Legal Knowledge and Information Systems
E. Schweighofer (Ed.)

doi:10.3233/FAIA210322

An Analytical Study of Algorithmic and Expert Summaries of Legal Cases

Aniket Deroy [a,1], Paheli Bhattacharya [a], Kripabandhu Ghosh [b], Saptarshi Ghosh [a]
[a] *Indian Institute of Technology, Kharagpur India*
[b] *Indian Institute of Science Education and Research, Kolkata*

Abstract. Automatic summarization of legal case documents is an important and challenging problem, where algorithms attempt to generate summaries that match well with expert-generated summaries. This work takes the first step in analyzing expert-generated summaries and algorithmic summaries of legal case documents. We try to uncover how law experts write summaries for a legal document, how various generic as well as domain-specific extractive algorithms generate summaries, and how the expert summaries vary from the algorithmic summaries. We also analyze which important sentences of a legal case document are missed by most algorithms while generating summaries, in terms of the rhetorical roles of the sentences and the positions of the sentences in the legal document.

Keywords. Case document summarization; extractive summarization; rhetorical roles; lead bias

1. Introduction

Summarization of legal case documents [1, 2] is a challenging problem, which has become important in recent years due to a massive increase in the amount of legal cases available online, and the huge length of most case documents. Several legal domain-specific algorithms as well as generic (domain-independent) algorithms have been used to generate summaries for legal case documents. Most prior works have compared the algorithmic summaries with expert-generated summaries for legal documents in terms of ROUGE scores [3]. However, there has not been much investigation on which parts (e.g., sentences) of a case document are actually selected by experts and by various algorithms for generating the summary of the document.

In this work, we take the first steps in this direction. Specifically, we seek answers to a set of Research Questions (**RQs**) that would help improve our understanding of how summarization algorithms perform in relation to summaries written by the legal experts, and thus give insights that may be valuable for the research fraternity. The Research Questions (**RQs**) that we address in this work are as follows:- **RQ1**: Which parts of a document are selected by summarization algorithms and domain experts while generating summaries? **RQ2**: Which *rhetorical roles* [4] (e.g., Facts, Issues, Ruling) are mostly selected by summarization algorithms and experts while generating summaries? **RQ3**: Are ROUGE scores (computed w.r.t. expert summaries) consistent with the rhetorical roles distributions selected by summarization algorithms? **RQ4**: Out of the sentences that are considered to be important by domain experts, which sentences are easier (or difficult) to identify for summarization algorithms?

[1]Corresponding Author: Aniket Deroy. Email: roydanik18@kgpian.iitkgp.ac.in

To answer these questions, we conduct a detailed analysis over 50 case documents from the Indian Supreme Court and their summaries written by two law experts (dataset obtained from our prior work [2]). We analyse the summaries generated by as many as 15 extractive summarization algorithms, including traditional unsupervised algorithms and supervised neural algorithms. We attempt to analyse how an algorithm or an expert chooses sentences for generating a summary in terms of (i) position of the sentences in the original legal document, and (ii) the rhetorical role [4] of the sentences. It can be noted that prior works have shown that rhetorical roles are necessary to detect important sentences to create good quality summaries of legal documents [1].

Based on the Research Questions stated above, we found the following insights.

1. Some summarization algorithms like BERTSUM [5], Luhn [6] and Letsum [7] tend to select sentences from the initial portions of the input document, which is termed as *lead bias* [8]. Supervised algorithms like BERTSUM which are *pretrained on news article corpora* especially suffer from the lead bias problem. On the other hand, supervised models like SummaRunner [9] and Chinese Gist [10] which can be trained from scratch do not suffer from lead bias problem (Sec. 4.2)
2. Some domain-specific algorithms like KMM [11], MMR [12] and DELSUMM [2] tend to focus on the *Ruling by present court* rhetorical role which is similar to what is done by the domain experts. The *Ruling by Present court* rhetorical role is significant because this rhetorical role includes the final judgement of a legal case (Sec. 4.3)
3. Most algorithms tend to include sentences from the *facts* rhetorical role (possibly due to *lead bias* as *facts* usually appear towards the beginning of a case). On the other hand, a large proportion of sentences belonging to the *Ratio of the decision* rhetorical role is missed by most summarization algorithms because these sentences occur mostly towards the latter portions of the document (Sec. 5.1)
4. The rhetorical role-wise ROUGE scores (computed w.r.t. expert summaries) is consistent with the rhetorical roles distribution selected by most of the summarization algorithms (though there are a few exceptions) (Sec. 4.4)

2. Related work

There have been several prior works on applying summarization algorithms to legal documents [1, 2]. In this section, we discuss about different categories of summarization algorithms that have been applied to legal case documents.

Unsupervised domain-independent: Lexrank [13] is a graph-based summarization technique that uses the idea of eigenvector centrality. Luhn summarizer [6] is a simple method for detecting the most important set of sentences in a document using the concept of TF-IDF vectors. LSA summarizer [14] uses Singular Value Decomposition to project the singular matrix from a higher dimensional plane to a lower dimensional plane to select the most important sentences in the document. Reduction summarizer [15] attempts to condense a long document into the most important parts by creating a rich semantic graph. DSDR [16] is an algorithm which works on the principle of data reconstruction, thereby minimizing the reconstruction error.

Supervised domain-independent: SummaRunner [9] is an algorithm which uses hierarchical Recurrent Neural Networks (RNNs) to learn sentence representations from the input document. To select the sentences for the summary, this algorithm uses relative and absolute position importance, salience, content and novelty. SummaRunner has 3

variations which are SummaRunner/RNN_RNN (which consists of two layers of RNNs), SummaRunner/CNN_RNN (which consists of one layer of Convolutional Neural Network and one layer of RNN), and SummaRunner/Attn_RNN that consists of an attention mechanism with a RNN layer. BertSum [5] is an algorithm which has been initially trained on large amount of news article data. The pre-trained model can be fine-tuned with document-summary pairs from a target domain (e.g., legal document-summary pairs in our case).

Unsupervised legal domain-specific: Letsum [7] divides the entire legal document into four parts namely Introduction, Context, Judicial analysis and Conclusion, and then takes portions of these four parts to form the summary. Case summarizer [17] uses parameters like TF-IDF values, number of dates in a sentence, number of named entities and whether a sentence is in the starting section of the document to select candidate sentences for the summary. MMR algorithm [12] is designed for legal cases related to post-traumatic stress disorder from the US Board of veterans appeal court. This method uses a pipeline consisting of a CNN classifier to select sentences for the summary. Delsumm [2] chooses sentences from the input legal document using a set of rules based on Integer Linear programming. KMM [11] stands for K-mixture model and this K-mixture model is used for selecting sentences to create the summary from the original document.

Supervised legal domain-specific: Chinese Gist [10] is a legal domain-specific supervised algorithm that uses several deep learning and machine learning methods (such as LSTMs) to create various classifiers that are together used with necessary features to generate summaries of legal documents.

Evidently, many prior works have applied summarization algorithms on legal case documents. However, there has not been any prior attempt toward analysis of the summaries generated by summarization algorithms as well as of gold standard summaries generated by experts. This work aims to fill this gap.

3. Dataset

We reuse the dataset from our prior works [2, 4] which consists of 50 case documents from the Indian Supreme Court. To improve the generalizability of the study, the 50 case documents are drawn from 5 different domains – (i) Criminal - 16 documents, (ii) Land and property – 10 documents, (iii) Constitutional – 9 documents (iv) Labour and Industrial – 8 documents, and (v) Intellectual Property Rights – 7 documents. Two senior law students from the Rajiv Gandhi School of Intellectual Property Law, India (one of the most reputed law schools in India) annotated the legal documents with rhetorical labels [4] for each sentence, as well as summarized the legal documents.

Annotation with rhetorical roles: Every sentence in every document has been annotated with one of the following 8 rhetorical labels (the annotation process is detailed in [4]):- **(1) Facts** (abbreviated as **FAC**) are the chronology of events which lead to the filing of the legal case (it includes events like doing FIR at the police station, filing of the case at the court, etc). **(2) Issues** (abbreviated as **ISS**) refer to the legal questions on which the legal case is based. **(3) Ruling by Lower court** (**RLC**) is the judgement given by a lower court on a case which is being contended in a higher court. Since here the legal cases that we are considering are Supreme Court cases so the cases have already being contended in the lower court(s) and the lower court's have passed a decision on that

	FAC	ARG	Ratio	PRE	RLC	RPC	ISS	STA
Original Document	**0.261**	0.082	0.364	0.141	0.033	0.033	0.013	0.069
Expert 1	**0.269**	0.091	0.380	0.074	0.001	0.070	0.032	0.079
Expert 2	**0.289**	0.078	0.371	0.088	0.002	0.067	0.026	0.075

Table 1. Distribution of the rhetorical roles in the original documents, and the summaries written by expert 1 and expert 2, averaged across the 50 documents. Each value is the fraction of sentences of a particular rhetorical role, out of the total number of sentences in the document / expert summaries, averaged over the 50 documents. Blue-underlined represents the rhetorical role with highest fraction of sentences. Violet-bold represents the rhetorical role with second highest fraction of sentences.

case. **(4) Arguments** (**ARG**) are presented by the lawyers of the parties involved in the case. **(5) Precedents** (**PRE**) are the past legal cases which are cited in the present case. **(6) Statutes** (**STA**) are the laws that are referred to, including orders, acts, notifications, articles, sections, rules, etc. **(7) Ratio of the decision** (abbreviated as **Ratio**) refers to the legal reasoning due to which the specific judgement is given. **(8) Ruling by present court** (abbreviated as **RPC**) is the final judgement given by the judge of the present court (the Supreme court of India, in our case).

Summaries of the documents: The two domain experts wrote summaries for each of the 50 documents. The summaries created by the experts are mostly extractive in nature; however, some sentences were slightly modified by the experts to improve the readability / grammatical flow of the summary The length of the summaries written by the experts was around 30% of the original legal document length. Specifically, the collection has on an average 5387.36 words per document, and 1648.76 and 1710.66 words on average in the summaries written by the two experts. Similarly, there are on average 172.86 sentences per document, and 54.06 and 56.72 sentences on average in the summaries written by the two experts. All these values are averaged across the 50 documents. Details of the summarization process can be found in [2].

Table 1 shows the distribution of rhetorical labels across the documents and the expert summaries. Each value is the fraction of sentences of a particular rhetorical role, out of the total number of sentences in the document / expert summaries, averaged over all the 50 documents / their expert summaries. We see that, for both the original documents as well as the expert-written summaries, *Ratio of the decision* is the largest class (these sentences occur most frequently across all documents/summaries) followed by the *Facts* and *Precedent*.

Training data for supervised algorithms: The supervised summarization algorithms (SummaRunner, GIST, BERTSUM) are trained/fine-tuned over a training set consisting of 7,100 Indian Supreme Court case documents and their headnotes (short abstractive summaries); further details can be found in [2]. We ensured that there was no overlap between this training set and the evaluation set of 50 documents.

4. Analyzing the Algorithmic and Expert Summaries

We applied all the summarization algorithms described in Section 2 on the 50 case documents. *For a particular document, all the algorithms were made to generate summaries of the same maximum word count as the average number of words in the two expert-summaries for the same doc.* This section compares the summaries generated by the summarization algorithms and the summaries written by the legal experts (for the same document).

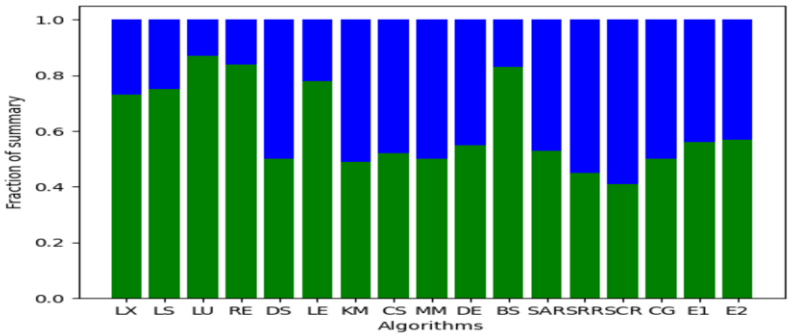

Figure 1. Fraction of the algorithmic summaries and expert summaries taken from the first half and second half of the original legal document, averaged for 50 legal documents. Green colour represents the fraction of the algorithmic / expert summaries taken from the first half of the original document, while blue colour represents the fraction of the summaries taken from the second half of the original document. The symbols on the X-axis are as follows. LX: Lexrank, LS: LSA summarizer, LU: Luhn summarizer, RE: Reduction summarizer, DS: DSDR, LE: Letsum, KM: KMM, CS: CaseSummarizer, MM: MMR summarizer, DE: Delsumm, SAR: SummaRunner/Attn_RNN, SRR: SummaRunner/RNN_RNN, SCR: SummaRunner/CNN_RNN, BS: BERTSUM, CG: Chinese Gist, E1: Expert 1 and E2: Expert 2.

4.1. Finding the closest matching sentence in the document for every sentence in the expert summary

While writing the summaries, the two legal experts mostly copied sentences directly from the document (extractive summarization), but sometimes they combined multiple sentences from the document and/or edited the text of some sentences to improve the fluency and grammatical structure of the summary. For various analyses reported later, we intend to find, for every sentence in the expert summaries, the *closest matching sentence in the document*. To this end, we proceed as follows. We take a sentence s_e from an expert summary and compare the sentence with every sentence in the corresponding original document. If we get an exact match between s_e and some sentence s_d in the document, then s_d is taken as the closest matching sentence for s_e. If we do not get an exact match for s_e, we perform an approximate matching – we calculate ROUGE-2 F1-score (that considers bigram overlap) of s_e with every sentence in the corresponding document. The closest matching sentence in the document for s_e is taken to be that sentence s_d in the document that has the highest ROUGE-2 F1-score match with s_e.

*4.2. **RQ1** : Which parts of a document are selected by summarization algorithms?*

We start by checking the location of the sentences (in a document) that are selected by various algorithms and the experts, for inclusion in the summary. To this end, we consider a document to be partitioned into *two equal halves*, and check what fraction of sentences selected by an algorithm / an expert lies in which half of the corresponding document. Figure 1 shows the fraction of sentences in the algorithmic summaries and expert summaries that are taken from the first half and second half of the original documents, averaged over all the 50 legal documents.

From Figure 1 we can observe the two experts write well-balanced summaries including approximately 55% of sentences from the first halves of the documents and around 45% of the sentences from the second halves of the documents.

In contrast, some summarization methods like BERTSUM, Luhn summarizer, LetSum, Lexrank, LSA summarizer and Reduction summarizer choose most sentences from the first half of the original legal document. This property, where a summarization algorithm tends to choose text mostly from the initial parts of the input document, is known as lead bias [8]. Algorithms such as BERTSUM that are *pre-trained on news article summarization corpus* (where the first few sentences of a news article is known to usually be a good summary of the article), is not able to come out of lead bias due to initial training on news articles, even after they are finetuned on legal documents. In constrast, SummaRunner is trained fully (from scratch) on legal documents and their summaries, and is hence able to avoid lead bias. Letsum is a domain-specific algorithm which divides the document into four parts namely Introduction, Context, Judicial analysis and Conclusion and picks up 10%, 25%, 60% and 5% from each of these individual parts of the document to form the summary. So a large fraction of the summary sentences are picked up from the initial portions of the documents because the Introduction and Context primarily occurs in the initial portions of the document.

It is observed that most unsupervised domain-independent algorithms like Lexrank, LSA, Luhn, Reduction summarizer display significant lead bias. On the other hand, most of the domain-specific algorithms like KMM, Case Summarizer, MMR, DELSUMM and Chinese Gist tend to pick up sentences uniformly from both halves of the document.

4.3. ***RQ2*** *: Which rhetorical roles are selected by summarization algorithms?*

For every sentence from an algorithmic summary, we find the closest matching sentence in the original document and also the rhetorical role of the sentence in the original document. In this way we detect the rhetorical roles that are being selected by the summarization algorithms. Table 2 shows the fraction of each rhetorical role captured by every algorithm and by the two experts, out of the total number of sentences of a rhetorical role present in the original document, averaged over all 50 documents.

From Table 2 we can observe that the experts focused most on the *Ruling by present court (RPC)* and *Issues (ISS)* though these classes are present in small proportions in the original documents (see Table 1). DELSUMM and Chinese Gist are domain-specific algorithms which also focus on *Ruling by present court*. DELSUMM gives highest weight to *Ruling by present court* followed by *Issues*. On the other hand, some domain-specific algorithms like Case Summarizer and Letsum focused most on *Facts* and less on *Ruling by present court*. Supervised algorithms like BERTSUM focused most on the initial portions of the document and picked up *Facts* which are mostly present in the initial portions of the document. LSA summarizer has focused most on *Facts* and *Arguments*. Chinese Gist has also focused well on *Facts*.

Table 2. Fraction of sentences of each rhetorical role captured by every algorithm and by the two experts, out of the total number of sentences of that rhetorical role present in the original text, averaged out of 50 documents. Blue-underlined represents the rhetorical role with highest value. Violet-bold colour represents the rhetorical role with the second highest value.

Algorithm	FAC	ARG	Ratio	PRE	RLC	RPC	ISS	STA
Unsupervised, Domain Independent								
Lexrank	**0.310**	0.319	0.162	0.105	0.292	0.027	0.307	0.183
LSA	0.333	**0.291**	0.145	0.133	0.256	0.015	0.213	0.147
Luhn	0.354	0.264	0.100	0.096	**0.284**	0.015	0.230	0.201
Reduction	0.285	**0.284**	0.108	0.085	0.274	0.013	0.265	0.190
DSDR	**0.333**	0.264	0.330	0.270	0.221	0.456	0.195	0.285
Unsupervised, Domain specific								
Letsum	0.568	0.230	0.190	0.201	0.220	0.029	**0.381**	0.280
KMM	0.245	0.317	0.260	0.235	0.243	0.274	**0.283**	0.250
Case Summarizer	0.298	0.268	0.293	0.124	0.181	0.115	**0.298**	0.194
MMR algorithm	**0.351**	0.317	0.343	0.271	0.266	0.427	0.299	0.243
Delsumm	0.422	0.543	0.239	0.300	0.0	**0.688**	0.739	0.319
Supervised, Domain Independent								
SummaRunner/Attn_RNN	0.389	0.326	0.285	0.300	**0.329**	0.182	0.296	0.141
SummaRunner/RNN_RNN	0.283	0.240	0.355	**0.345**	0.233	0.274	0.274	0.123
SummaRunner/CNN_RNN	0.305	0.257	0.335	**0.335**	0.278	0.594	0.331	0.156
BERTSUM	0.665	0.335	0.149	0.118	0.220	0.040	**0.356**	0.212
Supervised, Domain Specific								
Chinese Gist	**0.461**	0.274	0.432	0.348	0.280	0.608	0.365	0.255
Expert								
Expert 1	0.388	0.465	0.377	0.197	0.014	0.734	**0.665**	0.390
Expert 2	0.432	0.406	0.380	0.224	0.015	0.764	**0.583**	0.390

Table 3. Rhetorical role-wise and entire document-wise performance of all the summarization methods in terms of ROUGE-L F1-scores, averaged over the 50 documents. Values which are < 0.3 are represented in red underlined. Blue bold represents the best value for each rhetorical role.

Algorithm	Entire document	Final judgement	Issue	Facts	Statute	Precedent +Ratio	Argument
Unsupervised, Domain Independent							
Lexrank	0.5392	0.0619	0.3469	0.4550	0.2661	0.3658	0.4284
LSA	0.5483	0.0275	0.2529	0.5217	0.2268	0.3527	0.3705
Luhn	0.5521	0.0358	0.2754	0.5408	0.2662	0.2927	0.3781
Reduction	0.542	0.0352	0.3153	0.5064	0.2579	0.3059	**0.4390**
DSDR	0.5725	0.4987	0.1982	0.4501	0.3174	0.4631	0.3490
Unsupervised, Domain Specific							
LetSum	0.5846	0.0423	0.3926	0.6246	0.3469	0.3853	0.2830
KMM	0.5385	0.3254	0.2979	0.4124	0.3415	0.4450	0.416
Case Summarizer	0.5349	0.2474	0.3537	0.4500	0.2255	0.4461	0.4184
MMR	0.568	0.4378	0.3548	0.4442	0.2763	0.4647	0.3705
DELSUMM	0.6017	**0.7929**	**0.6635**	0.5539	**0.4030**	0.4305	0.4370
Supervised, Domain Independent							
SummaRunner/RNN_RNN	0.5821	0.4451	0.2990	0.5231	0.1636	**0.5215**	0.3090
SummaRunner/CNN_RNN	0.5757	0.5893	0.3586	0.5069	0.1998	0.5026	0.2765
SummaRunner/Attn_RNN	0.5877	0.3633	0.3176	0.6072	0.1869	0.4933	0.4191
BERTSUM	0.5529	0.0662	0.3544	**0.6376**	0.2535	0.3121	0.3262
Supervised, Domain Specific							
Chinese Gist	0.5501	0.5844	0.3856	0.4621	0.2759	0.4537	0.2132

4.4. ***RQ3**: Are ROUGE scores (computed w.r.t. expert summaries) consistent with the rhetorical role distributions selected by summarization algorithms?*

ROUGE scores are widely considered as a standard way for measuring the quality of an algorithmic summary with respect to the gold standard (expert) summary. Here we examine whether the rhetorical role-wise ROUGE scores calculated for a particular algorithm tally with the algorithmic distribution of rhetorical roles selected by that algorithm. Note that, rhetorical role-wise ROUGE scores are more suitable for this analysis, than ROUGE scores for the entire document.

Table 3 shows the rhetorical role-wise ROUGE-L F1 scores of the algorithms. For most algorithms, the results given by the rhetorical role-wise ROUGE scores (computed w.r.t. expert summaries) are similar to the results given by the algorithmic distribution of rhetorical roles (that were discussed as part of RQ2). For instance, methods like Lexrank and CaseSummarizer get lower ROUGE-L F1-scores; this agrees with the observations in Table 2 where *Ruling by present court* and *Issues* have been selected extensively by the experts but selected in much less proportions by these algorithms. On the other hand, algorithms such as LetSum and DELSUMM show higher ROUGE-L F1-scores because the rhetorical distributions chosen by these algorithms are closer to the experts' rhetorical distributions.

5. Which important parts do most summarization algorithms miss?

We now attempt to characterize which important parts (sentences) of a legal case document are missed (i.e., not included in the summary) by most summarization algorithms.

5.1. ***RQ4:** Out of the sentences that are considered to be important by domain experts, which sentences are easier / difficult to identify for summarization algorithms?*

Using the method described in Section 4.1, we found the closest matching sentence in the document for every sentence in the expert summaries. Now we focus on those sentences from the original document, that were selected by the experts for inclusion in the gold standard summaries. For each such sentence, we check how many algorithms have included that particular sentence in their summary. Table 4 states the number and percentage of sentences in the expert summaries that are selected by less than 3 algorithms, sentences selected by 3-9 algorithms, and sentences selected by 10 or more algorithms. To better understand which parts of the documents are being selected (or missed) by the summarization algorithms, we focus on the following two sets of sentences:

Frequently selected sentences: The set of sentences which appear in at least one expert summary and are chosen by 10 or more algorithms for inclusion in the summaries. There are 155 such sentences in total across all the 50 documents.

Frequently missed sentences: The set of sentences which appear in at least one expert summary but are chosen by less than 3 algorithms. There are 529 such sentences in total across all the 50 documents.

Note that, both these sets contain important sentences that are chosen by the Law experts while writing their summaries. We analyze these two sets of sentences with the objective of gaining a better understanding of the frequently missed sentences which are important sentences being missed by most summarization algorithms.

Characterizing the location of frequently selected/missed sentences: Out of the set of *frequently missed sentences*, 35% come from the first halves of the documents, while 65% sentences come from the second halves of the documents (numbers averaged over

Table 4. Number and percentage of sentences in the expert summaries that are selected by less than 3 algorithms (frequently missed sentences), 3-9 algorithms and 10 or more algorithms (frequently selected sentences). Blue-underlined cell represents the *frequently missed sentences* for a particular expert. Violet-bold cell represents the *frequently selected sentences* for a particular expert.

Number of algorithms →	less than 3 algorithms (Frequently missed sentences)	3-9 algorithms	10 or more algorithms (Frequently selected sentences)
Expert 1	456 (17.7%)	1968 (76.6%)	**142 (5.5%)**
Expert 2	478 (18.4%)	2063 (76.0%)	**140 (5.2%)**
Union of both experts	529 (17.8%)	2280 (76.9%)	**155 (5.2%)**

Table 5. Comparison of the *frequently missed sentences* and the *frequently selected sentences* in terms of their rhetoric distribution. Blue-underlined colour represents the rhetorical role which is present in the highest proportion in a row.

Rhetorical roles →	FAC	ARG	Ratio	PRE	RLC	RPC	ISS	STA
Frequently missed sentences	0.061	0.067	0.575	0.096	0.0	0.080	0.013	0.105
Frequently selected sentences	0.619	0.140	0.112	0.014	0.0	0.0	0.077	0.035

Table 6. Examples of frequently missed sentences that are selected by Law experts in their summaries, but are not selected by most of the summarization algorithms.

Sentence	Rhetorical role
the appeals are disposed of accordingly without any order as to costs	RPC
order was a legislative activity and therefore not subject to any principle of natural justice	ARG
no vested right as to tax holding is acquired by a person who is granted concession	Ratio
what the order does contemplate however is such enquiry by the government as it thinks fit	Ratio

all 50 documents). On the other hand, as many as 93.6% of the *frequently selected sentences* come from the first halves of the documents, and only 6.4% come from the second halves. These numbers re-confirm the lead bias of several algorithms, as was discussed in Section 4.2 (**RQ1**) – most of the *frequently missed sentences* come from the second half of the documents.

Characterizing the rhetorical labels of frequently selected/missed sentences: Table 5 shows the rhetorical role distribution of the frequently missed and frequently selected sentences. We see that for the *frequently missed sentences*, most sentences belong to the *Ratio of the decision* rhetorical role. For the *frequently selected sentences*, most number of sentences belong to the *Facts* rhetorical role. This observation can also be ascribed to the prevalence of lead bias of these algorithms, as discussed in Section 4.2 (**RQ1**), since *Facts* usually appear at the beginning of a case document and *Ratio of the decision* usually occurs at the latter portions of a document. Table 6 gives some examples of the *frequently missed sentences* and their rhetorical labels.

Characterizing the length and legal keywords content of frequently selected/missed sentences: We found that frequently missed sentences have a similar distribution of length (number of words) as frequently selected sentences. We checked the number of legal keywords contained in the two types of sentences, using terms from a legal dictionary provided by [18]. The frequently selected sentences contain 3.30 legal terms on average, while frequently missed sentences contain 2.89 legal terms on average. The fact that frequently selected sentences contain more legal terms (than frequently missed sentences) may be a potential reason why most summarization algorithms choose them.

6. Conclusion and Future Work

In this work, we compare the algorithmic and expert summaries of legal case documents to unearth the nature and position of the sentences chosen to create algorithmic summaries and expert summaries. This work gives us several insights that can help in improving the existing summarization algorithms that are capable of creating summaries aligning more with the notion of legal experts – potential end-users of such algorithms.

Our current work considers rhetorical role-wise ROUGE scores to analyse the quality of legal document summaries. In future, we can apply metrics other than ROUGE scores to evaluate the quality of legal summaries, since there are limitations of quantitative metrics like ROUGE scores. Also, we plan to generalize our observations through similar experiments on legal documents of other jurisdictions and countries.

Acknowledgements: The authors thank the Law domain experts from the Rajiv Gandhi School of Intellectual Property Law, India who annotated the legal documents and wrote the summaries. The research is partially supported by the TCG Centres for Research and Education in Science and Technology (CREST) through a project titled "Smart Legal Consultant: AI-based Legal Analytics".

References

[1] P. Bhattacharya, K. Hiware, S. Rajgaria, N. Pochhi, K. Ghosh, and S. Ghosh, "A comparative study of summarization algorithms applied to legal case judgments," in *ECIR*, 2019.

[2] P. Bhattacharya, S. Poddar, K. Rudra, K. Ghosh, and S. Ghosh, "Incorporating domain knowledge for extractive summarization of legal case documents," in *ICAIL*, 2021.

[3] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Text Summarization Branches Out*, (Barcelona, Spain), pp. 74–81, Association for Computational Linguistics, July 2004.

[4] P. Bhattacharya, S. Paul, K. Ghosh, S. Ghosh, and A. Wyner, "Identification of rhetorical roles of sentences in Indian legal judgments," in *JURIX*, 2019.

[5] Y. Liu, "Fine-tune bert for extractive summarization," *ArXiv*, vol. abs/1903.10318, 2019.

[6] A. Nenkova, S. Maskey, and Y. Liu, "Automatic summarization," in *ACL*, 2011.

[7] A. Farzindar and G. Lapalme, "Letsum, an automatic legal text summarizing system," in *JURIX*, 2004.

[8] M. Grenander, Y. Dong, J. C. K. Cheung, and A. Louis, "Countering the effects of lead bias in news summarization via multi-stage training and auxiliary losses," in *EMNLP*, 2019.

[9] R. Nallapati, F. Zhai, and B. Zhou, "Summarunner: A recurrent neural network based sequence model for extractive summarization of documents," in *AAAI*, 2017.

[10] C.-L. Liu and K.-C. Chen, "Extracting the Gist of Chinese Judgments of the Supreme Court," in *ICAIL*, 2019.

[11] J. Vermunt, "K-means may perform as well as mixture model clustering but may also be much worse: Comment on steinley and brusco (2011)," *Psychological methods*, vol. 16, 2011.

[12] L. Zhong, Z. Zhong, Z. Zhao, S. Wang, K. D. Ashley, and M. Grabmair, "Automatic summarization of legal decisions using iterative masking of predictive sentences," in *ICAIL*, 2019.

[13] G. Erkan and D. R. Radev, "Lexrank: Graph-based lexical centrality as salience in text summarization," *J. Artif. Intell. Res.*, vol. 22, pp. 457–479, 2004.

[14] J.-Y. Yeh, H.-R. Ke, W.-P. Yang, and I. Meng, "Text summarization using a trainable summarizer and latent semantic analysis," *Information Processing and Management*, vol. 41, pp. 75–95, 2005.

[15] I. Moawad and M. Aref, "Semantic graph reduction approach for abstractive text summarization," in *International Conference on Computer Engineering and Systems*, pp. 132–138, 11 2012.

[16] Z. He, C. Chen, J. Bu, C. Wang, L. Zhang, D. Cai, and X. He, "Document summarization based on data reconstruction," in *AAAI*, 2012.

[17] S. Polsley, P. Jhunjhunwala, and R. Huang, "Casesummarizer: A system for automated summarization of legal texts," in *COLING*, 2016.

[18] A. Mandal, K. Ghosh, A. Pal, and S. Ghosh, "Automatic catchphrase identification from legal court case documents," in *ACM CIKM*, 2017.