequency) measures the number of times the word is present in d, and IDF measures the inverse of the frequency of the word in the whole corpus.

For computing the similarity score between a pair of texts we simply find the cosine distance between the two TF-IDF vectors corresponding to the two texts.

3.2.3 Topic models, viz. Latent Dirichlet Allocation (LDA):.

Topic models are statistical models for discovering topics in a text corpus. We use the widely used topic modeling algorithm Latent Dirichlet Allocation (LDA) [2]. LDA identifies topics as mixtures of terms (words), and treat documents as mixtures of topics. Each topic is represented as a probability distribution over the vocabulary of all terms in the corpus, where the probability values convey the affinity for a given term to a particular topic. In simple terms, each topic (as identified by LDA) is a set of frequently co-occurring terms, and each document is assigned probabilities of belonging to the different topics.

We first train LDA over our entire corpus (document collection). While training LDA, the number of topics is to be provided as an input parameter, and the number of topics is the dimensionality of the vector provided by the LDA model. We have tried LDA by varying the number of topics as 40, 80, 120, 160 and 200. We found that the LDA model performed best with 80 topics; hence, all results for LDA presented in this paper uses the model trained with 80 topics. Thus, for any given text (e.g., a whole document, a paragraph, or a RFC), the LDA model gives a vector (of size 80) which is expected to capture the topical distribution of the text. To measure the similarity between two texts, we measure the cosine similarity of the LDA vectors of the two texts.

3.2.4 Word embeddings, viz. Word2vec:

Word2vec is a neural network model that, when trained over a document collection, can learn a vector for every distinct word in the collection [8]. Briefly, the vector for a given word, called the *word embedding*, is learned by looking at the other words which surround the occurrences of the given word. When trained with a sufficiently large set of text documents, the embeddings learned can provide meaningful contexts of the words. In other words, the embeddings of two semantically similar words (which have similar context) tend to lie closer to each other in vector space.

Word2vec requires that the dimensionality of the embeddings produced for each word be provided as an input parameter. We tried training Word2vec with different embedding sizes, and found that embeddings of size 200 produce the best results.[5] Hence all results reported in the paper are for embeddings of size 200.

We first trained Word2vec over our entire corpus. Word2vec gives embeddings for individual words only. But, for our problem we need to get vectors for a text portion (e.g., a paragraph, or an RFC), So, we needed to somehow combine the vectors of the constituent words of a text snippet to get the final vector for the text. To this end, we compute the *weighted average of the embeddings of*

[5] We used the Word2vec implementation from the open-source package Gensim (https://radimrehurek.com/gensim/models/word2vec.html).

the words contained in the text, where the embedding of each word is weighted by the TF-IDF score of the word. Note that word embeddings are known to be additive, i.e., the vector obtained by adding the embeddings of two words is expected to capture the overall context of the two words [8].

Once we get the Word2vec embeddings for two text portions, the similarity between the text portions is simply computed as the cosine similarity between the two vectors.

3.2.5 **Document embeddings viz. Doc2vec:** This embedding technique [7] is an extension to the model used in Word2vec. As the name suggests, this model directly converts a given text into a vector. Similar to the case of Word2vec, we used our whole corpus to train Doc2vec. As Doc2vec requires us to provide the size of the embedding as an input parameter, we tested its performance with many sizes and found that Doc2vec models with a size of 100 performs better than the rest. So, all the results for Doc2vec presented in this paper considers a model with a size of 100.[6] Computation of the similarity score is simple when we are provided with the Doc2vec vector representations of the pair of documents – we simply compute the cosine similarity between these two vectors.

3.3 Summary

In this section, we listed various options for two key questions that are necessary to address for effectively measuring the similarity between two legal case documents – (1) how to represent a document, and (2) how to measure the similarity between the representations of two documents. For representing a document, we discussed four options such as using the whole document, a summary, the set of paragraph, and the set of RFCs in the document. For the second question, we list four different methods of computing similarity – TF-IDF, topic models, word embeddings, and document embeddings. In the next section, we will compare among the different representations and methodologies for computing similarity.

4 EXPERIMENTAL SETUP AND RESULTS

In this section, we do experiments to investigate (1) which is the best document representation for legal documents, and (2) which is the most accurate document similarity technique. For evaluation, we test the consistency of these scores with those provided by the domain experts.

4.1 Dataset of legal documents

For our purpose, we have selected 33,545 court case documents of the Indian Supreme Court. The dataset contains all court case decisions of the Indian Supreme Court, ranging over a period of 67 years (from 1950 to 2016), available in text format. Each text starts with an optional headnote[7], and is followed by the complete litigation process of the case.

The texts were crawled from the *LIIofIndia* website (http://www.liiofindia.org/databases.shtml), which is a website hosted by the Legal Information Institute of India, that openly hosts a variety of legal databases. The documents were crawled using a simple web crawler implemented in Python language using *urllib*. The downloaded HTML files were then parsed to get the final texts. The headnotes were removed, and each text file was stored as a sequence of sentences.

Gold-Standard for legal document similarity: A gold-standard comprising of legal expert judgments on how similar two documents are, is essential to compare and evaluate our methods. To this end, we have used a set of 50 pairs of case documents from the prior work by Kumar et al. [6] (which is our baseline, described in section 4.3). The paper [6] stated similarity scores provided by legal experts for 50 pairs of Indian Supreme Court Case documents. Among these, there were three pairs for which at least one of the documents was missing in our dataset. So, measuring similarity for these three pairs was not possible. Hence, we consider similarity scores for the remaining 47 pairs of cases as our gold-standard. For each of these pairs, the experts were requested to assign a similarity score on a scale of 0 to 10, where 0 indicates the lowest similarity, while a score of 10 indicates very similar documents. Table 3 states the document-pairs (column heading – *Case Pairs*), along with the expert scores (column heading – *Expert Scores*) as reported in [6].

4.2 Evaluation measure

To evaluate our methods, we first compute similarity scores for each of the 47 test pairs by each of our methods. Then, for each method we check the correlation between the 47 similarity values given by the method and those given by the experts. We use *Pearson correlation coefficient* to quantify how well our methods perform when compared to the similarity scores given by the experts. It is mathematically defined as the ratio of covariance and standard deviations of the two variables. To calculate it, let X and Y be two variables.

$$\rho = \frac{cov(X,Y)}{\sigma_X . \sigma_Y} \qquad (1)$$

where $cov(X,Y)$ is the covariance of the variables, and σ_X and σ_Y are the standard deviations of the two variables. The value of ρ lies within $[-1.0, 1.0]$. When the correlation value is 1.0, it means that the variables are fully correlated. That is, for all observations in the two variables, the values of one variable can be defined by some linear function of the other variable. On the other hand, when the coefficient is -1.0, it means that the observations in one variable are negatively correlated to that of the other variable. Intuitively, Pearson Coefficient is a measure of how well the observations in the two variables match. Among the two variables used to calculate Pearson coefficient, one is the expert similarity score and, the other is a similarity score (by one of the methods described in this work). The observations in each variable are the similarity scores between the pairs of documents.

4.3 Baseline Method

The baseline we have considered, is a method proposed in [6]. In this paper, the authors have used a method called *Paragraph Link*, where the authors aim to find out whether the paragraphs of two documents are similar. They apply a hybrid of text and network

[6] We used the Doc2vec implementation from the open-source package Gensim (https://radimrehurek.com/gensim/models/doc2vec.html).

[7] Headnotes are essentially short summaries of a legal case that contains the different legal issues discussed inside it and lists out the written laws that were used during the litigation process

Portions used	TF-IDF	W2V	LDA	D2V
Whole Document	*0.62*	*0.60*	0.58	***0.69***
Summary	0.59	0.56	***0.61***	0.49
Paragraph-avg	0.51	**0.59**	0.56	**0.59**
RFC-avg	0.37	0.38	**0.51**	0.32

Table 2: Pearson Correlation Coefficients for all similarity measures w.r.t (i) the type of document representation used, and (ii) the methodology used to compute similarity between representations. The highest value in each row is in bold-font and the highest in each column is italicised. So, the best results are the one's that are both in bold-font and italicised. There are three such results. This Table is explained in details in Section 4.4.1.

based approaches. A brief description of the baseline method is as follows.

First, all the documents in the dataset are broken down into paragraphs. Then, for each paragraph in each document, we compute a paragraph vector by using the TF-IDF scores of each term present in the paragraph. Now for each pair of paragraphs in the entire data collection, we create a link if the cosine similarity between the two paragraph vectors is higher than a threshold value of 0.3 (the value used in [6]). Now, for a given pair of documents, we check the number of paragraph links. Two documents are considered similar if they share three or more common paragraph links. The similarity measure between the two documents is the number of paragraph links they share.

It is noteworthy that the dataset used by the baseline work [6] consisted of 3,866 Indian Supreme Court cases. While in our case, we have used a dataset of 33,545 cases, which is an order of magnitude larger. Hence, the number of paragraph-links for a document in our dataset is proportionately larger than that in [6], leading to much higher values in Table 3 (as compared to the number of PLs mentioned in [6]).

4.4 Results

We now describe the results of our experiments. We first investigate which of the methods discussed in this work yield the best similarity measures (i.e., has highest correlation with the judgements by legal experts). We then compare the best similarity measure with the baseline approach

4.4.1 **Selection of the appropriate document representation and similarity measure:** Given the various ways to represent a legal document and different ways to calculate similarity between them, we now look to find the most appropriate choices. Table 2 shows the performance of different similarity measuring techniques for different document representations – *Whole Document*, *Summary*, *Paragraph* and *RFC*. RFC and Paragraphs are the two methods for breaking up the text in parts. Whereas, the other two rows correspond to considering the whole text and summarisation. Each entry in the Table 2 is a correlation value corresponding

Sl.	Case Pairs	Expert Scores	Baseline Scores [6]	Whole Doc with Doc2vec
1	1992_47 & 1992_76	0	78	0.160
2	1992_76 & 1992_182	0	49	0.146
3	1972_11 & 1984_115	0	745	0.084
4	1969_57 & 1980_91	0	2303	0.271
5	1959_151 & 1982_28	0	522	0.238
6	1976_200 & 1959_151	0	583	0.051
7	1985_114 & 1959_151	0	756	0.263
8	1966_236 & 1967_267	0	322	0.353
9	1961_34 & 1979_110	0	486	0.322
10	1961_34 & 1987_37	0	146	0.193
11	1992_47 & 1987_315	0	279	0.358
12	1971_138 & 1992_47	0	645	0.459
13	1992_47 & 1992_76	0	78	0.160
14	1984_115 & 1987_315	0	390	0.238
15	1983_129 & 1983_27	1	806	0.561
16	1979_110 & 1953_28	2	327	0.178
17	1963_170 & 1979_158	2	2006	0.492
18	1983_27 & 1983_37	2	808	0.527
19	1983_27 & 1979_33	2	2420	0.581
20	1984_115 & 1981_49	2	536	0.500
21	1979_110 & 1989_233	3	356	0.351
22	1983_129 & 1976_176	5	574	0.266
23	1971_111 & 1972_291	5	1098	0.393
24	1990_171 & 1988_88	5	2058	0.297
25	1972_31 & 1984_115	5	2429	0.536
26	1984_118 & 1971_336	5	968	0.356
27	1987_154 & 1964_144	5	839	0.492
28	1973_186 & 1986_218	5	957	0.393
29	1990_96 & 1990_171	5	692	0.439
30	1958_3 & 1992_144	5	1073	0.372
31	1979_158 & 1965_111	7	618	0.529
32	1962_303 & 1972_291	7	535	0.540
33	1987_37 & 1989_233	7	32	0.234
34	1953_40 & 1953_24	7	2526	0.836
35	1966_154 & 1976_43	7	3223	0.431
36	1953_24 & 1957_52	7	3532	0.177
37	1984_115 & 1971_49	7	721	0.482
38	1980_221 & 1984_115	8	934	0.539
39	1980_39 & 1969_324	8	538	0.648
40	1991_48 & 1987_189	9	491	0.537
41	1979_104 & 1979_110	9	661	0.695
42	1985_113 & 1969_324	9	674	0.619
43	1979_33 & 1979_110	9	701	0.838
44	1968_197 & 1972_62	10	4265	0.584
45	1992_47 & 1984_115	10	487	0.540
46	1991_12 & 1985_113	10	675	0.725
47	1983_37 & 1979_33	10	2368	0.750
	Correlation		0.33	**0.69**

Table 3: Similarity measures for some pairs of document, as inferred by (i) legal experts, (ii) the baseline methodology [6], and (iii) the best methodology discussed in this work (Doc2vec over the whole document). The scores inferred by the Doc2vec method has much higher correlation with the expert scores, as compared to the baseline methodology.

		Doc2vec score	
		Similar	Not Similar
Domain Expert Score	Similar	13 out of 17 (76.5%)	4 out of 17 (23.5%)
	Not similar	5 out of 30 (16.67%)	25 out of 30 (83.33%)

Table 4: Confusion Matrix for evaluating the performance of the Doc2vec similarity methodology on the whole document. Here, the problem of identifying similar documents is modeled as a two-class classification problem, where document-pairs are to be classified as either similar or not-similar. The Doc2vec similarity achieves far better accuracies as compared to the baseline method [6] – 76.5% for documents that were judged more similar by legal experts, and 83.3% for documents that were judged less similar by legal experts.

to a similarity measure being evaluated. The correlation is measured between the calculated scores with those given by legal experts. Each column of the Table corresponds to the technique being used to represent the portions of texts as vectors, while each row corresponds to a method for selecting portions from the text or a method for breaking the text.

The highest correlation value in each row (i.e., with each representation) is in bold-font, and the highest value in each column is italicised. Hence the values in bold-font exhibit the best similarity score for a given technique for representing the document. Whereas, the italicised values show the best representation method suited for a given similarity measurement. So, the measures that are both in bold-font and in italics are the best results.

For the methods of breaking the text into parts (paragraphs and RFCs), we have discussed how the similarity measures were calculated. In the last step, we obtained multiple similarity scores for a given pair of documents. So, we had to choose a method for combining these scores to get a single score. To combine these similarity values we take an average of these similarity scores. The results of average is marked by *-avg*.

We see that when the whole document is considered, Doc2Vec (*D2V*) outperforms all the other methods. When the summary is considered, *LDA* produces the best result, while the other two methods are worse than TF-IDF. This performance is heavily dependent on the performance of the summarisation algorithm. For *Paragraph-avg*, the two neural embeddings (Word2vec and Doc2vec) produce comparable performances which are better than TF-IDF. For *RFC-avg*, the best performance is produced by LDA.

On the whole, the highest correlation value with the experts' score, i.e., *0.69*, is obtained by *D2V* (Doc2vec) for *Whole document*. So, in the forthcoming experiments we will restrict to calculating *D2V* similarity between whole documents.

4.4.2 Comparison with the baseline:

Now we compare the best performing methodology (Doc2vec over the full documents) with the baseline method [6]. We do a two-fold comparison, based on correlation with the expert judgements:

Pairwise correlation: Table 3 shows the three different similarity scores – given by (1) the legal experts, (2) the baseline method, and (3) *D2V* similarity between whole documents – for all 47 document-pairs (for which the expert judgement was obtained from [6] itself). In the last row, the Pearson correlation coefficient for each method w.r.t. the expert scores is given. The higher the value of the correlation, the better is the performance of the corresponding method. As seen in the last row of the Table, the correlation value for the baseline method is 0.33, which is significantly lower than the correlation value 0.69 produced by the proposed method. Hence, *D2V* similarity between the full documents produces far superior results over the baseline.

Overall agreement: Here we consider a two-class classification of the document-pairs. A document-pair is considered actually similar if the expert similarity score (in the range $[0, 10]$) is greater than 5, and not similar otherwise. Also, a document-pair is considered to be judged similar by the Doc2vec method (which gives scores in the range $[0.0, 1.0]$) if the computed similarity score is greater than 0.5; otherwise, the Doc2vec method is considered to have judged the pair to be not similar. Thus we get a two-class classification problem for the 47 document pairs, for which the confusion matrix is shown in Table 4.

We see that out of 17 pairs judged as similar by the experts, our method has identified 13 correctly as similar (76.5% accuracy). On the other hand, out of 30 pairs labelled as not similar by the experts, our method has successfully identified 25 as not similar (83.33% accuracy). These values are much higher than those reported in the baseline paper [6], which reported the two accuracy values to be 52% (for similar documents) and 42% (for not similar documents) respectively (see Table 3 in [6]). These high accuracy values further establish the superiority of the proposed method over the baseline.

5 CONCLUDING DISCUSSION

In this paper, we experimented with several methodologies for measuring the similarity between two legal documents. Prior to this, the text-based measures that were used for this purpose were limited to preliminary frequency-based techniques such as TF-IDF. To our knowledge, this is the first attempt at using advanced semantic techniques such as topic modeling and neural network based approaches (word embeddings and document embeddings) for measuring similarity between legal documents. We show that the advanced embedding based similarity measures, presented in this paper, significantly out-perform baseline techniques which utilize both text-based (TF-IDF) and network-based similarity measures [6].

Among the similarity measures discussed in this work, the Doc2vec similarity over the whole document was found to correlate most with the expert judgment. This result is somewhat counter-intuitive, considering that, according to legal experts, different paragraphs of a legal document tend to discuss different legal issues; hence, we had expected paragraph-based methods to perform better. However, the gold standard considered in this work was generated by legal experts looking at two full documents and estimating their similarity. This is possibly why measuring semantic similarity between the two full documents correlates best with the similarity scores assigned by the legal experts.

It should be noted that some of the methodologies discussed in this paper are more readily applicable than others. For instance, measures that use RFCs as document representations, can be used only when citation points have already been identified. Hence, such techniques would not be readily applicable when a practitioner only has a short description of a new case. Again, the performance of the similarity measures that use summaries as document representations, depends heavily on the quality of the summaries being generated. Better summaries would intuitively result in better similarity scores. So, the performance of the summary based measures can potentially be improved in future by developing better automatic summarization techniques – we plan to pursue this direction in future. On the other hand, methods like topic modeling and Doc2vec can be readily applied over the whole document, or over paragraphs of a document.

Finally, note that, though the text-based methods presented in this work perform well, they can be further improved by incorporating information from the citation network. In future, we hope to form better similarity measurement techniques by hybridizing citation information with embedding based measures.

ACKNOWLEDGEMENT

We thank the anonymous reviewers for their feedback which helped us to improve the paper. We also thank legal experts from West Bengal National University of Juridical Sciences, Kolkata, India for insightful discussions. Arpan Mandal is supported by the Visvesvaraya PhD Fellowship, Ministry of Electronics and Information Technology (MeITY), Government of India.

REFERENCES

[1] Lars Backstrom and Jon Kleinberg. 2014. Romantic Partnerships and the Dispersion of Social Ties: A Network Analysis of Relationship Status on Facebook. In *Proc. ACM Conference on Computer Supported Cooperative Work & Social Computing (CSCW).* 831–841.

[2] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *J. Mach. Learn. Res.* 3 (March 2003), 993–1022.

[3] Stefanie Brüninghaus and Kevin D. Ashley. 2001. Improving the Representation of Legal Case Texts with Information Extraction Methods. In *Proceedings of the 8th International Conference on Artificial Intelligence and Law (ICAIL '01).* ACM, New York, NY, USA, 42–51. DOI:http://dx.doi.org/10.1145/383535.383540

[4] Filippo Galgani, Paul Compton, and Achim Hoffmann. 2012. *Towards Automatic Generation of Catchphrases for Legal Case Reports.* Springer Berlin Heidelberg, 414–425.

[5] Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Aditya Singh. 2011. Similarity Analysis of Legal Judgments. In *Proc. ACM Compute Conference.* 17:1–17:4.

[6] Sushanta Kumar, P. Krishna Reddy, V. Balakista Reddy, and Malti Suri. 2013. *Finding Similar Legal Judgements under Common Law System.* 103–116.

[7] Quoc Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. In *Proc. International Conference on Machine Learning (ICML)*, Tony Jebara and Eric P. Xing (Eds.). JMLR Workshop and Conference Proceedings, 1188–1196.

[8] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. *CoRR* abs/1301.3781 (2013). http://arxiv.org/abs/1301.3781

[9] Akshay Minocha, Navjyoti Singh, and Arjit Srivastava. 2015. Finding Relevant Indian Judgments Using Dispersion of Citation Network. In *Proc. International Conference on World Wide Web (WWW) Companion.* 1085–1088.

[10] Paul Zhang and Lavanya Koppaka. 2007. Semantics-based Legal Citation Network. In *Proceedings of the 11th International Conference on Artificial Intelligence and Law (ICAIL).* 123–130.