IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 2, No. 1, March 2013, pp. 27~35
ISSN: 2252-8938

Legal Documents Clustering and Summarization using Hierarchical Latent Dirichlet Allocation

Ravi kumar V*, K. Raghuveer**
* Reseaarch Scholar, Departement of Information Science and Engineering, The National Institute of Engineering, Mysore, India
** Faculty, Departement of Information Science and Engineering, The National Institute of Engineering, Mysore, India

Article Info

Article history:

Received Jul 30, 2012
Revised Oct 27, 2012
Accepted Jan 07, 2013

Keyword:

Latent Dirichlet Allocation (LDA)
hierarchical Latent Dirichlet Allocation (hLDA)
Legal Documents Clustering
Similarity measure
Legal Document Summarization.

ABSTRACT

In a common law system and in a country like India, decisions made by judges are significant sources of application and understanding of law. Online access to the Indian Legal Judgments in the digital form creates an opportunities and challenges to the both legal community and information technology researchers. This necessitates organizing, analyzing and presenting it in a useful manner to the legal community for quick understanding and for taking necessary decision pertaining to a present case. In this paper we propose an approach, to cluster legal judgments based on the topics obtained from hierarchical Latent Dirichlet Allocation (hLDA), using similarity measure between topics and documents and to find the summarry of each document using the same topics. The developed topic based model, is capable of grouping the legal judgments into different clusters and to generate summary of each legal judgment in the cluster, in effective manner compare to our previous approach [1].

Corresponding Author:

Ravi kumar V,
Reseaarch Scholar, Departement of Information Science and Engineering,
The National Institute of Engineering, Mysore, India

1. INTRODUCTION

Legal documents clustering and summarization has become a helpful tool for the legal community and students in organizing, understanding the previous case in a quick time and also to relate any previous case decision to the present case if necessary.

Information retrieval (IR) is the area of study concerned with searching for documents, for information within documents, and for metadata about documents, as well as that of searching structured storage, relational databases, and the World Wide Web [2].

The critical part of any Information Retrieval (IR) approach is the representation of the document content. Normally, the documents are represented as 'bag of words', it means that the words are assumed to be occur independently. Many researchers have given different approaches to group the words into "topics", such that each group represents the important relationships between words with in that group. Techniques such as word clustering and document clustering have been used for many years to enhance document representations [3]. The technique of clustering words or terms was studied in 60s by [4]. The Latent Semantic Indexing (LSI) a well known IR technique based on the reduced vector space was introduced in 1990 by [5]. Later in 1999, Hoffman has proposed a new approach for IR called the probabilistic Latent Semantic Indexing (pLSI) [6]. This approach uses a latent variable model that represents documents as mixtures of topics.

Latent Dirichlet Allocation is one of the recent topic models to represent a document as mixture of topics proposed by [7], using machine learning techniques. It has been considered as one of the most popular

probabilistic text modeling techniques in machine learning. The role of LDA in IR is to cluster the documents based on the topics that improves the effectiveness of the retrieval.

In all the above three clustering techniques, it is not clear about how many number of clusters must considered for the given text corpus. The efficiency of these approaches can be measured, by performing the clustering repeatedly, by varying the number of clusters and observing the scores of precision, recall and perplexity. To deal with the issue of variable clusters, Blei et al. proposed the hierarchical Latent Dirichlet Allocation (hLDA) based on principles of nonparametric Bayesian techniques [8].

Document clustering is an unsupervised approach to group similar documents, such that the documents with in the cluster are more similar to each other and the documents across the cluster are less similar to each other. This can be achieved by representing the document in the reduced dimensionality as mixture of topics using hLDA and then clustering the documents by using the similarity measure between generated topics for the given text corpus and the documents in the corpus.

Summarization is important for legal community, because their usual practice is to reading summaries (headnotes) instead of reading entire judgments. A headnote is a summary of the key legal points that is added to the text of a court decision, to aid readers in interpreting the highlights of an opinion. Once the documents are clustered, the summary can be generated automatically for each document within the cluster using the topics used for clustering those documents.

2. RELATED WORK

The straight forward method of texts clustering model is based on explicit syntax. On the other hand, we can capture both syntax and latent semantics using topic models. The role of hierarchical topic models regarding text, is to identify and partition a document, into topics and arranging them into hierarchy based on their semantic relation, is important for many Natural Language Processing (NLP) tasks, including information retrieval, summarization, and text understanding. More accurate and predictive modeling of topics can be achieved using the hierarchical model, than the one constructed by flat models.

Freddy [9], gave an extended version of LDA, to analyze customer reviews to find the summary of the customers opinion. He extended the flat clustering algorithm of LDA to tree like hierarchical clustering using hLDA. The root of the tree represents topic terms that are more common to all documents, usually stop words and the terms at the bottom of the tree are of specialized topics. He also explained different probabilistic topic models, used for text document clustering, using reduced dimensionality in [10].

Elias Zavitsanos et al. [11] gave a nonparametric Bayesian priors approach of modeling the content of a given document collection, as a hierarchy of latent topics, given no prior knowledge. These topics represent and capture aspect of content meaning, by means of multinomial probability distributions, over the words of the term space of the documents. The assignment of documents to latent topics without any pre classification is a powerful text mining technique, useful among others, for ontology learning from text and document indexing.

Adler Perotte et al. [12] introduced hierarchically supervised latent Dirichlet allocation (HSLDA), a model for hierarchically and multiply labeled bag-of-word data. Examples of such data include web pages and their placement in directories, product descriptions and associated categories from product hierarchies, and free-text clinical records and their assigned diagnosis codes.

E. Gaussier et al. [13] proposed a hierarchical generative model for textual data, where words are grouped to form a topic using co occurrence basis and are arranged in a hierarchically to cluster and categorize the documents.

Qiang Lu et al [14] describes American legal documents clustering on a large scale, with soft clustering algorithm, based on topic-segmentation, using Meta data of the legal documents.

William M.et al. [15] 'PathSum' approach generates automatic text summarization for single document and multi documents, using hierarchical topic generated from hLDA. The summarization was the group of sentences, which travels in the same path of the hierarchical tree from root to leaf.

Ben Hachey et al. [16] developed a system for summarization of legal judgment, based on the rhetorical structure of the argument of the case. They have used Weka toolkit to train the machine for different features of the legal judgment and based on this the summary was generated.

Atefeh Farzindar et al. [17] gave an approach to legal documents (proceedings of the federal court of Canada) summarization by discovering the document's architecture and its thematic structure, to provide a table style summarization, that help the legal community, to understand the case, just by reading the summary, instead of reading the entire judgment.

Claire Grover et al. [18] show the use of rhetorical and discourse structure at sentence level of the legal cases, for finding the main verbs. The technique was based on [19], where sentences were classified according to their argumentative role.

3. OUR APPROACH

The main task has been divided into two subtasks, the first task is to cluster the legal judgments and second task is to find in summary for each of the judgment within the cluster, using topics tree generated from hLDA for the given corpus. The architecture of the proposed approach, to cluster legal judgments and to generate summary for each legal judgment in the cluster, using hLDA topic module is given in figure 1.0. The following steps are involved in this process.

1. Preprocessing of legal judgments to remove stop words and to consider legal terms.
2. Generation of topics tree, for the given legal judgment corpus, using hLDA topic model.
3. Legal judgments clustering using the topics tree, obtained from hLDA topic model for the give corpus.
4. Finding the sentence score, for each sentence in the legal judgment, present in the cluster, using the topics of that cluster.
5. Generation of extraction based legal judgment summary for each judgment present in the cluster.

3.1. Legal Judgments

The legal judgments present in the corpus are the one, used in our previous approach [1]. The corpus consists of Legal judgments pertaining to various Indian civil case text documents collected from [20]. This legal judgment corpus is used as input to the proposed system.The steps involved in the process clustering and generating extraction based legal judgment summary for the given corpus, are explained in detail in the following sections.

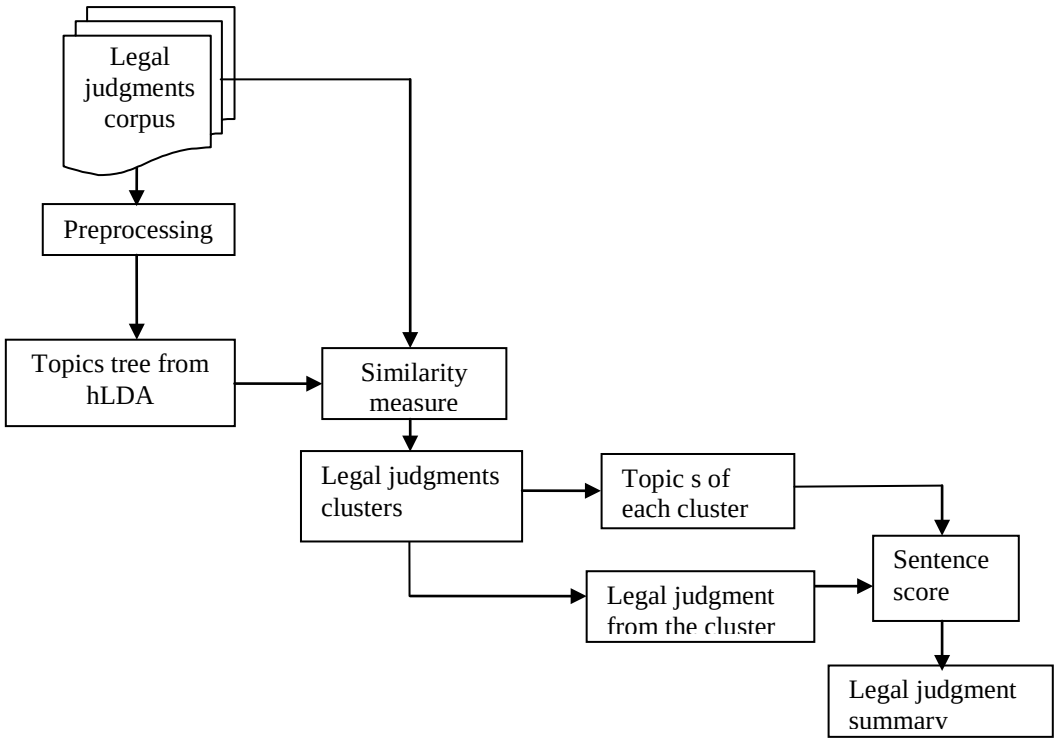

Figure 1.0. Architecture for legal judgment clustering and summarization using hLDA topic model.

3.2. Preprocessing

The judgment part of the Legal judgment document is similar to other documents consists of stop words like is, of, an, etc. We remove these stop words to avoid in getting these stop words as topic terms. Similarly there are various legal terms they are common in all most all types of legal judgments documents and give no information about the case, such terms are listed in consult with legal experts using legal judgments corpus. These terms considered as stop words and are removed from the input documents.

3.3. Generation of Topics Tree from hLDA for Legal Judgment Corpus

The hierarchical Latent Dirichlet allocation (hLDA) described by Blei et al. in [8]. In the basic LDA model [7], the topic generated by learning the vocabulary is flat; such that there is no relation between each

other is stated. On the other hand in hLDA allows us to determine the topics for the given corpus and represent the relation between each topic in a hierarchical tree. The tree for the hLDA is learned through a non parametric Bayesian approach, where there is no priori is set regarding the number of topics and the structure of the tree, but is determined directly from the data through posterior inference [8].

In the tree topics are arranged in a hierarchy using the nested structure such that the topics at the root are more general and more specific at the bottom. For the experimental purpose we have used java implementation of Mallet hLDA [21], where we can specify the maximum depth of the tree and number of iterations for the Gibbs sampling. Note that in our case instead of having tree of infinite depth as described in hLDA model [8], we have conducted experiments with different depth (level) and we found that the depth of eight is more efficient, hence we have chosen topic tree of depth/level eight.

3.4. Legal Judgments Clustering Using Topics Tree

Let D = $\{d_1...,d_N\}$ denote the set of documents to be cluster. K= $\{k_1,\ldots,k_M\}$ are the topics at the leaf nodes of the topics tree obtained from hLDA for the specified depth/level, of the given corpus D. As we know that the topic terms at the root are more general to the given corpus, the topic terms at the bottom are more specific to certain documents in the corpus and the intermediate nodes along the path from root to leaf represents the multiple subtopics of the documents of the given corpus. To cluster legal judgments we have considered the topics from node root-1 to leaf node, because the topic at the root represents terms common to entire corpus.

The number of cluster varies as we vary the sampling value for Gibbs Sampling. The common approach is to represent the documents to be clustered using vector-space model. A vector contains items from textual space, such as terms. We have considered two similarity measures for clustering purpose.

3.4.1. Cosine Similarity

The cosine similarity is applied to compute the similarity between two vectors x1 and x2 in the vector-space model, Cosine similarity is one of the most accepted similarity measure applied to text documents, such as in many information retrieval applications [22] and clustering too [23]. It is defined to be

$$\cos(x1, x2) = \frac{(X1 \cdot X2)}{||X1|| \times ||X2||}, \qquad 1.0$$

Where, ||x|| is the vector length. As a result, the cosine similarity is non-negative and bounded between [0, 1]. It is 0 when the objects are identical and 1 when they are totally different.

3.4.2. Jaccard Coefficient

The Jaccard coefficient occasionally referred to as the Tanimoto coefficient, measures similarity as the intersection divided by the union of the objects. For text document, it compares the sum weight of shared terms to the sum weight of terms that are present in either of the two documents but are not the shared terms. It is formally defined as:

$$SIM_J(\overrightarrow{t_a}, \overrightarrow{t_b}) = \frac{\overrightarrow{t_a} \cdot \overrightarrow{t_b}}{|\overrightarrow{t_a}|^2 + |\overrightarrow{t_b}|^2 - \overrightarrow{t_a} \cdot \overrightarrow{t_b}}. \qquad (2.0)$$

The Jaccard coefficient is a similarity measure and ranges between 0 and 1. It is 1 when the objects are identical and 0 when they are totally different.

We have considered number of cluster is equal to the number of leaves or the number of paths from root to leaf of the specified depth. We find the similarity between each document and topics from leaf node to the topics at root-1 node in different paths, using cosine and Jaccard coefficient. The document is placed into a cluster which is very close to a one of these topics terms. Once all the documents are placed into different clusters based on the similarity measure, we maintain the topics related to each cluster, for the purpose of each legal judgment summary present in the corpus.

3.5. Sentence Score Calculation

Once the documents are clustered using the topics tree obtained from hLDA topic model, we have topics with respect to each cluster. Consider each judgment from the cluster and find the sentences present in the document using Sentence Boundary method [24], to find the sentence score. The procedure to find the sentence score for each sentence is shown in figure 2.0.

Consider the sentences S_r, r Є {1,..., R} for each document in the cluster and the topics with respect to that cluster, in that path from the root-1 to the leaf node T_{jp}, j Є {1,...,K}, p Є {1,...,n}, Where K is the

number of leaf nodes or the paths of the topic tree obtained by running hLDA and p represents the sub topics of the documents belong to that cluster. Let the words of the sentence S_r be $\{W_1, W_2, ... Wq\}$.

Procedure Sentence_Score (document d_i, topic p)

```
//Input:  di  //document from a cluster, whose sentences score to be find.
          P // sub topic P of the topic Tj  w.r.to that cluster, represents topic terms.
//Output:   Sip  = {s1, s2,…,sm}      // Sentence score for each sentence in the input document, w.r.to
                                      //the sub topic.
1. for each sentence sr ∈ di do
2. Si = 0                           // initialize sentence score of ith sentence of  document di  to zero.
3. ∀Wq ∈ Sr                         // consider each word from the sentence.
4. if (Wq ∈ P) then                 // to check whether the word occurs in sub topic or not.
5. Si = Si +1                       // if the word occurs then add 1 to the sentence score.
6. endfor  // end of sentence score calculation for each sentence w.r.to.the  sub topic P.
7. return Sip
```

Figure 2.0. Procedure to find sentence score each sentence using topic terms.

Consider each document from the cluster as input to the procedure and for each sentence in the document, find the occurrence of each word in the sentence with respect to each topic within the cluster and consider this as the score of that sentence. Finally the sentence score of each sentence of the input document is returned as output.

3.6. Generation of Extraction Based Legal Judgment Summary

Extraction based summary is the condensed version of the original document. The condensed version should cover, the main topics discussed in the document. In this approach we have considered the topics obtained from hLDA for the given corpus for a particular cluster. The topics considered here have hierarchical relation between each other, representing the subtopics discussed in the documents belongs to a particular cluster. The algorithm to find the summary of each legal judgment of the corpus, belongs to different clusters is given in figure 3.0. Once we get the sentence score for each sentence in a document using the above said algorithm, the next step is to find the summary, consisting of maximum of two sentences from each topic. The algorithm to find the summary is given in figure 3.0.

Algorithm Judgment_Summary ()

```
    //Input: D={D1, D2,…,Dk} //legal judgments cluster for summarization.
             Tj ={ T1, T2,…,Tk }    //Topics from hLDA for each cluster.
  //Output:    Summary= {sm1, sm2,…, smm}  //  Summary of the each document  in the cluster.
1. for each cluster Di ∈ Dk  // consider each cluster one by one.
2. for each document di ∈ Di do //consider each document in the cluster for summary.
3. for each sub topic P ∈ Tj  do // consider sub topics P of the topic Tj  w.r.to that cluster.
4. Sip=Sentence_Score (di, P) // call the procedure to find sentence score of ith document w.r to pth sub topic.
5. Arrange the sentences in the descending order based on the sentence score
6. endfor  // end of calculation of sentence score w. r. to, each sub topic P ∈ Tj  .
7. for each L from 1 to P  do //  P represents the number of sub topics of Tj considered for summary.
8. Select top 2 sentences from Sil whose score is greater than or equal to average score
   considering all the sentence  in that document // here i represents ith document.
9. if (any of the sentence appears already with respect to previous topic)then
10. Select the next sentence.
11. endfor // end of extraction of sentences w.r.to each sub topic P ∈ Tj  .
12. Arrange the sentences according to the sentence number of di to smi  // summary of the ith  document.
13. endfor // end of summary generation of all the documents in the cluster.
14. endfor // end of all the clusters.
```

Figure 3.0. Algorithm to find legal judgment summary using hLDA topics.

Once the documents are clustered, we consider each legal judgment from the cluster as input to the algorithm. The sentence score for each sentence in the legal judgment is calculated using the procedure Sentence_Score with respect each topic of that cluster. The top 2 sentences with respect to each topic are selected for final summary by eliminating redundancy. After the sentences are extracted, they are arranged in the ascending order according to sentence number of the original document. This summary is the extraction based summary of the legal judgment,

4. EXPERIMENTAL SETUP

4.1. Dataset

The dataset used for the experiments are similar to the one used in our previous approaches [1], consists of documents from various sub domains of civil cases in India and are collected from [20]. These documents are part of the corpus of 250 Legal judgments documents of different domains. The documents in the dataset consisting of judgments are dated up to the May 2012. The judgments belongs to different sections like Sales Tax, Rent Control, Motor Vehicle, Family Law, Patent, Trademark and Company law, Taxation, Property and Cyber Law, etc.

4.2. Parameter Selection

As we mentioned earlier, instead of having topic tree for hLDA of infinite depth, we have conducted experiments with different depth and we found that the depth of eight is more efficient, hence we have chosen topic tree of depth eight. In addition to this we have chosen smoothing parameters $\alpha = 10$, $\beta = 0.1$ and γ=1.0 for nCRP with number of sampling as 200.

5. EXPERIMENTS AND RESULTS

5.1. Experiment-1

Experiment has been conducted to cluster the legal judgments using Cosine and Jaccard similarity between the legal judgments and topics obtained by running hLDA with level L=5, 6, 7, 8 and 9. It has been observed that with the level L=8, the topic distribution gives effective result compared with the other values for L. The table 1 gives the number of documents specified at the leaf of hLDA topic tree, sharing the topics from root to leaf and the number of documents clustered based on the similarity between the documents and topics from root-1 to the leaf using Cosine and Jaccard similarity for the level L=8.

Table 1. The number of documents at the leaf of hLDA topic tree and the number of documents clustered using Cosine and Jaccard similarity for the level 8.

Path from root to leaf node	documents at the leaf	clustered documents using Cosine similarity	clustered documents using Jaccard similarity
1	8	8	11
2	15	14	16
3	9	8	7
4	35	36	34
5	6	6	5
6	1	1	1
7	2	1	1
8	27	28	26
9	7	6	6
10	2	2	3
11	3	5	5
Total	115	115	115

In the table 1 the column 1 represents the various paths, obtained from hLDA for the corpus of size 115 documents, specified depth of 8. The column 2 represents the actual number of documents sharing the topics, from the root to the leaf node, specified at the leaf node, according to hLDA. The column 3 represents the number of documents clustered, using the cosine similarity between documents and topics with respect to that path. The column 4 represents the number of documents clustered, using the Jaccard similarity between documents and topics with respect to that path.

5.1.1. Accuracy

To find the accuracy of the clustering at each level, we have considered misclassification as the evaluation criteria and the result is given in table 2, for different levels.

$$\text{Misclassification} = |D_{hLDA} - D_{Cosine|Jacard}|$$

Where D_{hLDA} is the total number of documents at each leaf node according to hLDA topic tree and $D_{Cosine|Jacard}$ is the total number of documents clustered based on the topics from root-1 to leaf node using Cosine and Jaccard similarity measure.

Table 2. Shows the documents clustering accuracy based on the total misclassification at different level.

Sl.No	Level L	Total Misclassification using Cosine Similarity	Total Misclassification using Jaccard Similarity
1	5	37	35
2	6	17	33
3	7	49	47
4	8	17	19
5	9	30	28

From the result, we can observe that, for the level 8, the misclassification is less compare to other levels and hence we have chosen topics at level 8 for clustering and for summarization.

5.2. Experiment-2

Experiment has been conducted to generate extraction based legal judgment summary for each legal judgment present in different clusters. As we mentioned in the previous experiment, we have considered topics belongs to each cluster at level 8, to generate legal judgment summary. The performance of our system to generate legal judgment summary has been evaluated by comparing the system generated summary with legal experts generated summary as reference summary used in [1].

To evaluate the results we have used precision, recall and F-measure that are commonly used in information retrieval tasks. The precision, recall and F-measure are calculated using the equation 3.0, for each document using manually extracted summary denoted as S_{ref} and system generated summary denoted as $S_{sys.}$ Table 3, shows the mean scores of recall, precision and F-measure of the summary generated.

$$P = \frac{|Sref \cap Ssys|}{Ssys} \qquad R = \frac{|Sref \cap Ssys|}{Sref} \qquad F1 = \frac{2 * P * R}{P + R} \tag{3.0}$$

The obtained result for summarization has been compared with our previous approach [1] and table 4, shows the results comparisons. The comparisons shows that, there is an increase both in precision and recall compared to our previous approach, because of the reason that, in our previous approach we have taken the same set of topics for all the documents to generate summary, but here we have considered the most appropriate set of topics for each documents based on the hLDA topic tree.

Table 3. The mean scores of recall, precision and F-measure of the summary generated.

Domain	**Precision**	**Recall**	**F-Measure**
Income Tax	0.623	0.607	0.614
Rent control Act	0.609	0.587	0.597
Motor Act	0.571	0.561	0.565
Negotiable Instrument Act	0.554	0.536	0.544
Sales Tax	0.563	0.582	0.572

Table 4. Comparison of result for summary generation with the approach [1]

Domain	Precision		Recall		F-Measure	
	LDA[1]	hLDA	LDA[1]	hLDA	LDA[1]	hLDA
Income Tax	0.604	0.623	0.587	0.607	0.595	0.614
Rent control Act	0.589	0.609	0.568	0.587	0.578	0.597
Motor Act	0.551	0.571	0.542	0.561	0.546	0.565
Negotiable Instrument Act	0.526	0.554	0.513	0.536	0.519	0.544
Sales Tax	0.532	0.563	0.553	0.582	0.542	0.572

6. CONCLUSION AND SCOPE FOR FUTURE

We made an attempt to cluster Indian Legal Judgments using hLDA topic model. With the use of hierarchical approach (hLDA), we are able to get better results in generating summary for the given Indian Legal judgment, when compared to our previous approach [1]. This can be further improved by segmenting the Legal Judgment that matches the rhetorical structure of the legal document and generating the summary based on these segments that gives summary which is more structural.

ACKNOWLEDGMENT

We would like to thank Mr. Shivakant Aradya, advocate, for the assistance and guidance given to us in preparing legal summary and legal fraternities Miss. Depu and her colleague for their help in preparing the reference summary

REFERENCES

[1] Ravi kumar V and K Raghuveer. "Legal Document Summarization using Latent Dirichlet Allocation". *International Journal of Computer Science and Telecommunications*. 2012; 3(7): 114-117.
[2] http://en.wikipedia.org/wiki/Information_retrieval
[3] Xing Wei and W Bruce Croft. "*LDA-Based Document Models for Ad-hoc Retrieval*". In Proc. of SIGIR'06, Seattle, WA, USA. 2006.
[4] Sparck Jones. "K Automatic keyword classification for information retrieval". Butterworths, London. 1971.
[5] Deerwester S, Dumais ST, Furnas, GW, Landauer TK and Harshman R. "Indexing by latent semantic analysis". *Journal of the American Society for Information Science*. 1990; 41(6): 391-407.
[6] Thomas Hofmann. "*Probabilistic Latent Semantic Indexing*". In Proc. of the Twenty-Second Annual International SIGIR Conference on Research and Development in Information Retrieval, 1999.
[7] DM Blei, AY Ng, and MI Jordan. "Latent Dirichlet allocation". *Journal of Machine Learning Research*. 2003; 3: 993-1022.
[8] D Blei, T Griffiths, and M Jordan. The nested Chinese Restaurant Process and Bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*. 2010.
[9] Freddy Chong Tat Chua. "Summarizing Amazon Reviews using Hierarchical Clustering". *Technical report*. 2009.
[10] Freddy Chong Tat Chua. "Dimensionality Reduction and Clustering of Text Documents". *Technical report*. 2009.
[11] Elias Zavitsanos, Georgios Paliouras. "Non-Parametric Estimation of Topic Hierarchies from Texts with Hierarchical Dirichlet Processes". *Journal of Machine Learning Research*. 2011; 12: 2749-2775
[12] Adler Perotte Nicholas Bartlett, et al. "Hierarchically Supervised Latent Dirichlet Allocation". *Neural Information Processing Systems*. 2011.
[13] E Gaussier, C Goutte, et al. "*A hierarchical model for clustering and categorizing documents*". In Proc. of European Colloquium on IR Research (ECIR-02), Mar. 25-27, 2002.
[14] Qiang Lu, William Keenan, Jack G. Conrad and Khalid Al-Kofahi. "*Legal Document Clustering with Built-in Topic Segmentation*". In Proc. of CIKM'11, Glasgow, Scotland, UK. 2011: 383-392.
[15] William M Darling and Fei Song. "*PathSum: A Summarization Framework Based on Hierarchical Topics*". 2011.
[16] Ben Hachey and Claire Grover. "*Sequence Modelling for Sentence Classification in a Legal Summarization System*". In Proceedings of the 2005 ACM Symposium on Applied Computing.
[17] Atefeh Farzindar and Guy Lapalme. "Legal Texts Summarization by Exploration of the Thematic Structures and Argumentative Roles". *In Text Summarization Branches Out Conference held in conjunction with ACL*. 2004: 27-34.

[18] Claire Grover, Ben Hachey, and Chris Korycinski. "Summarising legal texts: Sentential tense and argumentative roles". *In HLT-NAACL 2003 Workshop: Text Summarization (DUC03)*, Edmonton, Alberta, Canada. May 31 – June1, pages 33-40.
[19] Simone Teufel and Marc Moens. "Summarising scientific articles - experiments with relevance and rhetorical status". *Computational Linguistics*. 2002; 28(4): 409-445.
[20] http://www.keralawyer.com/asp/sub.asp?pageVal=judgements.
[21] http://mallet.cs.umass.edu/dist/ mallet-2.0.7.tar.gz
[22] RB Yates and BR Neto. *Modern Information Retrieval*. ADDISON-WESLEY, New York. 1999.
[23] B Larsen and C Aone. *Fast and effective text mining using linear-time document clustering*. In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 1999.
[24] http://alias-i.com/lingpipe/demos/tutorial/sentences/.