Summarization of Legal Texts with High Cohesion and Automatic Compression Rate

Mi-Young Kim, Ying Xu, and Randy Goebel

Department of Computing Science, University of Alberta,
AB T6G 2E8 Canada

{miyoung2, yx2, rgoebel}@ualberta.ca

Abstract. We describe a method for extractive summarization of legal judgments using our own graph-based summarization algorithm. In contrast to the connected and undirected graphs of previous work, we construct directed and disconnected graphs (a set of connected graphs) for each document, where each connected graph indicates a cluster that shares one topic in a document. Our method automatically chooses the number of representative sentences with coherence for summarization, and we don't need to provide a priori, the desired compression rate. We also propose our own node/edge-weighting scheme in the graph. Furthermore, we do not depend on expensive hand-crafted linguistic features or resources. Our experimental results show our method outperforms previous clustering-based methods, including those which use TF*IDF-based and centroid-based sentence selection. Our experimental results also show that our method outperforms previous machine learning methods that exploit a variety of linguistic features.

Keywords: Information Extraction, Summarization, legal case, graph representation

1 Introduction

Text summarization is the process of automatically creating a compressed version of a given text, which provides useful summary information for human readers. Of course the information content of a summary depends on a user's needs. Topic-oriented summaries focus on a user's topic of interest, and extract information from the text that is related to the specific topic. On the other hand, generic summaries try to cover as much of the information content as possible, preserving the general topic organization of the original text. Up to now, many domain-specific summarization systems have been proposed, all of which depend on their selected linguistic and sentential features specific to each domain. These systems have the limitation that new features must be manually devised whenever a new domain is targeted. Here we focus on single-document extractive legal text summarization which does not depend on hand-crafted expensive linguistic features: our application is in the domain of legal texts, where the goal is to produce a summary of legal judgments from about five Lord judges for each case [13].

In the legal domain, many kinds of summarization have been proposed. For example, [23] proposed CRF-based legal document summarization, and [2, 6] explored thematic structures and argumentative roles for summarization. There were also citation-based summarization [8], but it has a limitation that we first have to identify the citation component in the legal text, which is another research field. [14] provided a supervised machine learning algorithm based on manually annotated rhetorical structure information.

We can divide the methods of choosing representative sentences in summarization into three types: TF*IDF-base, centroid-base, and lastly, graph-based selection [12, 20]. TF*IDF-based summarization selects sentences with high TF*IDF value, and centroid-based summarization is to select sentences close to the centroids of clusters. Graph-based summarization selects key sentences based on the node value of a graph representation, and then selects other sentences linked to the key sentences based on the edge values. From those alternatives, we pursue a graph-based sentence selection method. Since that approach does not require language-specific linguistic processing beyond identifying sentence and word boundaries, it can also be applied to other languages, e.g., Brazilian Portuguese, and Persian [25]. Graph-based approaches have been shown to work well for both single document and multi-document summarization [5,17].

Our method creates directed graphs, and we use our own asymmetric edge weight as opposed to the commonly used symmetric cosine similarity measure. Our methods have two strengths: First, we do not need to provide a target compression rate for summarization. In previous methods, users have to determine the compression rate, and the same rate should be applied to every document. We know that each individual document's compression rate can be different. Thus, we propose a sentence selection method that determines a compression rate automatically.

Second, previous methods choose the representative sentences according to the rank of each sentence, measured to cover diversity of topics in a document, so the results of summarization typically have less cohesion. In contrast, we choose sentences with high cohesion in each connected graph based on connected edges. Our graph for a document is an unconnected graph (set of connected graphs), and we can ensure diversity by choosing sentences from each connected graph. The chosen sentences still ensure compression rates very close to those of human-constructed abstracts compression.

The rest of the paper is organized as follows. Section 2 explains our summarization method based on directed and unconnected graph representation, and Section 3 shows the experimental results and discussion. Section 4 analyzes previous work, and finally Section 5 concludes with a summary and future work.

2 Graph-based Summarization

The summarization of legal case decisions include not only the decision sentence, but also supporting sentences which show the rules and proofs that support the decision. A graph representation is a useful tool for summarization of legal judgments, because we can trace supporting sentences (nodes) starting from the decision sentence (node) following the edge information between nodes in the graph.

For text ranking, we propose an automatic and unsupervised graph-based ranking algorithm that gives improved results when compared to other ranking algorithms.

Detecting topic words is one important step in our method, which is a component of the weight measure of graph edges. In the following subsections, we explain the details of how we detect topic signature words and create graph representations.

2.1 Detection of topic signature words

For the detection of topic signature words, we use TF*IDF[32]. The approach is likely to score longer sentences higher, simply because they contain more words. So, we normalize TF*IDF by dividing by a count of the maximum occurrence of a word in the document.

This weighting exploits counts from a corpus which serves as indication of how often a word may be expected to appear in an arbitrary text.

The normalized TF*IDF that we use is computed as follows:

$$normalized\_TFIDF(w,d) = \frac{\log(tf(w,d)+1)\cdot\log\frac{|D|}{df(w)}}{\max\{tf(x,d) : x \in d\}},$$

where *df(w)* is a document frequency of a word *w*, and *tf(w,d)* is a term frequency of a word *w* in a document *d*. For the normalization, we divide the *tf***idf* by the maximum term frequency of any word *x* in the document *d*.

The TF*IDF weights of words are good indicators of importance, and they are easy and fast to compute. These properties explain why TF*IDF is incorporated in one form or another in most current systems.

2.2 Graph representation for a document

Indicator representation approaches do not attempt to interpret or represent the topics discussed in the input. Instead they construct a representation of the text that can be used to directly rank sentences by importance [18]. Graph methods are unique because, in their most popular formulations, they base summarization on a single indicator of importance, derived from the centrality of sentences in a graph representation of the input [5]. In contrast, other approaches employ a variety of indicators and combine them either heuristically or using machine learning to decide which sentences are worthy of inclusion in the summary. Using a graph model, we reduce reliance on heuristics and manually obtained linguistic features.

In the previous graph models inspired by the PageRank algorithm [5, 17], the input is represented as a highly connected graph. Vertices represent sentences and edges between sentences are assigned weights equal to the similarity between the two sentences. The method most often used to compute similarity is cosine similarity with TF*IDF weights for words. Previous systems assume sentences that are related to many other sentences are likely to be central and would have high weight for selection in the summary.

We also use a graph-based representation of a document, and we propose our own weighting measure for edges and nodes, instead of the commonly used cosine similarity measure.

To establish connections (edges) between sentences, previous work defines a 'similarity' relation where 'similarity' is measured as a function of content overlap. Our connection weight is more like an “embedding probability”. That is, we measure how likely the content of one sentence is embedded in another sentence. Previous systems such as LexRank [5] and K. Sankar [22] use a symmetric similarity measure, and draw undirected and connected graphs, but we use this notion of embedding probability as a connection measure between two sentences and so create a directed graph. Our graph is a set of connected graphs, and the overall graph may not be connected.

The measure of weight for an edge between two vertices is as follows:

$$E(v_i, v_j) = \frac{\sum_{w \in d_i, d_j} (normalized\_tfidf(w, d_i)}{\sqrt{\sum_{x \in d_i} (normalized\_tfidf(x, d_i))^2}},$$

where $E(v_i,v_j)$ shows the degree to which words in document d_i are embedded in document d_j. In the denominator of the function, we use the length of sentences only in document d_i weighted by the term *tfidf*. The proposed edge function shows different results between $E(v_i,v_j)$ and $E(v_j, v_i)$, and it detects how likely the content of a sentence is embedded in another sentence. We choose the edges only above the threshold 0.4, which is determined by many experiments using variations of the threshold. Here we show one example document as written in Figure 1. The graph for the example document is shown in Figure 2. We can see three connected graphs in Figure 2.

Previous systems assign the summation of the weights of linked edges of a node as the weight of a node. They then choose nodes in the order of high weights for summarization. As a result, the constructed summarization shows less coherence. Furthermore, their systems have a limitation that the same rate of sentences is chosen for summarization of each document.

Instead of using the summation of edge weights for the weight of a node, we use a common sense notion that the conclusion is the sentence which includes the words “agree” or “dismiss”. For summarization of judgments, the most important sentence is the decision of the judgment, which is typically included in the conclusion. To indicate how important a sentence is for summarization, we first consider how likely words in a sentence appear in the conclusion. From the chosen sentence which includes words that appear often in the conclusion, we choose another cohesive sentence based on our graph representation's directed links.

To choose a representative node, we first compute the key value of each word in the sentence corresponding to the node as follows:

$$Weight(v_i) = \max\{key(w) : w \in d_i\},$$
$$key(w) = \sum_i \left(\frac{tf(w, d_{i,last\_sen})}{tf(w, d_{i,\neg last\_sen})} \cdot \frac{\max\{tf(x, d_{i,\neg last\_sen} : x \in d_{i,\neg last\_sen}\}}{\max\{tf(x, d_{i,last\_sen} : x \in d_{i,last\_sent}\}}\right),$$

where *key(w)* is the key weight of word *w* that indicates how likely *w* appears in the last sentence. $d_{i,last\_sen}$ is the modified document *di* including only the last sentence in each document, and $d_{i}, _{\neg last\_sen}$ is another modified document of *di* excluding last sentence. We use normalized term frequency for the weight. If $tf(w, d_{i,\neg last\_sen})$ is zero, then we assign 1.

We determine the lower of the key values of two words "agree" and "dismiss" as the threshold. We know that the conclusion of the law judgment is one of two cases: "Agree" or "Dismiss". So, we determine the threshold of *key(w)* based on the key values of the words "agree" and "dismiss". Then, we choose the conclusive sentences that include a word whose key value is over threshold, which means $Weight(v_i)$. From the conclusive sentences, we extend summaries by including the supporting sentences that explain facts, proofs, or rules following directed links.

2.3 Sentence selection for summarization with cohesion

After the graph construction, we select sentences according to their representativeness and coherence. Linguistic coherence makes the text semantically meaningful. A text has meaning as a result of combining meaning or senses in the text. The coherence cues present in a sentence are directly visible when we go through the flow of the document. Our approach aims to achieve this objective with graph information.

The strategy for summary selection – one sentence per important topic - is easy to implement but possibly too restrictive. The question that stands out is that maybe for some topics more than one sentence should be included in the summary. Other sentence scoring techniques for lexical chain summarization have not been explored, i.e., sentences that include several of the highly scoring chains may be even more informative about the connection between the discussed topics.

In order to avoid problems with coherence that may arise with the selection of single sentences, the authors of another approach [24] argue that a summarizer should select full paragraphs to provide adequate context.

In a similar way, when we choose representative sentences in a connected graph, we also choose cohesive sentences with the representative sentences. In the legal case decision, these cohesive sentences support the main sentence with supported proofs,

....
[3] These appeals are brought by part-time workers to challenge the compatibility in relation to their employment of sections 2(4) and 2(5) of the Equal Pay Act 1970 (as amended by section 8(6) of and paragraph 6(1) of Part I of Schedule 1 to the Sex Discrimination Act 1975), section 2(5) being read with effect from 6 April 1978 with regulation 12 of the Occupational Pension Schemes (Equal Access to Membership) Regulations 1976 (SI 1976 No 142).
….
[9] "(1) The Equal Pay Act shall be so modified as to provide that where a court or an industrial tribunal finds that there has been a breach of a term in a contract of employment which has been included in the contract, or modified, by virtue of an equality clause and which relates to membership of a scheme, ...
……
[20] The first question posed asked (in part (a)) whether the requirement of section 2(4) that a claim could only be referred to an Industrial Tribunal if a woman had been employed in the employment within the six months preceding the date of reference meant that it was excessively difficult or impossible in practice for rights under article 119 to be exercised.
….
[37] Such a rule as that in section 2(5) of the 1970 Act was therefore incompatible with Community law as was a procedural rule like regulation 12 of the 1976 Regulations which prevented the entire record of service completed by those concerned before the two years preceding the date on which they commenced proceedings from being taken into account for the purpose of calculating the benefits which would be payable even after the date of the claim:p 997, para 42 and 43.
…
[40] Future pension benefits have therefore to be calculated by reference to full and part-time periods of service subsequent to 8 April 1976, the date of the Court's judgment in Defrenne v Sabena (Case 43/75)[1976] ICR 547 (when the court held that article 119 of the EC Treaty had direct effect: see Vroege v NCIV Instituut voor Volkshisvesting BV (Case C-57/93); Fisscher v Voorhuis Hengelo BV (Case C-128/93)[1995] ICR 635.
…
[130] I would accordingly allow the appeal to the extend:(a) of declaring that the respondents cannot rely on the two-year rule in section 2(5) of the 1970 Act to prevent the applicants from retroactively gaining membership of the pension scheme in the period of employment back to 8 April 1976 or to the date of commencement of employment....
[131] I would refer the question as to which of the Appellants can satisfy that condition back to the Employment Tribunal.
…
[135] I have had the opportunity of reading in draft the opinion prepared by my noble and learned friend Lord Clyde.
…
[140] I agree with it; and on the basis there set out I concur in the order proposed by my noble and learned friend Lord Slynn of Hadley.
…
[144] I agree with it, and for the reasons which he was given I would allow the appeal to the extent that he has indicated and make the same order as he has proposed.
…
[176] On the basis of the wider approach to the problem of comparison which my noble and learned friend Lord Slynn of Hadley has adopted I am in full agreement with him that the rules of procedure for a claim under section 2(4) of the 1970 Act are not less favourable than those which would apply to a claim for breach of contract in the circumstances of the present cases.
….
[177] I would accordingly agree with the conclusion which he has reached and with the form of order which he proposes.
…

Figure 1. Example legal document

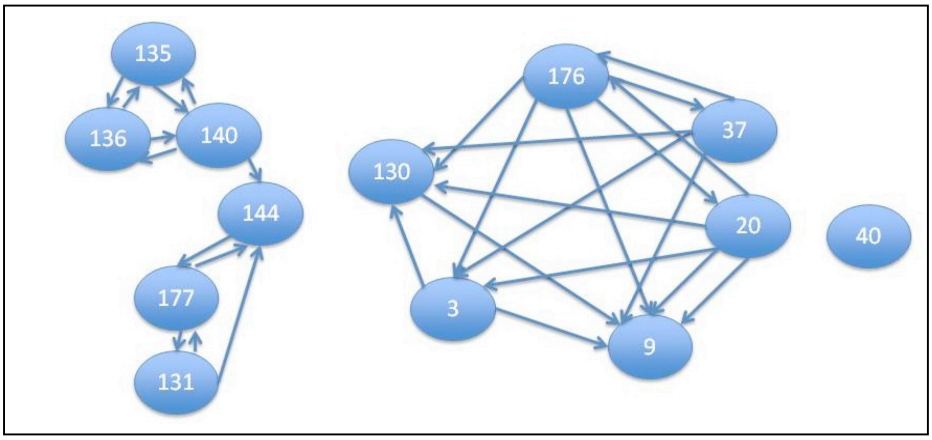

Figure 2. Graph representation of the example document

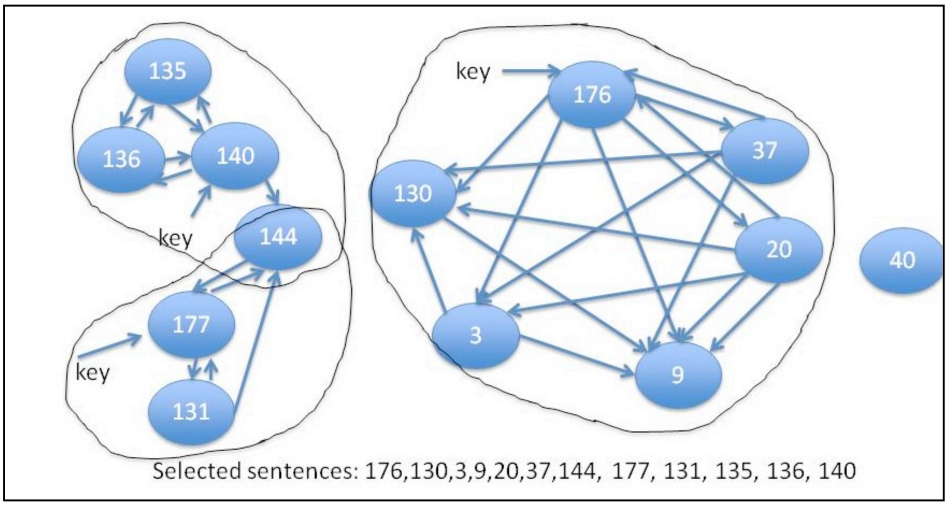

Figure 3. Selected sentences for summarization

facts, and rules. The algorithm is as following: For each connected graph, we choose every node '*A'* whose weight is over threshold as a representative node, and we also include all the cohesive nodes of node '*A'*. We define 'cohesive node' of node '*A'* as the node which embeds parts of the content of node *A*, but does not have exactly the same meaning with 'A'. The 'cohesive node' may include its own another meaning, which supports the content of node 'A'. We consider all the children nodes of a node 'A' are the cohesive nodes. A child node of node 'A' should have a directed edge starting from node 'A'.

In our experiments, the nodes that are chosen as the representative sentences are {177, 176, 140} in Figure 3. The selected cohesive sentences for summarization are shown in Figure 3, and the compression rate is automatically determined.

3 Experimental Results

3.1 Comparison of Experimental Results

We use the corpus of judgments of the House of Lords[1], which C. Grover et al.[13] gathered and annotated. These texts contain a header providing structured information, followed by a sequence of Law Lord's judgments consisting of free-running text. The structured part of the document contains information such as the *respondent*, *appellant,* and the *date of the hearing*. The decision is given in the opinions of the Law Lords, at least one of which is a substantial speech. This corpus consists of 188 judgments from the years 2001-2003 from the House of Lords website. We just extract raw sentences from the HTML documents, and do not use any kind of manually annotated linguistic information. The number of words in the corpus is 2,887,037 and the total number of sentences is 98,645. The average sentence length is approximately 29 words. A judgment contains an average of 525 sentences while an individual Lord speech contains an average of 105 sentences. They annotate sentences for "relevance" as measured by whether they match sentences in hand-written summaries. In this version of corpus, there are 47 judgments that have been annotated for relevance.

We compared our performance with that of B. Hachey [14], who applied a supervised machine learning algorithm which exploited a variety of deep linguistic and sentential features with annotated rhetorical structures. Table 1 shows that our algorithm outperformed previous result even though we did not use any deep linguistic features and rhetorical structures.

To compare the performance with the existing clustering algorithm, we tried X-Means clustering, and then applied our algorithm to each cluster. Table 2 shows that clustering method shows poorer performance than ours. Table 2 also shows our performance is better than existing TF*IDF-based and centroid-based sentence selection [12]. We also experimented with change to the edge-weighting scheme and node-weighting scheme. When the existing measures were used based on symmetric cosine similarity, the results showed worse performance than ours as shown in Table 2. We also compare the compression rate of our method with that of gold standards. Our compression rate varies according to each document. For one document we choose only 8 sentences, and the correct number of sentences are 34. In another document, we choose 66 sentences when the sentence number in gold standards are 36. But, except for these two extreme cases, the rest of the documents shows very close compression rate with that of gold standards. Our method chooses 1930 sentences in total, and the number of correct sentences are 1660 among all 12939 sentences. The average compression rate of our method is 14.9% and that of the gold standard is 12.8%.

---

[1] Until 2009, a subset of the British House of Lords, the upper house of the British Parliamentary system (cf. House of Commons) served as the highest judicial appeal court of the United Kingdom, which is now performed by Supreme Court of the United Kingdom.

Table 1. Our performance vs. other performance

	Precision (%)	Recall (%)	F-measure (%)
Hackey and Grover [14]	31.7	30.7	31.2
Our performance	31.3	36.4	33.7

Table 2. Performances using other previous algorithms

	Precision (%)	Recall(%)	F-measure(%)
Our performance	31.3	36.4	33.7
Clustering + graph-based sentence extraction	38.4	26.2	31.9
Clustering + tfidf-based sentence extraction	24.2	22.9	23.5
Clustering + centroid-based sentence extraction	21.9	20.0	20.9
Not using our edge weight (using cosine similarity)	19.3	61.5	29.0
Not using our vertex weight (using the sum of edge weights)	14.7	62.9	23.9

Graph-based summarization has many advantages. First of all, we can easily determine which content of a sentence embeds the content of another sentence, by constructing directed graphs. Based on the number of edges, we can see how many related sentences appear in a sentence, and we can also detect how many topics are covered in a document by counting the number of connected graphs. Since the set of connected graphs also function as clusters, we focus more on selecting sentences with cohesion in each connected graph, rather than on selecting sentences that cover various topics. As a result, our weighting measure for directed edges and nodes showed better performance than the previous algorithms.

3.2 Discussion

In this subsection, we analyze the errors of our system. Our method currently chooses cohesive sentences starting from the decision sentences based on the embedding probability. However, some sentences exist which are included in the gold standard summary, even though they do not share many words with the decision sentence, and it makes our recall lower. To improve recall, we need additional feature above lexical information such like rhetorical structure information.

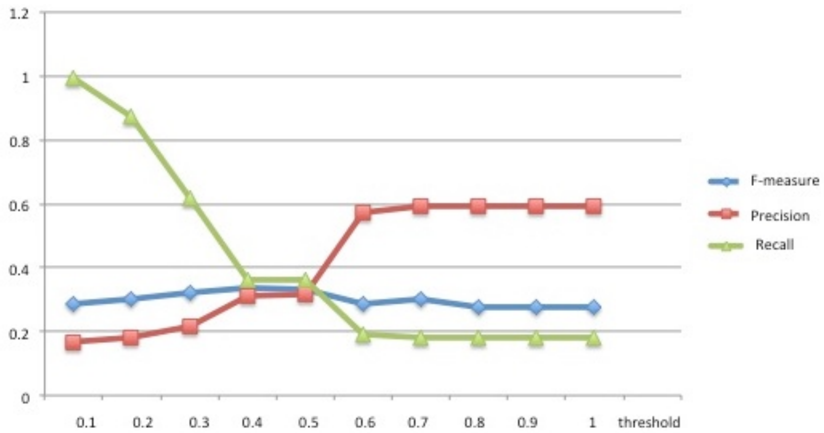

Figure 4. Performances according to the threshold

Among all the children nodes of a conclusion node, we choose cohesive nodes which support the decision based on the embedding probability. When this method chooses unnecessary sentences for summarization, it makes the precision lower. To improve the precision, we can consider more sophisticated features besides embedding probability. Currently, our system shows better performance than the previous system using rhetorical structure. We can expect improved performance when combining our method with the information of rhetorical structure in future work.

In order to choose the conclusion sentence, we consider two words 'agree' and 'dismiss'. However, the two words can occur in other sentences which are not conclusions, and these noises also make our precision lower. We can also consider location information to choose conclusion sentence in future work.

Let us explain the first method, clustering+graph in Table 2. Graph-based representation also provides a kind of clustering because it brings out the set of connected graphs. Since the first method performs clustering on the graph, the connected graphs are divided into smaller units. Therefore, supporting sentences for a decision can be distributed across different clusters, which results in low recall. Subsequently, when we choose cohesive supporting sentences from a decision sentence in a cluster, some supporting sentences are missing because they are included in another cluster. This results in higher precision because the size of constructed summary becomes smaller, but results in lower recall. Table 2 shows that the F-measure combining precision and recall does not outperform our proposed method.

The second method in Table 2 is work by selecting sentences in the order of high TF*IDF for each X-Means cluster. In a similar way, the third method chooses sentences close to the centroid of each cluster. These two methods have the following limitations: First, they do not consider coherence when choosing sentences. Second, the compression rate needs to be provided by a user in advance. Third, they do not have information which cluster is more important for summarization. They just determine the number of chosen sentences for each cluster based on the cluster size. A larger cluster does not mean it includes more key sentences for summarization. In other words, a user does not know which cluster includes decision sentences which are most important in summarization. Therefore, decision sentences may not be included in the summary in many cases, and that results in low recall.

Since cosine similarity of the fourth method (see Figure 2) measures the lexical similarity between two sentences, we can also choose the sentence which shows the same meaning of the other sentence. This violates the principle of summarization, and it shows lower performance than ours.

The last method in Table 2 uses the sum of edge weights in order to choose nodes for summarization. In legal judgment, the most important sentence for summarization is the legal decision, but the high score of a node does not mean that the node represents legal decision for summarization. Our method which detects the most important sentence based on the keyword "dismiss" and "agree" shows higher performance than that of the last method in Table 2.

The change of performance according to the threshold of edge value is shown in Figure 4. The higher the threshold value, the more precision and the less recall. Even though the F-measure based on the recall and precision is not changed much according to the threshold value, we need to determine a threshold value which shows reasonable value for both of the precision and recall.

We choose an intermediate value [0.4~0.5] between [0,1] for the threshold heuristically. In our experiments, 0.4 is used as threshold. We will verify if the same threshold is okay in larger experimental data in future work.

At this threshold, the compression rate was 14.9%, and it was closest to the compression rate of human-constructed summary of 12.8. Even though the overall compression rate is 14.9%, the compression rate of each document varies based on the characteristics of each document, and is automatically determined.

4 Related Work

Many text summarization systems have been proposed to date. Edmundson's early work [4] set the direction for later investigation which applies machine learning techniques for summarization. Machine learning approaches offer great freedom to summarization because the number of indicators of importance is practically endless [7, 15, 16, 28, 31]. Some common features include the position of the sentence in the document (first sentences of news are almost always informative), position in the paragraph (first and last sentences are often important), sentence length, similarity of the sentence with the document title or headings, weights of the words in a sentence determined by any topic representation approach, presence of named entities or cue phrases from a predetermined list. The proposed features are different in each paper, and new features are selected for each new domain.

Lexical chains [1, 9, 26] and some related approaches represent topics that are discussed throughout a text by exploiting relations between words. They capture semantic similarity between nouns to determine the importance of sentences. That approach heavily relies on WordNet , which is clearly a bottleneck for the approaches above, because success is constrained by the coverage of WordNet. Because of this, robust methods such as latent semantic analysis that do not use a specific static hand-crafted resource have much appeal.

The Latent Semantic Analysis (LSA) [3] method is a robust unsupervised technique for deriving an implicit representation of text semantics based on observed co-occurrence of words. The original proposal of Gong and Liu [11] was to select one sentence for each of the most important topics. They perform dimensionality reduction, retaining only as many topics as the number of sentences they want to include in the summary. This strategy suffers from the same drawback as the lexical chains approach because more than one sentence may be required to convey all information pertinent to that topic. More recent proposals suggest alternative procedures, which have led to improved performance of the summarizer in content selection. One improvement is to use the weight of each topic in order to determine the relative proportion of the summary that should cover the topic, thus allowing for a variable number of sentences per topic. In this method, it's also difficult to determine how to put weight of each topic, and they are usually highly heuristics.

A problem inherent in the supervised leaning paradigm is the necessity of labeled data on which classifiers can be trained. Asking annotators to select summary-worthy sentences is a reasonable solution [27], but it is time consuming and even more importantly, annotator agreement is low [21]. Another option for training a classifier is to employ a semi-supervised approach [28, 29].

In the legal domain, [6] annotated semantic roles and citations manually, and as a result, they produce a table style summary. They construct linguistic markers manually for each thematic segment and also manually construct citation indicators. When they choose best candidate units in document for summarization, they depend on heuristic functions: position of paragraphs and sentences. M. Yousfi-monod [30] constructed context-free grammar to obtain linguistic cues. They also obtained hints from HTML emphasis tags. They have five categories for classification: not in summary, introduction, context, reasoning and conclusion. They learned this classification using Naïve Bayes, based on three kinds of features (surface, emphasis, and content features). M. Saravanan [23] used a supervised learning algorithm based on Conditional Random Fields (CRF) using various linguistic features including rhetorical structures. These previous approaches to characterization of legal domains need manual annotation of diverse features, and this labor is expensive. Since the characteristics of sentences in each domain are different, these expensive features do not ensure they can be used in another domain. Rhetorical structure of a document is also different in each domain.

Up to now, many different techniques have proposed the selection of the most important part of the text with statistical methods which include Aggregation Similarity Method, Location Method, Frequency Method, TF-Based Query Method, linguistic methods which include Graph Theory, Lexical Chain, WordNet and Clustering [18].

5 Conclusion

We have presented an approach to the automatic summarization of legal texts. Our approach focuses on a new graph-based summarization system, which constructs a directed graph for each document. We propose our own edge weighting measure that focuses on the probability of embedding one sentence into another sentence.

Therefore, this measure is asymmetric, which leads to the creation of a directed and unconnected graph (a set of connected graphs). We also propose our own node weighting measure, which computes how likely words in a sentence appear in conclusion of judgments.

For cohesive summarization, when we choose nodes above threshold as representative sentences, we also choose the children nodes that have directed edges from the representative sentences. The directed edge from node 'A' to node 'B' means the meaning of 'A' is partly embedded in 'B'. The node 'B' usually includes more judgment meaning such as supporting rules, facts, or proofs.

Experimental results showed that our approach outperforms previous clustering method, edge/node weighting measures, as well as TF*IDF and centroid-based sentence selection methods.

Since the data size of the current experiment is small, we will increase the size of the gold standard in future work to see if the experimental result shows stable performance. In addition, we will apply our graph-based summarization into legal judgments of other languages. We also consider applying our method to other domains, and then determine those factors that affect the graph representation for different domains.

Acknowledgements

This research was supported by the Alberta Ingenuity Centre for Machine Learning (AICML) and the iCORE division of Alberta Innovates Technology Futures.

References

1. R. Barzilay and M. Elhadad. Text summarizations with lexical chains. In Inderjeet Mani and Mark Maybury, editors, Advances in Automatic Text Summarization, pages 111–121. MIT Press, 1999
2. E. Chieze, A. Farzindar, and G. Lapalme, “An Automatic System for Summarization and Information Extraction of Legal Information", Semantic Processing of Legal Texts, LNAI 6036, p. 216-234, 2010
3 S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman, Indexing by latent semantic analysis. Journal of the American Society for Information Science, pp. 391–407, 1990.
4. H. Edmundson, “New methods in automatic extracting”. Journal of the ACM, 16(2):264-285, 1969
5. G. Erkan and D.R. Radev, “LexRank: Graph-based centrality as salience in text summarization”, Journal of Artificial Intelligence Research, Vol. 22, pp. 457–479, 2004.
6. A. Farzindar and G. Lapalme, “Legal Texts Summarization by Exploration of the Thematic Structures and Argumentative Roles”, In Text Summarization Branches Out: Proceedings of the ACL-04 Workshop, pp. 27-34, 2004
7. M. Fuentes, E. Alfonseca, and H. RodrÅLıguez. Support vector machines for query-focused summarization trained and evaluated on pyramid data. Proc. of the Annual Meeting of the Association for Computational Linguistics, Companion Volume: Proceedings of the Demo and Poster Sessions, pp 57–60, 2007.

8. F. Galgani, P. Compton, and A. Hoffmann, “Citation Based Summarization of Legal Texts”, Proc. of PRICAI, pp.40-52, 2012
9. M. Galley and K. McKeown. Improving word sense disambiguation in lexical chaining. Proc. of the international joint conference on Artificial intelligence, pages 1486–1488, 2003.
10. R.A. Garcia-Hernandez and Y. Ledeneva, “Word Sequence Models for Single Text Summarization”, Proc. of Conference on Advances in Computer-Human Interaction, pp. 44-48, 2009
11. Y. Gong and X. Liu. Generic text summarization using relevance measure and latent semantic analysis. Proc. of the Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 19–25, 2001.
12. V. Gupta, "A Survey of Text Summarization Extractive Techniques", Journal of Emerging Technologies in Web Intelligence, 2(3):258-268, 2010
13. C. Grover, B. Hachey, I, Hughson, “The HOLJ Corpus: supporting summarization of legal texts", Proc. of the 5th International Workshop on Linguistically Interpreted Corpora, 2004
14. B. Hachey and C. Grover, “Extractive summarization of legal texts”, Artificial Intelligence and Law , 14:305-345, 2006
15. D. Hakkani-Tur and G. Tur, Statistical sentence extraction for information distillation. Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing, volume 4, pages IV–1 –IV–4, 2007
16. A. Louis, A. Joshi, and A. Nenkova. Discourse indicators for content selection in summarization. In Proceedings of the Annual Meeting of the Special Interest Group on Discourse and Dialogue, pages 147–156, 2010.
17. R. Mihalcea and P. Tarau, “TextRank: Bringing order in texts,” Proc. of the Conference on Empirical Methods in Natural Language Processing, pp.404-411, 2004
18. A. Nenkova, and K. McKeown, "A survery of text summarization techniques", Mining Text Data, pp.43-76, LLC 2012
19. M. Osborne. Using maximum entropy for sentence extraction. In Proc. of the ACL Workshop on Automatic Summarization, pp. 1–8, 2002
20. M. Rada Graph-based ranking algorithms for sentence extraction, applied to text summarization, ACL 2004, pp. 170-173, 2004
21. G. Rath, A. Resnick, and R. Savage. The formation of abstracts by the selection of sentences: Part 1: sentence selection by man and machines. American Documentation, 2(12):139–208, 1961
22. K. Sankar, and L. Sobha, “An Approach to Text Summarization”, Proc. of Third International Cross Lingual Information Access Workshop, pp. 53-60, 2009
23. M. Saravanan, B. Ravindran and S. Raman, "Improving Legal Document Summarization Using Graphical Models", Proc. of JURIX, pp. 51-60, 2006
24. G. Salton, A. Singhal, M. Mitra, and C. Buckley, “Automatic text structuring and summarization”, Information Processing and Management, 33(2):193-208, 1997
25. H. Shakeri, S. Gholamrezazadeh, M. A. Salehi, and F. Ghadamyari, “A New Graph-Based Algorithm for Persian Text Summarization”, Lecture Notes in Electrical Engineering 114, pp.21-30, 2012
26. H. Silber and K. McCoy. Efficiently computed lexical chains as an intermediate representation for automatic text summarization. Computational Linguistics, 28(4):487–496, 2002.
27. J. Ulrich, G. Murray, and G. Carenini. A publicly available annotated corpus for supervised email summarization. Proc. of the AAAI EMAIL Workshop, pp. 77–87, 2008.
28. K. Wong, M. Wu, and W. Li. Extractive summarization using supervised and semi-supervised learning. Proc. of the 22nd International Conference on Computational Linguistics (Coling 2008), pp. 985–992, 2008.
29. S. Xie, H. Lin, and Y. Liu. Semi-supervised extractive speech summarization via co-training algorithm. In INTERSPEECH, the 11th Annual Conference of the International

Speech Communication Association, pp. 2522–2525, 2010.
30. M. Youshfi-Monod, A. Farzindar, and G. Lapalme, "Supervised Machine Learning for Summarizing Legal Documents", Proc. of Canadian AI, LNAI 6085, pp. 51-62, 2010
31. L. Zhou and E. Hovy. A web-trained extraction summarization system. Proc. of the Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology, pp. 205–211, 2003
32. G. Salton, G, and C. Buckley. Term Weighting Approaches in Automatic Text Retrieval. Information Processing and Management 24(5), pages 513-523, 1988