Legal Knowledge and Information Systems
E. Francesconi et al. (Eds.)

doi:10.3233/FAIA220455

Conditional Abstractive Summarization of Court Decisions for Laymen and Insights from Human Evaluation

Olivier SALAÜN [a,1], Aurore TROUSSEL [b], Sylvain LONGHAIS [b], Hannes WESTERMANN [b], Philippe LANGLAIS [a] and Karim BENYEKHLEF [b]
[a] *RALI, DIRO, Université de Montréal*
[b] *Cyberjustice Laboratory, Faculty of Law, Université de Montréal*

Abstract. Legal text summarization is generally formalized as an extractive text summarization task applied to court decisions from which the most relevant sentences are identified and returned as a gist meant to be read by legal experts. However, such summaries are not suitable for laymen seeking intelligible legal information. In the scope of the JusticeBot, a question-answering system in French that provides information about housing law, we intend to generate summaries of court decisions that are, on the one hand, conditioned by a question-answer-decision triplet, and on the other hand, intelligible for ordinary citizens not familiar with legal documents. So far, our best model, a further pre-trained BARThez, achieves an average ROUGE-1 score of 37.7 and a deepened manual evaluation of summaries reveals that there is still room for improvement.

Keywords. text summarization, court decisions, French texts, housing law, access to legal information

1. Introduction

In the province of Quebec in Canada, the Tribunal administratif du logement (TAL, Housing Law Tribunal) is a court with an exclusive jurisdiction within the framework of provincial housing law for all legal disputes involving a lease contract among landlords and tenants. Such litigations are generally motivated by late payment of the rent or substandard housing. Since the TAL has to deal with a massive number of cases every year (i.e. over 51.7k introduced cases and 55.8k audiences held in the year 2020-2021 [1]) and as the parties involved, especially tenants, are usually unfamiliar with or even intimidated by formal legal procedures [2], the Cyberjustice laboratory built a tool to facilitate access to legal information by landlords and tenants.

Such tool, whose preliminary foundations were laid by [3], was released in July 2021 as the JusticeBot[2], a decision-tree-like system in which laymen are guided through paths of questions. For each question, such as the one shown in Figure 1, users are given binary answers they have to choose from, so that they can be given more refined questions and

[1] Corresponding Author: Olivier Salaün, salaunol@iro.umontreal.ca
[2] JusticeBot, https://justicebot.ca/

more relevant information as they continue on a pathway that corresponds the most to their case. To ensure that each question and the implications of each optional answer are well understood, the user is given past decisions for illustration purposes. For instance, for the question in Figure 1, the concept of "being often late in paying the rent" has no straightforward definition. Therefore, the user is provided in the bottom light blue panel ten different decisions from the TAL, with one half in which the judge determined that the tenant was "often late" and another half in which it was not the case (shown in the bottom unwrapped panel).

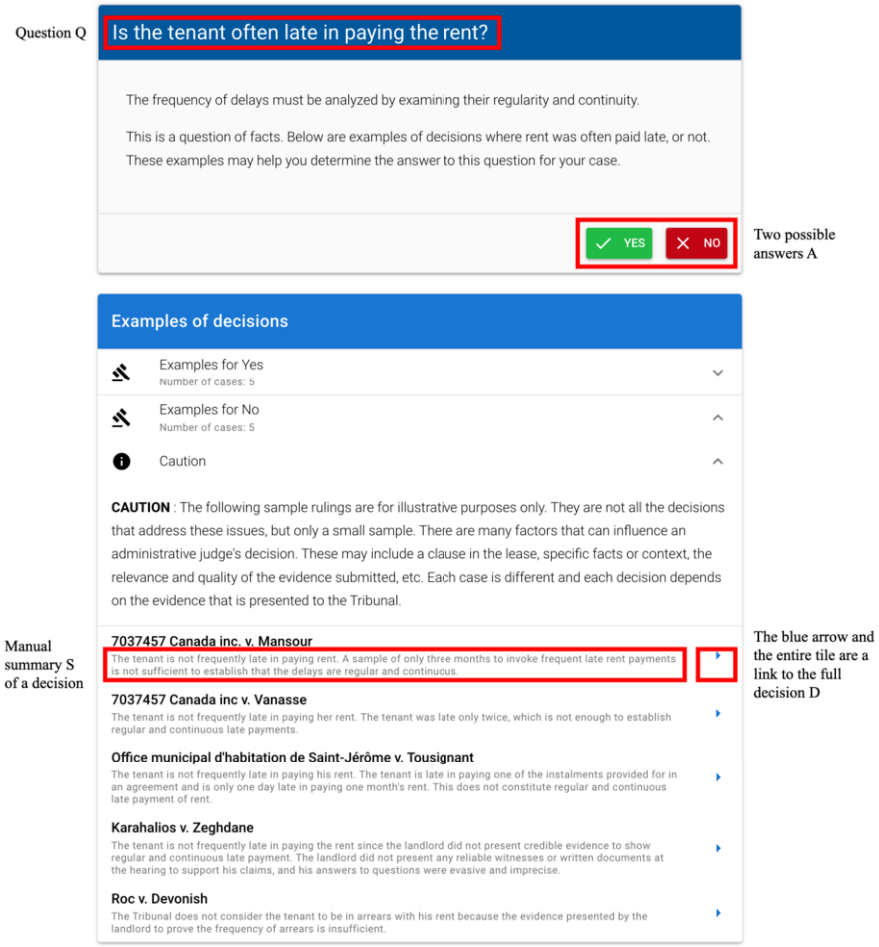

Figure 1. JusticeBot interface (translated from French) with a binary question: "Is the tenant often late in paying the rent?" The user can answer "Yes" or "No" as well as consult past court decisions that correspond to each answer as shown in the unwrapped bottom panel.

Although users have the possibility to read the full original case on CanLII[3] by clicking on it, we observed that 80.5% of them limit themselves to reading the short grey

[3]Canadian Legal Information Institute, https://www.canlii.org/

summary below the bold decisions titles. For each question-answer pair, relevant cases were manually collected and summarized by law graduate students following drafting instructions. Our goal is to investigate to what extent we can automate the summarization of court decisions for laymen within the scope of Quebec housing law.

2. Related Work

Automatic summarization applied to legal texts is generally framed as an extractive task. One of the earliest works was made by [4,5] who relied on rule-based thematic segmentation for selecting the most salient sentences for each section of Canadian judgments. A similar approach, based on entities extracted from text, was used by [6] for Australian court cases. In most experiments, in order to find the sentences that best summarize the entire judgment, authors generally use a scoring pipeline for deciding which sentences to add to the output summary. Such pipelines rely not only on the content of the sentence itself but also on its rhetorical role [7] (i.e. whether the sentence belongs to the Fact, the Issue or the Conclusion section of the case) whose importance was highlighted by [8], [9] and [10] for British, Taiwanese and Indian courts cases, respectively.

Besides these works based on sentence-extraction, more and more sophisticated models were developed for abstractive summarization by [11,12,13], but such approaches were mostly applied to news datasets. To the best of our knowledge, the only benchmark available for abstractive summarization of legal documents is BigPatent [14], though this corpus consists of patents and not court judgments as in the aforementioned extractive summarization tasks. Several reasons may explain the lack of accessible benchmarks for legal judgment abstractive summarization:

- Summaries written by legal experts are prohibitively costly to obtain, due to the length and complexity of decisions, plus the scarcity of legal practitioners who must be familiar with the target legal area ;
- Since decisions are significantly longer than documents in generic NLP corpora, they can be hard to process for models whose maximum input length is limited despite the emergence of transformer-based [15] models such as [16,17] that can process longer documents but at a high computing cost. Therefore, an extractive approach is more widespread for this type of document.

Unlike the aforementioned tasks, our summarization goal is slightly different as we aim at generating summaries intended for laymen with no prior legal knowledge instead of legal practitioners. Hence, an abstractive approach is preferred over an extractive one. Moreover, our task implies that the model makes a summary conditioned by a predefined triplet of question-answer-decision such as those shown on the interface. Finally, we must emphasize that our corpus is in Canadian French, a language not as widespread as English in the legal NLP field.

Moreover, our work does not fit the usual scope of legal summarization for domain experts as it is closer to that of other NLP experiments trying to make justice more accessible to laymen. For instance, [18] made a retrieval task in which the most relevant articles of Belgian law must be retrieved given a question asked by ordinary citizens. Similarly, [19] pursued the same goal with *plumitifs*, court dockets with complex abbreviations and legal jargon, that they tried to convert into intelligible texts for laymen through a text generation approach.

3. Data and Task Description

We extracted from the JusticeBot database and from CanLII a total of 156 instances. Although the dataset is small, it reflects diverse real-life issues faced by users. Each instance has 4 pieces of text:

- the question `Q` and the answer `A` from the JusticeBot interface ;
- the main text of a decision `D` that we extracted with heuristics from the HTML page from CanLII (metadata are removed) ;
- the summary `S` about `D` provided by an annotator to the JusticeBot user.

The task consists in mapping `Q`, `A` and `D` to the target summary `S`. The mean/median number of tokens for `Q`, `A`, `D` and `S` amount to 13/13, 1/1, 1030/745 and 44/41, respectively. Because of the dataset size, we will run our experiments with 10 folds, each fold having a train/validation/test split ratio of approximately 80:10:10. Performance results are averaged scores across folds and standard deviations are reported.

4. Models and Summary Generation Experiments

We decided to use the transformer-based encoder-decoder model BARThez [20] (same architecture as BART [13]) as it delivered state-of-the-art performance in French for news and dialogue summarization [21]. It is also a suitable starting point for making a legal-French-oriented language model [22].

4.1. Further Pretraining through Unsupervised Denoising Task

We used in our experiments two versions of BARThez: one with default pretrained parameters[4] called **Van**BART (**Van**illa BARThez), another called **FPT**BART in which default parameters are **F**urther **P**re**T**rained through the unsupervised denoising task with the FairSeq library [23]. In such a task, as described by [13], the model is given as input a corrupted version of a text segment from which it must generate the original one. The further pretraining corpus is made of 531,564 TAL decisions which we split into train and validation sets with an 80:20 ratio. Two resources frequently cited by TAL magistrates, 3.5k articles from Civil Code of Québec[5] (C.c.Q.) and TAL law[6], were also added to the train set. The denoising task is performed during 2 million steps with a 10^{-5} learning rate and 10^{-2} weight decay with Adam [24] optimization of cross-entropy loss. After roughly 12 days of pretraining on a single NVIDIA GeForce RTX 3090, the perplexity decreased from 1.78 to 1.33 on the validation set.

4.2. Supervised Text Summarization and Combinations of Text Inputs

Given text inputs `Q`, `A` and `D`, and target summary `S`, we tried several combinations of text inputs such as:

1. **D**: all paragraphs of all sections of **d**ecision `D` ;

[4]Pretrained checkpoint available at https://huggingface.co/moussaKam/barthez
[5]Code civil du Québec, RLRQ c CCQ-1991, http://canlii.ca/t/6b4rq
[6]Loi sur la régie du logement, RLRQ c R-8.1, http://canlii.ca/t/69m68

2. **QD**: a concatenation of **q**uestion `Q` and **d**ecision `D`'s paragraphs ;
3. **QDr**: same as above, but the paragraphs of decision `D` are in **r**everse order ;
4. **QAD**: a concatenation of **q**uestion `Q`, **a**nswer `A` and **d**ecision `D` ;
5. **QADr**: same as above, but the paragraphs of decision `D` are in **r**everse order.

Several reasons explain the reverting of paragraphs order in inputs 3 and 5:

- the annotators who drafted the summaries emphasized that the most relevant information was usually located towards the end of the decision, just before the verdict section. Therefore, they would tend to spend more time reading the pre-verdict part of the document as it gives the gist of the case instead of reading it from top to bottom. Such observations are consistent with those drawn by domain experts in the summarization task conducted by [6] ;
- although BART architecture has a larger maximum input sequence length (1024 tokens) with respect to commonly used transformer models (512 tokens for BERT [25]), reverting the paragraphs order of a court case allows minimizing the risk that important information located towards the end of the decision is not included in the input sequence of the model.

Table 1. Average scores in terms of automatic text generation metrics for each model and combination of text inputs (standard deviations are shown between parentheses and best scores are in bold font).

	Model and input combination	BLEU	ROUGE-1	ROUGE-2	ROUGE-L
VanBART	decision	7.4 (6.2)	31.0 (7.4)	14.4 (7.5)	25.0 (7.4)
	question + decision	8.2 (3.7)	34.5 (4.3)	16.5 (4.6)	27.5 (5.3)
	question + decision (reversed order)	9.9 (5.7)	35.1 (5.1)	18.4 (5.5)	28.6 (5.5)
	question + answer + decision	8.4 (2.8)	33.9 (3.5)	17.0 (3.5)	27.5 (3.5)
	question + answer + decision (reversed order)	9.1 (4.0)	34.3 (3.2)	17.0 (3.2)	27.8 (3.0)
FPTBART	decision	10.0 (5.2)	32.8 (5.9)	16.2 (6.1)	26.0 (5.9)
	question + decision	13.0 (5.8)	**37.7 (6.1)**	20.1 (6.1)	29.9 (6.5)
	question + decision (reversed order)	12.0 (4.9)	37.3 (4.5)	19.8 (5.8)	29.6 (4.8)
	question + answer + decision	**14.0 (6.7)**	37.1 (6.7)	20.5 (7.5)	29.9 (6.7)
	question + answer + decision (reversed order)	13.3 (6.4)	**37.7 (6.9)**	**20.8 (7.4)**	**30.1 (7.0)**

Each aforementioned combination of text is provided as input to both VanBART and FPTBART. For each fold, the model is fine-tuned with the Adam optimizer. Given the small dataset size, the batch size is 1. In order to smooth out the optimization of cross-entropy loss, we apply an initial learning rate of 10^{-4} with a scheduler that halves it at the end of each training epoch if the ROUGE-1 score does not improve on the validation set. The training is stopped if this score does not improve after 10 consecutive epochs. The model whose parameter setting achieves the highest ROUGE-1 score on the validation set is used for summary generation, in which the maximum number of output tokens and the number of beams for beam search are set to 200 and 3, respectively.

5. Results and Discussion

For each model and each combination of text inputs, Table 1 gives the average scores across 10 folds for BLEU [26], ROUGE-1, ROUGE-2, ROUGE-L [27] along with stan-

dard deviation. Overall, for a given input combination, FPTBART delivers a higher performance with respect to VanBART. Such improvement is consistent with the fact that unsupervised pretraining of a transformer model helps for tasks in specialized domain as shown by [28]. Considering FPTBART results, the combination of question, answer and decision (QAD and QADr) seems to perform best in terms of ROUGE and BLEU as they contain more information and slightly outperform the combination of question and decision only (QD). Still, as it is hard to appreciate such syntax-based measures given the dataset size and the nature of our experiment, we retained output summaries from settings QD, QAD and QADr for manual evaluation.

5.1. Manual Evaluation

For a given fold, we took 16 test instances for which we considered a total of 48 output summaries generated by FPTBART with inputs QD, QAD and QADr. Three experts (co-authors of this paper), including one NLP specialist and two law graduate students, evaluated these summaries with an online form[7]. On the basis of guidelines provided by [29,30], we designed an intrinsic evaluation framework with two parts. The first one is related to the form (fluency) of candidate summaries:

- **1.0 grammar**: does the candidate summary contain any grammar or spelling mistakes?
- **1.1 readability**: does the summary contain repeated words ("hallucinations")? Does it make intelligible sense?
- **1.2 style**: is the choice of words appropriate for a JusticeBot layman user?

The latter part is related to the summary usefulness (adequacy) with respect to the JusticeBot's objective to ease access to legal information:

- **2.0 adequacy with respect to the decision**: does the candidate summary accurately reflect the use case and the relevant elements described in the decision?
- **2.1 adequacy with respect to the question**: does the summary address the issue described in the question shown to the JusticeBot user?
- **2.2 linking the decision and the question**: does the summary explain how the decision illustrates the answer suggested to the question? Is the summary meaning consistent with the answer?
- **2.3 consistency with manual summary**: is the generated summary consistent with the elements provided in the manual one already displayed in the JusticeBot?

All criteria are assessed by evaluators on a 4-point ordinal scale. The scores are shown in Table 2 along with Krippendorff's alphas (KA), a measure of inter-evaluator agreement[8]. Average instance-wise and question-wise KA across evaluators amounts to 0.525, but such a value hides disparities. As shown in Table 2, the KA for fluency questions are close to 0 and even negative, denoting a lack of agreement among evaluators despite efforts made to make each criterion as clear as possible. This could also be due to overly specific fluency questions. On the other hand, the agreements are more noticeable for adequacy questions, especially for questions 2.2 and 2.3. On the basis of unweighted

[7] The evaluation form is available at https://forms.gle/uX8n4LuQ5sddxsfD8

[8] Krippendorff's alpha ranges from −1 to 1. 1 denotes perfect agreement, 0 denotes absence of agreement beyond chance, and negative alpha indicates disagreement [31].

Table 2. Average of manual evaluations scores for each criterion (scale from 1 to 4 included) along with Krippendorff's alphas. For each criterion, the highest average of evaluators' scores is in bold font.

	Manual evaluation criterion	Krippendorff's alpha	Average of evaluators' scores		
			QD	QAD	QADr
Fluency	1.0 grammar	-0.093	3.73	3.69	**3.90**
	1.1 readbility	0.059	3.56	3.54	**3.62**
	1.2 style	-0.200	3.56	**3.62**	**3.62**
	Unweighted average of fluency scores		*3.62*	*3.62*	***3.72***
Adequacy	2.0 adequacy with respect to decision	0.490	2.90	3.21	**3.00**
	2.1 adequacy with respect to question	0.621	2.96	**3.17**	2.94
	2.2 linking decision and question	0.736	2.69	**2.83**	2.62
	2.3 consistency with manual summary	0.776	2.54	**2.67**	2.50
	Unweighted average of adequacy scores		*2.77*	***2.97***	*2.77*
	Unweighted average of all scores		*3.13*	***3.25***	*3.17*

average of all manual scores, the FPTBART achieves the best performance with QAD (3.25) input followed by QADr (3.17) and QD (3.13). The fact that QAD outperforms QADr for adequacy criteria but underperforms for fluency criteria suggests that reversing decision paragraphs order within the text input has little influence on output summaries.

5.2. Correlation among Automatic and Manual Metrics

Given the important manual evaluation cost, in particular in the specialized domain of housing law, we tried to find whether some automatic metrics can be used as proxies for the different manual evaluation criteria. We took the scores obtained in the previous subsection 5.1 for summaries generated by FPTBART with input combinations QD, QAD and QADr. For each pair of candidate and reference summaries, we compute automatic metrics available for evaluation of text generation: ROUGE-1, ROUGE-2, ROUGE-L, BERTscore [32], BLEU, chrF [33]. Once we have the manual criteria (MC) scores on one hand and the automatic metric (AM) ones on the other hand, we compute a correlation matrix (Kendall's τ coefficients) among these two sets of metrics that is shown as a heatmap in Figure 2. Overall, correlations between AM and fluency-related MC struggle to exceed 0.5 as shown in the region surrounded by green dashed lines. On the contrary in the region surrounded by solid blue edges, adequacy-related scores MC and AM have higher correlations. Computing the average correlation of each AM with these four MC suggests that ROUGE-1 (0.664) and ROUGE-2 (0.633) are the best proxies for adequacy-related metrics, despite them being merely syntax-based.

5.3. Qualitative Analysis

Upon manual examination, some summaries appeared easier to generate than others, especially those that consisted in a single sentence and/or are related to rent arrears (the most frequent issue in our dataset and in real life), although models also struggle with time duration. This is shown in Example (a) in Figure 3 where all candidate summaries accurately describe the tenant as being late in payment, with QADr adding 3 months of delay. For more complex cases, on the contrary, output summaries are less convincing as models tend to repeat phrases used in somewhat similar cases but without properly

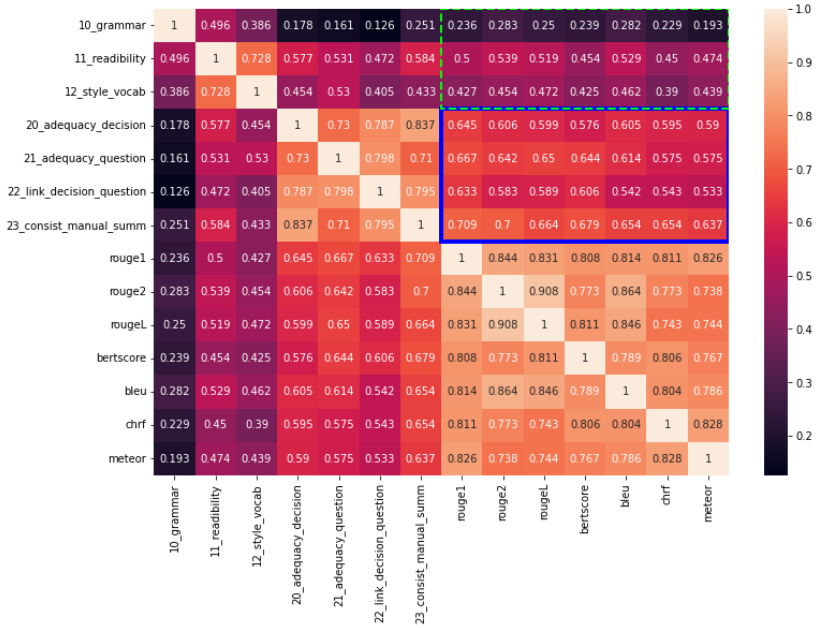

Figure 2. Heatmap of a correlation matrix among automatic and manual metric scores for summaries generated by FPTBART with input combinations QD, QAD and QADr. Green and blue rectangles highlight correlation values between automatic metrics with fluency criteria and adequacy criteria, respectively.

addressing the instance at hand. Example (b) provides a good illustration with the blue highlighted segments that contradicts the reference summary. Moreover, the last candidate (QADr), although being consistent with the input triplet and target summary, is making an extrapolation about article 1943 C.c.Q. that indeed makes the notice invalid but specifies nothing regarding what language should be used. Therefore, that language issue was handled by the court itself. As shown by these examples, automated summarization may generate summaries with correct wording at first sight. However, the possibility of them containing legal errors still makes a manual evaluation by experts necessary.

6. Conclusion

We implemented a novel conditional text generation task aimed at making summaries of court decisions for laymen by employing a state-of-the-art generative model. In addition to using automatic metrics (AM), we designed domain-specific manual criteria (MC) for sensible human evaluation and to assess to what extent generated summaries meet JusticeBot's user requirements. Overall, the best performance in terms of AM and MC was achieved by a further-pretrained BARThez that gets a question-answer-decision triplet as input, but a closer examination of output summaries reveals large room for improvement. As future work, we plan to: obtain a much larger dataset that covers more cases ; find methods to guide and control summary generation at a finer grain in order to incorporate legal reasoning ; examine how transformer attention weights handle each component of triplet input.

(a)	**Input question:** Is the tenant more than 3 weeks late in paying rent? **Input answer:** Yes **Reference summary:** The tenant is more than 3 weeks late in paying the rent as he has not paid his rent for 2 months. **QD, QAD:** The tenant is more than 3 weeks late in paying rent as he has not paid his rent for 2 consecutive months. **QADr:** The tenant is more than 3 weeks late with his rent as he has not paid his rent for 5 consecutive months.
(b)	**Input question**: Does the notice sent by your landlord comply with the law? **Input answer:** No **Reference summary:** The notice is not valid because it is written in French while the lease is in English and this error causes prejudice to the tenant. **QD:** The notice is wrong because it is not written in French while the lease is in English. However, despite this error, the tenant understood the content of the notice very well and was not prejudiced in any way by this situation. **QAD:** The notice is written in French whereas the lease is in English. However, despite this error, the tenant understood the content of the notice very well and was not prejudiced in any way. **QADr:** The notice of modification of the lease does not comply with the requirements of article 1943 C.c.Q. since it is written in French whereas the lease is in English.

Figure 3. Examples of summaries (translated from French) generated by FPTBART. QD, QAD and QADr correspond to the input used.

Acknowledgements We would like to thank the Cyberjustice Laboratory at the Université de Montréal, the LexUM Chair on Legal Information and the Autonomy through Cyberjustice Technologies project for supporting this research.

References

[1] Tribunal administratif du logement. Rapport annuel de gestion 2020-2021; 2021. Report retrieved on 11th of August 2022 from https://www.tal.gouv.qc.ca/sites/default/files/Rapport_annuel_2020-2021.pdf.

[2] Gallié M, Brunet J, Laniel RA. Les expulsions pour arriérés de loyer au Québec: un contentieux de masse. McGill Law Journal/Revue de droit de McGill. 2016;61(3):611-66.

[3] Westermann H, Walker VR, Ashley KD, Benyekhlef K. Using Factors to Predict and Analyze Landlord-Tenant Decisions to Increase Access to Justice. In: Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law; 2019. p. 133-42.

[4] Farzindar A, Lapalme G. LetSum, an automatic Legal Text Summarizing. In: Legal knowledge and information systems: JURIX 2004, the seventeenth annual conference. vol. 120. IOS Press; 2004. p. 11.

[5] Farzindar A, Lapalme G. Legal Text Summarization by Exploration of the Thematic Structure and Argumentative Roles. In: Text Summarization Branches Out. Barcelona, Spain: Association for Computational Linguistics; 2004. p. 27-34.

[6] Polsley S, Jhunjhunwala P, Huang R. CaseSummarizer: A System for Automated Summarization of Legal Texts. In: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: System Demonstrations. Osaka, Japan: The COLING 2016 Organizing Committee; 2016. p. 258-62.

[7] Saravanan M, Ravindran B. Identification of rhetorical roles for segmentation and summarization of a legal judgment. Artificial Intelligence and Law. 2010;18(1):45-76.

[8] Hachey B, Grover C. Extractive summarisation of legal texts. Artificial Intelligence and Law. 2006;14(4):305-45.

[9] Liu CL, Chen KC. Extracting the gist of Chinese judgments of the supreme court. In: proceedings of the seventeenth international conference on artificial intelligence and law; 2019. p. 73-82.

[10] Bhattacharya P, Poddar S, Rudra K, Ghosh K, Ghosh S. Incorporating Domain Knowledge for Extractive Summarization of Legal Case Documents. In: Proceedings of the 18th International Conference on Artificial Intelligence and Law (ICAIL); 2021. p. 22-31.

[11] Nallapati R, Zhou B, dos Santos C, Gulçehre Ç, Xiang B. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In: Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning; 2016. p. 280-90.

[12] See A, Liu PJ, Manning CD. Get To The Point: Summarization with Pointer-Generator Networks. In: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics; 2017. p. 1073-83.
[13] Lewis M, Liu Y, Goyal N, Ghazvininejad M, Mohamed A, Levy O, et al. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics; 2020. p. 7871-80.
[14] Sharma E, Li C, Wang L. BIGPATENT: A Large-Scale Dataset for Abstractive and Coherent Summarization. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics; 2019. p. 2204-13.
[15] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. In: Advances in neural information processing systems; 2017. p. 5998-6008.
[16] Zhang J, Zhao Y, Saleh M, Liu P. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. In: International Conference on Machine Learning. PMLR; 2020. p. 11328-39.
[17] Zaheer M, Guruganesh G, Dubey KA, Ainslie J, Alberti C, Ontanon S, et al. Big bird: Transformers for longer sequences. Advances in Neural Information Processing Systems. 2020;33:17283-97.
[18] Louis A, Spanakis G. A Statutory Article Retrieval Dataset in French. In: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics; 2022. p. 6789-803.
[19] Beauchemin D, Garneau N, Gaumond E, Déziel PL, Khoury R, Lamontagne L. Generating Intelligible Plumitifs Descriptions: Use Case Application with Ethical Considerations. In: Proceedings of the 13th International Conference on Natural Language Generation. Dublin, Ireland: Association for Computational Linguistics; 2020. p. 15-21.
[20] Kamal Eddine M, Tixier A, Vazirgiannis M. BARThez: a Skilled Pretrained French Sequence-to-Sequence Model. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing; 2021. p. 9369-90.
[21] Zhou Y, Portet F, Ringeval F. Effectiveness of French Language Models on Abstractive Dialogue Summarization Task. In: LREC 2022; 2022. p. 3571-81.
[22] Garneau N, Gaumond E, Lamontagne L, Déziel PL. CriminelBART: a French Canadian legal language model specialized in criminal law. In: Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law; 2021. p. 256-7.
[23] Ott M, Edunov S, Baevski A, Fan A, Gross S, Ng N, et al. fairseq: A Fast, Extensible Toolkit for Sequence Modeling. In: Proceedings of NAACL-HLT 2019: Demonstrations; 2019. p. 48-53.
[24] Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. In: ICLR (Poster); 2015. .
[25] Devlin J, Chang MW, Lee K, Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In: Proceedings of NAACL-HLT; 2019. p. 4171-86.
[26] Papineni K, Roukos S, Ward T, jing Zhu W. BLEU: a Method for Automatic Evaluation of Machine Translation; 2002. p. 311-8.
[27] Lin CY. ROUGE: A Package for Automatic Evaluation of Summaries. In: Text Summarization Branches Out. Barcelona, Spain: Association for Computational Linguistics; 2004. p. 74-81.
[28] Zheng L, Guha N, Anderson BR, Henderson P, Ho DE. When does pretraining help? assessing self-supervised learning for law and the casehold dataset of 53,000+ legal holdings. In: Proceedings of the Eighteenth International Conference on Artificial Intelligence and Law; 2021. p. 159-68.
[29] Celikyilmaz A, Clark E, Gao J. Evaluation of text generation: A survey. arXiv preprint arXiv:200614799. 2020.
[30] Howcroft DM, Belz A, Clinciu MA, Gkatzia D, Hasan SA, Mahamood S, et al. Twenty Years of Confusion in Human Evaluation: NLG Needs Evaluation Sheets and Standardised Definitions. In: Proceedings of the 13th International Conference on Natural Language Generation; 2020. p. 169-82.
[31] Zapf A, Castell S, Morawietz L, Karch A. Measuring inter-rater reliability for nominal data–which coefficients and confidence intervals are appropriate? BMC medical research methodology. 2016;16(1):1-10.
[32] Zhang T, Kishore V, Wu F, Weinberger KQ, Artzi Y. BERTScore: Evaluating Text Generation with BERT. In: International Conference on Learning Representations; 2020. .
[33] Popović M. chrF: character n-gram F-score for automatic MT evaluation. In: Proceedings of the Tenth Workshop on Statistical Machine Translation. Lisbon, Portugal: Association for Computational Linguistics; 2015. p. 392-5.