Generating Intelligible *Plumitifs* Descriptions: Use Case Application with Ethical Considerations

David Beauchemin[*1], **Nicolas Garneau**[*1], **Eve Gaumond**[*2], **Pierre-Luc Déziel**[2], **Richard Khoury**[1], and **Luc Lamontagne**[1]

[1]Department of Computer Science, Université Laval, Québec, Canada
[2]Faculty of Law, Université Laval, Québec, Canada

david.beauchemin.5@ulaval.ca, eve.gaumond@observatoire-ia.ulaval.ca
{nicolas.garneau,richard.khoury,luc.lamontagne}@ift.ulaval.ca
pierre-luc.deziel@fd.ulaval.ca

Abstract

Plumitifs (dockets) were initially a tool for law clerks. Nowadays, they are used as summaries presenting all the steps of a judicial case. Information concerning parties' identity, jurisdiction in charge of administering the case, and some information relating to the nature and the course of the preceding are available through *plumitifs*. They are publicly accessible but barely understandable; they are written using abbreviations and referring to provisions from the Criminal Code of Canada, which makes them hard to reason about. In this paper, we propose a simple yet efficient multi-source language generation architecture that leverages both the *plumitif* and the Criminal Code's content to generate intelligible *plumitifs* descriptions. It goes without saying that ethical considerations rise with these sensitive documents made readable and available at scale, legitimate concerns that we address in this paper. This is, to the best of our knowledge, the first application of *plumitifs* descriptions generation made available for French speakers along with an ethical discussion about the topic.

1 Introduction

The right to access judicial information is a fundamental component of Canadian democracy and its judicial process (*Vancouver Sun (Re)*, 2004; *CBC. v Canada (A.G)*, 2011)[1]. This right has two main purposes. First, to enhance judicial accountability by providing opportunities to the public to scrutinize it and put forward criticisms of the judicial process (*Sierra Club of Canada v Canada (Minister of Finance)*, 2002; *CBC. v New Brunswick (A.G)*, [1996]). Second, it has an educational purpose: by accessing judicial information, people acquire a better understanding of the court process (*Edmonton Journal v Alberta (A.G)*, [1989]). Given these purposes, the necessity to provide access to judicial information in an intelligible form cannot be ignored. Indeed, getting a copy of a document is not enough; people have to understand its contents. This is particularly crucial in a digital context since citizens face an overload of judicial information online (Eltis, 2011). As a consequence, litigants have great difficulty in finding relevant information for their case online (Dionne, 2019).

Studies show that, in the province of Quebec, the *plumitif* (a public register where one can find an official trace of all the actions taken by the courts) lacks intelligibility (Tep et al., 2019). Some users have called it "non-sense" for non-attorneys (Parada et al., 2020). Yet, the *plumitif* is necessary for every litigant as it provides information concerning the parties' identity, the jurisdiction responsible for administering cases, and information relating to the nature and the course of proceedings. In this work, we aim at leveraging both information extraction and natural language generation to increase the intelligibility of excerpts of the Court of Quebec's *plumitif* regarding criminal offenses under the Criminal Code of Canada (CCC).

Improving the comprehension of textual legal documents has been the subject of several studies in the past. For example, patent claims are long legal pieces of texts that contain complex sentences making it hard for a layperson to reason about. Sheremetyeva (2014) framed this problem into an automatic text simplification task while Farzindar et al. (2004) and Hachey and Grover (2006) proposed extractive summarization techniques to make them easier to understand. The *plumitifs*, while also lying in the "legal texts" family, take a completely different form; they are not written in a valid grammatical form, and contain many abbreviations and references to the CCC. This makes our use case application rather unique.

To handle this type of document, we have de-

[*]Contributed equally to this work.
[1]Italic references refer to case laws.

Proceedings of The 13th International Conference on Natural Language Generation, pages 15–21, Dublin, Ireland, 15-18 December, 2020.

signed a *description generation* pipeline, divided into three steps. The first step consists of segmenting a *plumitif* into different parts. In the second step, we extract, for each part, the relevant information using a Named Entity Recognition (NER) model. For the final step, we generate sentences from the data extracted by the NER model. To this end, we use a template-filling approach to ensure there are no factual fallacies introduced in the generation, an essential concern in legal text generation. Moreover, we use a statistical language model in a controlled setting to augment the generation with vital contextual information, namely texts from the CCC, making our approach a hybrid generation model. Our contributions, in this work, are twofold;

1. We propose a simple yet robust data-to-text multi-source textual generation pipeline to make *plumitifs* easier to understand for the litigants (made available through a web application, see Appendix I);
2. We bring a discussion on the ethical considerations about privacy and discrimination that such an application may cause.

We further describe our architecture, related work and methodology in Section 2 and evaluate its generation capabilities in Section 3. We bring important ethical considerations in Section 4 and open the discussion for future work in Section 5.

2 Generating Intelligible *Plumitifs* Summaries

Plumitifs are used as summaries presenting all the steps of a case heard by the court. In the context of criminal proceedings, they contain information about the plaintiff, the accused, different charges along with their associated penalty (if applicable). We present a *plumitif* example in Appendix A, Figure 2. *Plumitifs* are freely available in person at any courthouse and are also accessible on the *Société québécoise d'information juridique* (SOQUIJ) website[2] where they can be consulted for a fee. In this section, we detail our proposed architecture, which is broken down into three steps; segmenting the *plumitif* into parts, extracting relevant information from each part, and generating descriptions by also leveraging the CCC. We illustrate the whole architecture in Appendix B, Figure 4 and further detail each component in the following subsections.

[2]https://soquij.qc.ca/

2.1 Segmenting the *Plumitif*

We identify three parts in a *plumitif*; the accused, the plaintiff, and the charges. Since the *plumitif* structure is pretty regular, it allows us to identify each one using simple heuristics based on the presence of specific strings (e.g. "ACC." for "accused") with 100% accuracy. Splitting into parts simplifies the NER step since these models typically use a narrow contextual window of a few tokens on either side to make their prediction. It also provides more data points overall.

2.2 Extracting Relevant Information

As mentioned in Section 1, we frame the retrieval of the relevant entities in the *plumitif* as an information extraction problem. That is, given a raw part of the *plumitif*, a NER model extracts entities from the text to fill in a normalized view. We established nine types of entities that need to be extracted; *Adresses* (Addresses), *Accusations et spécifications d'accusations* (Charges and Charges Specifications), Dates, *Décisions* (Decisions), *Lois* (Laws), Accusations, *Organisations* (Organizations), *Personnes* (Persons), *Plaidoyer* (Pleas) and *Peines* (Sentences). For the rest of the paper, we will use the French entities within the French templates and rules, and the English entities otherwise (i.e. in the text).

We manually annotated 816 *plumitifs* from eight districts over the last five years, to cover as much variety as possible. These eight districts are the ones with the most cases for this date range. We train a NER model on the annotated dataset, which achieves, on average, a F1-Score of 0.965, thanks to the regularity in the form the *plumitifs* can take[3].

Once the relevant information is extracted and normalized, we use it in the third step of the pipeline, which consists of a data-to-text generation model, described in the following subsection.

2.3 Realisation of *Plumitif* Summaries

Even though statistical and deep Natural Language Generation (NLG) has seen tremendous breakthroughs in recent years (Radford et al., 2018, 2019; Brown et al., 2020), we decide not to strictly rely on this kind of Transformer model (Vaswani et al., 2017) for our use case. Several architectures (Ziegler et al., 2019; Keskar et al., 2019; Dathathri et al., 2020) attempt to control the generation of

[3]Training details are available in Appendix C

such pre-trained models by using conditioning elements that propose a specific stylistic or emotion for example. However, Brown et al. (2020) showed that one of the best neural language models to date (GPT-3) may generate non-factual utterances, often called hallucinations (Rohrbach et al., 2018; Rebuffel et al., 2020), or even hide significant biases that may put the credibility of generation at stake.

Since we generate legal textual content that can be used in various sensitive applications (e.g. HR screening, (Parada et al., 2020)), we can't afford to let a model "statistically" generate a non-factual decision (e.g. guilty but the accused is not) or a charge (e.g. something that the accused has not done). Thus, we prefer to sacrifice variability for control by using a template-filling approach. Puzikov and Gurevych (2018) showed that a template-based approach can be as good as a neural encoder-decoder model on generating restaurant descriptions from sets of key-value pairs. Deemter et al. (2005) also argues that "template-based approaches to the generation of language are not necessarily inferior to other [statistical] approaches as regards their maintainability, linguistic well-foundedness and quality of output". This approach has been shown recently to perform well in different areas like weather reports (Ramos-Soto et al., 2015), financial analysis (Nesterenko, 2016) and soccer game reports (van der Lee et al., 2017) where they are used in production.

2.3.1 Template-Based, Data-to-Text Generation

In the same way Deemter et al. (2005) did, we manually deduce 66 patterns from a subset of the *plumitifs* to generate the description text using the extracted information from the model introduced in Section 2.2[4]. The generation rules (especially the sentence ones) have been written by a legal expert. Following the example in Figure 2, with the corresponding extracted information about the accused and a really simple yet efficient rule, we can generate texts about the accused and the plaintiff, as illustrated in Appendix D.

In the next subsection, we present how we combine the information extracted from the *plumitif* with a parsed version of the CCC [5] using a Masked Language Model.

[4]We present a complete generation example in Appendix H based on the *plumitif* presented in Figure 2.

[5]`https://laws-lois.justice.gc.ca/eng/acts/c-46/`

2.3.2 Leveraging the Criminal Code of Canada

The *Criminal Code of Canada* (CCC) is an act that contains most of the criminal law in Canada. It contains around 1,500 provisions (referred to with numbers) where each of them comprises paragraphs and subparagraphs. The *plumitifs* refers to provisions from the law using only the provision numbers, which provides little to no context to the litigants. Therefore, it is essential to extract the law's text from the Criminal Code when generating the *plumitif*'s summary. However, the CCC is only available in HTML or PDF format, making it hard to query it programmatically. Thus, we parsed the HTML version into the JSON format, which allows us to easily query for different articles, paragraphs and subparagraphs [6].

A *plumitif* may contain several charges. Each charge may refer to one or two provisions from the law. The first provision is most likely referring to the description of the law, where the title briefly summarizes the description. The second provision (if any) is usually there to specify the charge [7].

Given the following template (see Appendix G for a translated version);

<Accusé> `est accusé` **<Article>**.

we wish to insert the provision title syntactically. To this end, we propose to "stitch" the two pieces of the template using a Masked Language Model. We use the French pre-trained version of BERT (Devlin et al., 2019), CamemBERT (Martin et al., 2020), which has been trained on the French subset of OSCAR (Ortiz Suárez et al., 2020), a huge multilingual corpus obtained by language classification and filtering of the Common Crawl corpus.

One of BERT's abilities is to predict randomly masked tokens in a sentence, usually referred to as a *Cloze* task in the literature (Taylor, 1953). We specifically leverage this ability to our benefit, and let CamemBERT predict the proper preposition that should be inserted between the template and the charge's title (*défaut de se conformer à une ordonnance* here). The realisation of the previous template would then look like the following (Appendix G);

John Doe `est accusé` **pour** `défaut de se conformer à une ordonnance.`

[6]We were able to properly extract the 1518 provisions publicly release the JSON version of the French CCC here: `https://bit.ly/3kiBdFd`

[7]In this work, we do not leverage the second provision.

Using the 134 unique charges titles included in our dataset, we find that CamemBERT can predict the right preposition 84% of the time.

2.3.3 Pleas, Decisions and Sentences

The generation of the pleas and decision text is simple since there are only a few possible situations, using 14 generation rules out of the 66 deduced. For the first, it is either guilty or not guilty. For the second, it is guilty, not guilty, or ten other technical situations such as "arret" (i.e. case where the court orders a stay of proceedings). In both cases, the mapping between the pleas and decision is one-to-one with the associated generated text (i.e. a guilty decision can generate only one text). We illustrate this case in Appendix E.

On the other hand, generating Sentences is more complex. In our set of 66 deduced generation rules, 50 are used to generate the Sentences. This complexity is mostly due to the occurrences of different convictions in one Sentence, meaning that the mapping is one-to-many (i.e. a Sentence can have an unknown number of convictions). Given the Sentence's extracted convictions, we order them by types (i.e. the penalty inflicted of, fines and fees, community work, other convictions, probation and surcharge) and fill-in an "on-the-fly merged generation template" given the list of convictions. It is important to note that generation rules are not applied "in cascade" i.e. for a given list of convictions, there is one possible generation template. We illustrate the generation of the first Sentence's section in Appendix F.

3 Evaluating the Realisation of the Summaries

Since our generation model mostly relies on rules, it is straightforward to evaluate its performance; we first need to make sure all the relevant information is fully extracted (NER step) and that it properly fills in the corresponding template (generation step). We thus quantify our model's performance in terms of "Error Rate" where a generation error is the lack of realizing a specific part (accused, plaintiff or list of charges paragraphs), instead of evaluating the textual generation. The counts are computed per text. Errors are split into two categories; Extraction-based Errors (EE) and Generation-based Errors (GE). For clarity, we display the Errors Rates by districts in Table 1.

In most cases, we find that a wrong extraction of the Plaintiff (due to the NER model) causes EE. We can see that Granby and Sherbrooke have the highest EE rate; this is mostly due to the many different values an Organisation can take in these districts [8].

GE are mainly due to edge cases found in *plumitifs* which our rules do not cover. As we can see from the GE Rates in Table 1, our generation rules commit most errors on the Montréal, Sherbrooke and Gatineau districts. This is due to the numerous and diverse convictions these *plumitifs* hold. For example, a particular combination of convictions may not be associated to any generation rule. We illustrate this problem with an example in Figure 1, where the Sentence comprises multiple convictions and are essentially edge cases about the duration.

District	**EE**	**GE**	***Plumitifs***
Chicoutimi	0.0%	0.0%	9
Gatineau	6.7%	6.7%	15
Granby	33.3%	5.6%	18
Longueuil	5.9%	0,0%	17
Montréal	13.8%	9.2%	65
Québec	0.0%	0.0%	18
Sherbrooke	25.0%	8.3%	12
Trois-Rivières	15.4%	0.0%	13
Average	**13%**	**5%**	

Table 1: Error rates of the Extraction (EE) and Generation (GE) errors for each district.

PROBATION DE 2 ANS SURV.
PROBATION DPAC:8.5MS/EMPR:6.5M
TC 75 HS DEL 12 MS/SUIVI PROB 1 1/2 AN

Figure 1: Example of a complex Sentence containing a edge case about the duration of the different convictions. For this specific example, our model failed to generate a meaningful piece of text.

This highlights the need to have a better model at parsing *and* generating Sentences' paragraphs. Using a generative, sequence-to-sequence model, such as the one proposed by (Bahdanau et al., 2015) may be a better option, but we leave this study as future work. All in all, our model achieves low Error Rates (13% EE and 5% GE on average), allowing simple yet accurate textual generation of intelligible plumitifs. While these results are interesting, it raises some ethical concerns, that we discuss in the next section.

[8]This corroborates with the results of the NER model for the entity **Organisation**, in Section 2.2

4 Ethical Considerations

There is some ethical considerations regarding our dataset's privacy that ought to be addressed. *Plumitifs* contain sensitive information such as the names, dates of birth, addresses and criminal backgrounds of accused people. The identity of judges, plaintiffs, clerks, and attorneys taking part in a criminal case are also found in the *plumitifs*. As explained in Section 1, all of this information must be publicly accessible. As long as this data is protected by practical obscurity [9], the actual risks from public access of this information are limited (Vermeys, 2016).

However, if this data was to be released in bulk to the scientific community, it would not be "scattered [. . .] bits of information" (*US Department of Justice v. Reporters Committee for Freedom of the Press*, 1989) that require time and resources to retrieve anymore. Information could be easily searched, aggregated or combined with information from other public sources. This poses a risk to the privacy of judicial stakeholders.

In this subsection, we explain why we decided not to release our data set publicly (raw or synthesized). To put it in straightforward terms: information collected in public records should not be "up for grabs". Its use can result in privacy violations. This is especially true in the digital context where aggregation, linkage and analytics are made easier (Martin and Nissenbaum, 2017). There are several examples of privacy violations that occurred due to the malicious use of judicial information that was publicly accessible. For instance, more than 270 cases of identity theft have been linked to a security lapse in an American Municipal Court's website. (Bailey and Burkell, 2017). The Office of the Privacy Commissioner of Canada had to intervene to end an extortion scheme relying on data available from the Canadian Legal Information Institute and SOQUIJ's websites (*A.T. v Globe24h.com*, 2017). United State's "Public Access to Court Electronic Records" system made the identity of some cooperating defendants and undercover agents publicly available, which contributed to the intimidation and harassment of witnesses in order to discourage them from testifying (Eltis, 2011). There have also been some documented cases of discrimination in the context of employment (Solove, 2002) and housing (*Gichuru v Purewal and another*, 2017) caused by judicial information available online. Moreover, academics have expressed significant concerns about the secondary use of judicial information for marketing purposes.

This is now prohibited by the Personal Information Protection and Electronic Documents Act, (Office of the Privacy Commissionner of Canada, 2014), but (Bailey and Burkell, 2017) argues that this regulatory framework is not sufficient to prevent inappropriate uses of judicial data. Our team is currently working to develop a framework for the management of personal information contained in digital court records. However, for the moment, since the law provides no satisfactory solution, we chose not to release the dataset used to train our algorithm.

[9] A term broadly used to explain that documents might be accessible to all in principle, but that the access is hindered by some obstacles such as fees to consult a document or the need to go physically to a location - as is the case for the *plumitif*.

5 Conclusion and Future Work

In this paper, we introduce a simple yet effective multi-source architecture able to generate digestible *plumitifs* for Canadian citizens. We also show that we are in a position to easily divulge who has been accused of what and the outcome of it, which raises some important ethical concerns. In the future, we plan to explore statistical natural language generation further by using case law, provide more diverse *plumitifs* descriptions and improve the generation of Sentences. Finally, we hope that our application will provide better insights to the community and give the right direction for the next applications of not only NLG, but Machine Learning in general, in the field of law.

6 Acknowledgements

We thank the reviewers for their insightful comments on our manuscript. This research was enabled in part by the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Social Sciences and Humanities Research Council (SSHR). Also, this research was made possible with the help of our partner, SOQUIJ.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Jane Bailey and Jacquelyn Burkell. 2017. Privacy as Contextual Integrity. *Ottawa Law Review*, 48.

Tom B. Brown, Benjamin Pickman Mann, Nick Ryder, Melanie Subbiah, Jean Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, G. Krüger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric J Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language Models are Few-Shot Learners. *ArXiv*, 2005.14165.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric C. Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and Play Language Models: A Simple Approach to Controlled Text Generation. *ArXiv*, 1912.02164.

Kees Van Deemter, Mariët Theune, and Emiel Krahmer. 2005. Real versus Template-Based Natural Language Generation: A False Opposition? *Computational Linguistics*, 31:15–24.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

Alexandra Bahary Dionne. 2019. L'accès à la justice en contexte numérique : l'information juridique par et pour les justiciables sur les médias sociaux. *Windsor Yearbook of Access to Justice*, 35:337–362.

Karen Eltis. 2011. The judicial system in the judicial age: revisiting the relashionship between privacy and accessibility in the cyber context. *MLJ*, 2.

Atefeh Farzindar, Guy Lapalme, and Jean-Pierre Desclés. 2004. Résumé de textes juridiques par identification de leur structure thématique : Résumé automatique de textes.

Ben Hachey and Claire Grover. 2006. Extractive summarisation of legal texts. *Artificial Intelligence and Law*, 14:305–345.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing.

Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL: A Conditional Transformer Language Model for Controllable Generation. *ArXiv*, 1909.05858.

Chris van der Lee, Emiel Krahmer, and Sander Wubben. 2017. Pass: A dutch data-to-text system for soccer, targeted towards specific audiences. In *INLG*.

Kristen Martin and Helen Nissenbaum. 2017. Privacy Interests in Public Records: An Empirical Investigation. *Harvard Journal of Law & Technology*, 31.

Louis Martin, Benjamin Muller, Pedro Javier Ortiz Suárez, Yoann Dupont, Laurent Romary, Éric Villemonte de la Clergerie, Djamé Seddah, and Benoît Sagot. 2020. CamemBERT: a Tasty French Language Model. In *ACL*.

Liubov Nesterenko. 2016. Building a system for stock news generation in russian. In *WebNLG*.

Office of the Privacy Commissionner of Canada. 2014. Publicly available information - interpretation bulletin.

Pedro Javier Ortiz Suárez, Laurent Romary, and Benoît Sagot. 2020. A monolingual approach to contextualized word embeddings for mid-resource languages. In *ACL*, pages 1703–1714.

Alexandra Parada, Sandrine Prom Tep, Florence Millerand, Pierre Noreau, and Anne-Marie Santorineos. 2020. Digital Court Records : a Diversity of Uses. *IJR*, 9.

Yevgeniy Puzikov and Iryna Gurevych. 2018. E2e nlg challenge: Neural models vs. templates. In *INLG*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Alejandro Ramos-Soto, Alberto Bugarín, Senén Barro, and Juan Taboada. 2015. Linguistic descriptions for automatic generation of textual short-term weather forecasts on real prediction data. *IEEE TFS*, 23:44–57.

Clément Rebuffel, Laure Soulier, Geoffrey Scoutheeten, and Patrick Gallinari. 2020. Parenting via model-agnostic reinforcement learning to correct pathological behaviors in data-to-text generation.

Anna Rohrbach, Lisa Anne Hendricks, Kaylee Burns, Trevor Darrell, and Kate Saenko. 2018. Object hallucination in image captioning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4035–4045, Brussels, Belgium. Association for Computational Linguistics.

Svetlana Sheremetyeva. 2014. Automatic Text Simplification For Handling Intellectual Property (The Case of Multiple Patent Claims).

Daniel J. Solove. 2002. Access and Aggregation: Privacy, Public Records, and the Constitution. *Minnesota Law Review*, 86.

Wilson L. Taylor. 1953. “Cloze Procedure”: A New Tool for Measuring Readability. *JMCQ*, 30:415 – 433.

Sandrine Prom Tep, Florence Millerand, Alexandra Parada, Alexandra Bahary, Pierre Noreau, and Anne-Marie Santorineos. 2019. Legal Information in Digital Form: the Challenge of Accessing Computerized Court Records. *IJR*, 8.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. *ArXiv*, 1706.03762.

Nicolas Vermeys. 2016. Privacy v. Transparency: How Remote Access to Court Records Forces Us to Re-examine Our Fundamental Values”,. *Karim Benyekhlef and al. (éd.), eAccess to Justice, Ottawa, University of Ottawa Press*.

Daniel M. Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B. Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-Tuning Language Models from Human Preferences. *ArXiv*, 1909.08593.

Case Law References

A.T. v Globe24h.com. 2017. FC 114.

CBC. v Canada (A.G). 2011. SCC 2.

CBC. v New Brunswick (A.G). [1996]. 3 SCR 480.

Edmonton Journal v Alberta (A.G). [1989]. 2 SCR 1326.

Gichuru v Purewal and another. 2017. BCHRT 19.

Sierra Club of Canada v Canada (Minister of Finance). 2002. SCC 41.

US Department of Justice v. Reporters Committee for Freedom of the Press. 1989. 489 U.S. 749.

Vancouver Sun (Re). 2004. SCC 43.