University of Amsterdam

VerbCL: A Dataset of Verbatim Quotes for Highlight Extraction in Case Law

VerbCL: A Dataset of Verbatim Quotes for Highlight Extraction in Case Law

Julien Rossi
j.rossi@uva.nl
University of Amsterdam
Netherlands

Svitlana Vakulenko
s.vakulenko@uva.nl
University of Amsterdam
Netherlands

Evangelos Kanoulas
e.kanoulas@uva.nl
University of Amsterdam
Netherlands

ABSTRACT

Citing legal opinions is a key part of legal argumentation, an expert task that requires retrieval, extraction and summarization of information from court decisions. The identification of legally salient parts in an opinion for the purpose of citation may be seen as a domain-specific formulation of a highlight extraction or passage retrieval task. As similar tasks in other domains such as web search show significant attention and improvement, progress in the legal domain is hindered by the lack of resources for training and evaluation. This paper presents a new dataset that consists of the citation graph of court opinions, which cite previously published court opinions in support of their arguments. In particular, we focus on the verbatim quotes, i.e., where the text of the original opinion is directly reused. With this approach, we explain the relative importance of different text spans of a court opinion by showcasing their usage in citations, and measuring their contribution to the relations between opinions in the citation graph. We release VerbCL[1], a large-scale dataset derived from CourtListener and introduce the task of highlight extraction as a single-document summarization task based on the citation graph establishing the first baseline results for this task on the VerbCL dataset.

CCS CONCEPTS

• **Information systems → Specialized information retrieval**; *Content analysis and feature selection*; *Summarization*; **Expert search**; *Information extraction*; *Question answering*; • **Computing methodologies** → *Information extraction*.

KEYWORDS

case law, information retrieval, summarization

---

[1]https://github.com/j-rossi-nl/verbcl

1 INTRODUCTION

Courts of common law systems rely on statutes and precedents for deciding the law applicable to a case. A court opinion on a case, being a decision taken by a court when litigating a specific case, is a text assembled by the court judges using the parties' arguments and rebuttals of arguments. Previously published opinions (also known as case law) are cited to support the argumentation of the parties and the opinion of the court on the litigated case.

Thereby a legal discourse is made of a mix of factual details from the case as well as legal discussions arising from the qualification of the facts, driven by the past discussions with a similarity, either in facts or in reasoning. We provide an example, taken from the VerbCL dataset that illustrates how case law citation is used in practice, in Table 1.

For any legal-domain practitioner, search for an appropriate citation is a key task in preparing the argumentation for a case, which needs to be facilitated by an information retrieval system. This task was considered in case law retrieval [18, 33] as an information retrieval task where the query is a draft of an opinion and the result is a ranked list of past opinions. Relevance in this case may be guided by the similarity between the query and the retrieved opinion, or by the fact that the retrieved opinions address legal facets or questions in the query, as well as by the consideration whether the retrieved opinions might be worth being cited or not in the context of the query.

In this work, we introduce the new task of *highlight extraction*: given the text of an opinion, predict the subset of text spans that are likely to be cited in the future. In this task, the question is whether we can successfully predict which parts of an opinion will have an impact on future litigation, i.e., will be cited.

The motivation behind this task is to facilitate the work done by a legal commentator, who is tasked to identify interesting points in an opinion at the time of its publication. The legal commentator inspects all recently published opinions looking for potentially valuable excerpts that are worth being cited. The evaluation whether a part of an opinion is of a particular interest, or might be of interest in future cases is informed by the knowledge of the jurisprudential landscape, so the commentator can contrast in a new opinion what makes part of this landscape, and what is novel.

We aim to inform the task of highlight extraction through the construction of a large-scale dataset that will allow the development and evaluation of data-driven models able to solve the task. To this end, we use a public dataset of US court opinions, CourtListener[2], to extract a citation network of court opinions.

The citation network naturally emerges from the repository of opinion documents since each opinion usually quotes multiple

---

[2]https://www.courtlistener.com

Citing opinion	**ID**: 2346822
	Title: LL Ex Rel. Doe v. Chimes Dist. of Columbia, Inc
	Text:
	(...) a special relationship that gave rise to a duty on the part of the United States to protect or warn her.[1] There was no such special relationship between L.L. and the United States, however, and the United States motion must accordingly be granted. "Ordinarily, the owner or possessor of land is under no duty to protect invitees from assaults by third parties while the invitee is upon the premises ... [unless] there is a special relationship between [the] possessor of land and his invitee giving rise to a duty to protect the invitee from such assaults." Wright v. Webb, CITATION_1239944 , 920-21 (1987). Chimes submits that this is a "special relationship" case. The cases on which Chimes relies for that proposition presented distinct factual bases for finding a "special relationship." (...)
Cited opinion (score = 1.71)	**ID**: 1239944
	Title: Wright v. Webb
	Text:
	(...) We will assume, without deciding, that Webb was the Wrights' business invitee. Thus, the Wrights owed Webb the duty of ordinary care to maintain their parking lot in a reasonably safe condition. See Tate Rice, 227 Va. 341, 345, 315 S.E.2d 385, 388 (1984). Ordinarily, the owner or possessor of land is under no duty to protect invitees from assaults by third parties while the invitee is upon the premises. Restatement (Second) of Torts	314A (1965) recognizes exceptions to the rule of non-liability for the assaults of a third party where there is a special relationship between a possessor of land and his invitee giving rise to a duty to protect the invitee from such assaults. We alluded to this Restatement rule in both Klingbeil Management Group Co. Vito, 233 Va. 445, 447, 357 S.E.2d 200, 201 (1987) and Gulf Reston, Inc. Rogers, 215 Va. 155, 158, 207 S.E.2d 841, 844 (1974), but made it plain in Gulf Reston that this was only a reference to the Restatement rule. Our statement in Klingbeil was simply a comment upon the reference in Gulf Reston. (...)

Table 1: A sample of a verbatim quote from the VerbCL dataset.

previously published opinions for the purpose of the legal argumentation of the case. We consider this network as a directed acyclic graph, where each node stands for an opinion document. Edges represent the citations made in opinions, directed from the citing opinion's node towards the cited opinion's node. In this paper, we consider the citing-cited relation as follows: a citing opinion is making an argument by referring to a cited opinion.

In the text of the citing opinion, a citation is introduced by an *anchor*, that we consider to be the supported legal argument. More specifically, we differentiate between two types of anchors: abstractive anchors, where the argument is cited as a paraphrase of the original opinion document, and extractive anchors (or *verbatim quotes*) where the argument is directly extracted from the cited opinion's text by copying the relevant text span. We focus on the verbatim quotes in this work since they are much easier to detect and reproduce than paraphrases.

We further consider that verbatim quotes emphasize which part of the cited opinion have a legal weight, bringing novelty and importance into the legal landscape at the time of the publication of an opinion document. We define a *highlight* of an opinion as the set of all text spans that are later cited as verbatim quotes. Therefore, we can produce the highlights for the previously published opinions by considering the citation graph. The highlight of an opinion can be constructed by considering all the verbatim quotes of this opinion used in the citing opinion documents (see an example of the highlight in Table 1).

To aggregate information about all citing opinions, the citation graph is used. We illustrate the citation graph around the cited opinion from Table 1 in Figure 1, which shows all the citing opinions as well as the identifiers of their verbatim quotes as the edge labels.

The goal of highlight extraction is to predict the highlights for new opinions, i.e., predicting the future citations based on the opinion text. We proceed by establishing the first baseline for the highlight extraction task using the state-of-the-art single-document summarization approaches: TextRank [1] and PreSumm [27], as well as a sentence-level classifier based on DistilBERT [36] . The goal of our experiments is to verify to which extent the highlight extraction task in the citation network can be addressed using single-document summarization approaches without any information about the citation graph structure.

Our main contributions are two-fold:

(1) the dataset with a citation graph, verbatim quotes and opinion highlights;
(2) the task of highlight extraction with baselines.

We start by describing how VerbCL was constructed in Section 2 and discuss its main characteristics in Section 3. We formalize the task of highlight extraction and report on the experiments we conducted with the baseline models for this task in Section 4. The overview of existing datasets for case law retrieval, and related work on highlight extraction and citation-based summarization in other domains is given in Section 5. We conclude and propose directions for future work in Section 6.

2 DATASET CONSTRUCTION
VerbCL is based on the Court Listener dataset, a collection, which is publicly available for free. In this section, we describe the process of constructing the dataset.

Field Name	Type	Comment
citing_opinion_id	int	The opinion_id of the **CITING** opinion.
cited_opinion_id	int	The opinion_id of the **CITED** opinion.
sentence_id	int	The sentence_id of the source sentence in the CITED opinion.
verbatim	str	The span of text in the CITING opinion that we consider to be a potential verbatim quote from the CITED opinion
snippet	str	The span of text from the CITING opinion around the in-text citation of the CITED opinion. It contains 100 words before the in-text citation, and 100 words after the in-text citation.
score	float	The score of verbatim for being an actual verbatim quote from the CITED opinion, placed in the CITING opinion. A score of −1 indicates our model does not consider it as being a quote from the CITED opinion.

Table 2: Fields of the documents in VerbCL Citation Graph.

Field Name	Type	Comment
opinion_id	int	The unique identifier of the opinion.
sentence_id	int	The unique identifier of the sentence: its index within the list of sentences of the opinion.
raw_text	str	The complete text of the sentence.
highlight	bool	The binary label of the sentence. *True* means the sentence has been quoted verbatim in a citing opinion.
count_citations	int	The number of times the sentence has been quoted verbatim.

Table 3: Fields of the documents in VerbCL Highlights.
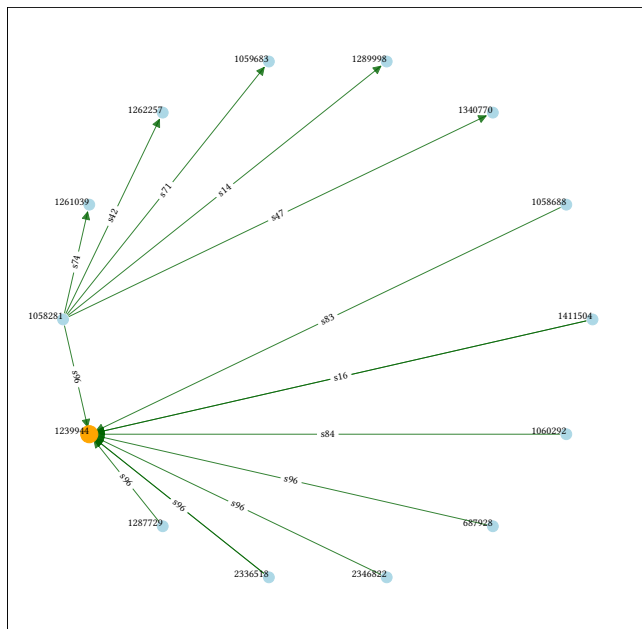
Figure 1: Citation graph, where each node is an opinion document with edges pointing from a citing opinion to a cited opinion. Every edge is labeled with the identifier of a sentence of the cited opinion that was quoted verbatim. For example, sentence 96 of opinion 1239944 is cited by the opinion 1058281, which contains sentence 44 cited by opinion 1058321.

2.1 Court Listener

CourtListener is a project of the Free Law Project[3], a US non-profit organization started in 2010, with the aim of "making the legal world more fair and efficient".

The original Court Listener dataset is a collection of every court opinion published by every court in the United States. It covers 406 jurisdictions (out of 423), with opinions from the year 1754 up to now. It is constantly updated with newly filed opinions, and digitized archives.

We obtained the Court Listener dataset by downloading the bulk data on September 1st 2019.[4] More recent versions will have more opinions available, the whole processing will apply as long as the underlying relational database scheme remains unchanged.

The unit of data in Court Listener is the court opinion. Each court opinion is a single unique decision made by a specific court at a specific date on a specific case. A single case can be litigated by many courts, in case of appeals for example, so opinions are clustered per case. The Court Listener dataset follows the scheme of a relational database, where each opinion belongs to one cluster. Opinions are also clustered in dockets, a single docket gathering the different opinions made by a unique court on a unique case, as it happens that the same court may emit multiple opinions on the case at different dates.

At a more granular level, Court Listener represents each opinion as an HTML document, where specific tags mark where an in-text citation of case law appears in the opinion. Free Law implements a citation parsing and matching program[5] that allows for an automated annotation of the unstructured text of the original court opinion.

2.2 Data Processing

For the construction of VerbCL, we focus on the opinions of CourtListener whose full text is available as HTML code including tagged in-text citations (looking for the field `html_with_citations`). The HTML code for each opinion has specific tags to identify citations made in the text of the opinion, each citation being identified with the unique identifier of the cited opinion.

We consider that each citation is introduced by an anchor, which is located in the vicinity of the citation tag. The anchor is the "raison d'être" of the citation, it is an argument to the current case, it is participating to the litigation of the case, and its validity is affirmed by invoking a similar argument that has already been made in an existing court opinion.

We focus specifically on the anchors that are extracts of the text of the cited opinion (*verbatim quotes*), similar to the example provided in Table 1.

Algorithm 1: Data Processing Pipeline

Result: Verbatim Quotes
initialization;
forall *opinions* **do**
    forall *citations* **do**
        snippet = N words before / after in-text citation;
        candidates = potential verbatim quotes;
        forall *candidates* **do**
            if *candidate is verbatim* **then**
                identify original sentences in cited opinion;
                store data point;

Our data processing procedure includes extracting the text around the in-text citation, identifying potential verbatim quotes (often marked by quotation marks) and eventually classifying those who actually are verbatim quotes from the text of the cited opinion. Our code includes a framework for parallel distributed processing of massive dataset, using pyarrow, Elasticsearch[6] through its Python API, and MongoDB[7] together with pymongo.

2.2.1 Snippets extraction. Although the anchor is in the vicinity of the in-text citation, we observe a wide range of variations in the presentation and wording of these anchors, in relation to the in-text citation. Our strategy is to reduce this problem to identify a verbatim quote in the direct surroundings of the in-text citation, defined as the N words before and after the citation. We opted for a value $N = 100$, which is a compromise between high recall (the more text around the citation we consider, the more likely it is we will identify the anchor), and computational footprint. Experiments included: considering entire paragraphs, but it yielded too much text; identifying nearby sentences, but the high numbers of acronyms and abbreviations around in-text citations renders sentence splitting useless.

2.2.2 Verbatim candidates. The identification of potential verbatim quotes is entirely rule-based, around the usage of different quote

---

characters. Each snippet will generate multiple spans that were enclosed between those quote characters.

2.2.3 Qualifying verbatim quotes. At this stage, we want to confirm for each verbatim candidate whether it is a text that originates from the cited opinion. In the context of studying a specific in-text citation in a citing opinion, the cited opinion is known, so we have to evaluate whether a text span is lifted from the full text of the cited opinion.

The problem of identifying segments of text within a long text has proven to be difficult. As most of the opinions originate from printed books, OCR artifacts are expected, such as misplaced spaces or misspellings. This will affect the quality of the text in both cited and citing opinion. The common practice of using ellipsis in the citing opinion's text (as can be seen in the sample shown in Table 1) will result in the suppression of fragments of the original text.

Traditional approaches to this fuzzy matching problem rely on heuristics based on the computation of editing distances, such as the Levenshtein Distance [21]. The design of the heuristics is a first challenge, given the sheer variety of differences that could be observed between the original text in the cited opinion and its counterpart in the citing opinion. More importantly, the computational footprint of calculating an editing distance renders this method inapplicable to our dataset.

We built a classifier based on "Interval Queries" offered by Elasticsearch. We refer the reader to the provided source code for the exact parameters of this query. The classifier predicts that a span of text belongs to the positive class when the query returns a non-empty result, and predicts the negative class when this query does not produce any search result. We manually annotated a test dataset for this classifier which revealed it had high precision and recall for both classes. The reader can refer to Chapter 2.3.

As a result of this stage, we have built the VerbCL citation graph, a directed graph where nodes are opinions and edges are verbatim quotes.

2.2.4 Identifying highlights. Highlights in the cited opinion are the sentences that were used for verbatim quotes. For each span of text identified as a verbatim quote of the cited opinion, we identify which sentence or sentences, are actually quoted from the text of the cited opinion. This is achieved through another set of Elasticsearch queries that are run over an indexed collection of opinion sentences. For sentence tokenization, we used the Punkt sentence tokenizer from NLTK [2].

As a result, we have built the VerbCL Highlights dataset, a dataset of opinions annotated at sentence level with a binary label where the positive class is made of sentences that were later on cited as verbatim quotes.

2.2.5 Dataset structure. Following this stage, we have produced the following data collections:

- **VerbCL Highlights**: An annotated collection of all opinion sentences, with a binary label for which True indicates that the sentence was cited verbatim;
- **VerbCL Citation Graph**: An annotated citation graph for all case law citations with a verbatim quote of the cited opinion. It is a directed acyclic graph (V, E) where V is the

set of all opinion documents and E is the set of verbatim quotes.

VerbCL Highlights is used to inform the highlight extraction task since it contains the original opinion text and the subset we consider as the correct highlight. VerbCL Citation Graph was used to produce VerbCL Highlights and provides auxiliary information about the opinion text reuse that can be used for the highlight extraction task.

See Tables 2-3 for the structure of these subsets. A tutorial on loading the data is available as a `Jupyter Notebook` in the code repository[8].

2.3 Quality Assurance

We manually annotated 180 random anchors, coming from random opinions, and evaluated whether or not the candidate snippet was an actual verbatim quote out of the cited opinion. The code for this manual annotation task is included in our codebase, it is based on the `Django` framework.[9]

On this random sample of 180 texts, with 60 of them from the positive class (actual verbatim quotes from the cited opinion), our binary classifier had a Precision $P > 0.96$ and Recall $R > 0.92$ for both classes. We consider our solution for this problem to offer a sufficiently good compromise between computation and accuracy.

3 DATASET STATISTICS

In the following we describe the main characteristics of the VerbCL Citation Graph and the VerbCL Highlights.

3.1 VerbCL Citation Graph

From Court Listener, we consider the subset of opinions that either cite other opinions with a verbatim quote, or are cited verbatim by other opinions. This subset contains circa 1.5M opinions.

The main characteristics of our dataset are summarized in Table 4. Considering that we are studying the nodes of a citation network, we also disclose our analysis of the corresponding graph.

In the citation network, nodes are opinions and directed edges materialize the citations, from the citing opinion towards the cited opinion. Thereby, the in-degree of a node v is the number of times this opinion is cited by other opinions, while the out-degree indicates the number of opinions that the opinion corresponding to the node v cites.

We plot the distribution of in-degree of the nodes in the citation network in Figure 2, as well as the Zipf's law, a power law with an exponent $k = -1$, which is a common distribution in the field of text mining. This shows us that these verbatim quotes have a behavior closer to the appearance of a term in a collection of documents, than to the expected distribution of links in a web graph (a power law with exponent $k \approx 2.1$).

We imported the citation network, using `networkx` [12], in order to compute some descriptive centrality statistics of the graph. For computing time reasons, we used approximations instead of the actual values, following the experiments in [4].

Court Listener (CL)

Opinions	4,265,231
Citing opinions	3,062,334
Cited opinions	2,020,779
Citations	30,318,321

VerbCL Citation Graph

Opinions	1,493,561
	35% of opinions in CL
Verbatim quotes	6,210,703
	20% of citations in CL
Verbatim citing opinions	1,086,238
	35% of citing opinions in CL
Verbatim cited opinions	946,962
	47% of cited opinions in CL
Edges in the citation graph	4,002,137
Graph density	1.8×10^{-6}
Number of words in a verbatim	$[5 - 100]$
Average	15
Quartiles 25-50-75	12 - 20 - 30

Table 4: Dataset statistics of Court Listener (CL) and VerbCL.
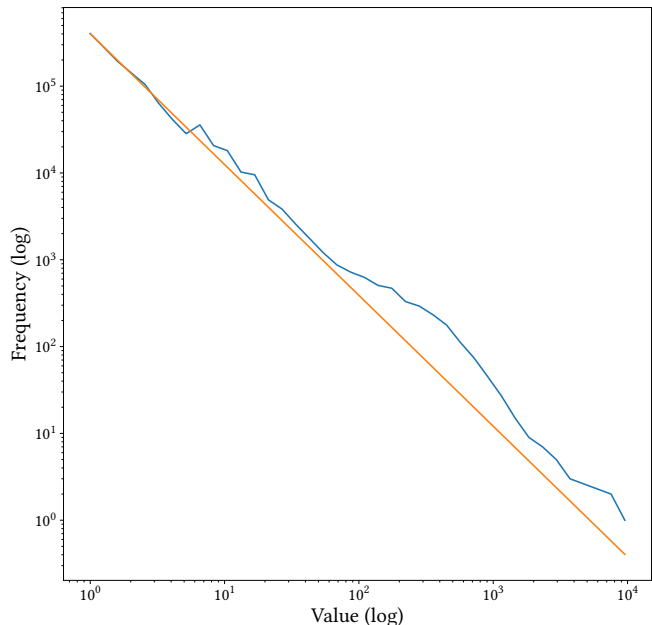
Figure 2: Distribution of number of citations for an opinion.

The distribution of betweenness centrality of the nodes of the citation graph is plotted in Figure 3, where we observe that the network contains a few very central nodes, corresponding to the group with the highest values. A node with a high betweenness centrality is a node which is on the shortest path between many nodes of the graph, reflecting its importance in the network. Betweenness centrality [3] of a node v is defined as the sum of the fraction of all-pairs shortest paths that pass through v, see Equation 1. Since
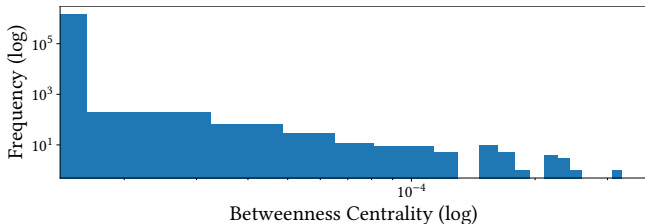
Figure 3: Distribution of betweenness centrality.

the exact calculation of betweenness centrality for every node of the network is too computationally expansive, we made use of an approximation algorithm [4].

$$c_B(v) = \sum_{s,t \in V} \frac{\sigma(s,t|v)}{\sigma(s,t)} \qquad (1)$$

When applied to academic literature and the corresponding citation network, betweenness centrality is considered to measure importance of a paper in the field, i.e., seminal academic papers were reported to have a relatively high betweenness centrality [22, 31]. In contrast, our citation network has only a few nodes with high relative betweenness centrality but the absolute values are still very low (maximum: 3×10^{-4}).

We hypothesize that this difference can be explained by much higher content redundancy in the court opinions in comparison with academic literature. Novelty and originality of contributions are the main characteristics for an academic paper, which makes the paper take a unique and irreplaceable place in the network. In contrast, courts of the justice system serve any valid case presented regardless of its originality. Therefore, it is only natural that multiple cases can often make for equally good candidates in support of the same argumentation line. This has an effect of the more even citation distribution across the whole network of court opinions in comparison with an academic network, which also reduces the centrality measures for court opinions as our data confirms.

Comparing the ranking of nodes by centrality with the ranking by outdegree (i.e. the number of times this opinion is cited by others), we compute the Kendall's Tau and Spearman's Rho coefficients of rank correlation: $\tau = 0.21$, $\rho = 0.27$, both with p-value $p = 0.0$, showing weak but significant correlation. We attribute this weak correlation to the historicity factor: contrary to a webpages network, each node in our citation network is constrained by the capacity to cite only past opinions; we also consider the novelty factory, where a legal practitioner is biased towards most recent opinions when citing, therefore pushing the "central" opinions out of the shortest paths.

3.2 VerbCL Highlights

The main characteristics of our dataset are summarized in Table 5. These statistics come from a random sample of circa 10k opinions.

Sentences from one opinion that are cited verbatim by other opinions are considered as highlights of the cited opinion. The task of identifying the highlights from the full text of an opinion is introduced in this paper under the name "Highlight Extraction", in Section 4.

VerbCL Highlights	
Opinions	1,493,561
Number of sentences per opinion	99.5% under 5000 sentences
Average	437
Number of tokens per sentence	99.9% under 171 tokens
Average	24
% highlight sentences (per opinion, the number of highlights divided by the number of sentences)	99.5% under 21.8%
Average	3.5%
Number of citations per highlight	99% under 434 citations
Average	34

Table 5: Dataset statistics for VerbCL Highlights.

Court opinions are typically large documents, we can illustrate this by showing the distribution of the number of words per opinion in Figure 4.
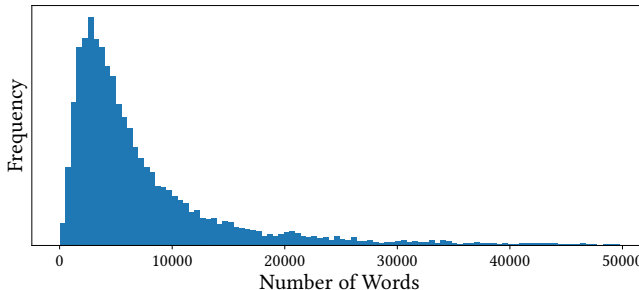
Figure 4: Number of words per opinion.

4 HIGHLIGHT EXTRACTION

In this section, we will formally define the task of highlight extraction, then describe the empirical experiments we conducted and their results.

4.1 Task Description

We formally define the task of highlight extraction as follows. Given a set of opinions V, where each opinion $v \in V$ is associated with an opinion text T_v split into a set S_v of n_v sentences. The task of highlight extraction is to predict a subset of sentences $\widehat{S_v} \subseteq S_v$ which will be cited in the future. The input of this system is the text of a single opinion. It is implied that the system's parameters are learned from the complete collection of opinions, and the system's training should not be restricted to document level tasks. Thereby, the task of identifying highlights, when formulated at sentence level can be considered as a sentence binary classification (ranking) problem.

In this work, we define this task as a sentence highlighting task. The text spans we will consider are sentences from the opinion, we want to build model that predicts which sentence(s) in a given opinion might be cited, considering the historical observations as a ground truth that can be used for training.

This task involves the detection of network and timeline effects, as novelty arises from being the first to make a certain argument, or the first to legally qualify some facts as being compliant or not with generic statutes. As such we do expect it to be solvable at document level, at text level, if and only if we can assume that the opinion drafters are aware of the novelty of elements in the opinion, and draft them textually in a way that the task can be reduced to a Language Modeling task.

4.2 Evaluation Metrics

In line with the formulation of highlight extraction as a sentence ranking task, we will evaluate the different models with ranking metrics, such as Precision at 1 (P@1), Precision at R (P@R), Mean Average Precision (MAP) and Mean Reciprocal Rank (MRR).

In the context of supporting the work of a legal practitioner, we value the presence of relevant sentences early in the ranked list of sentences. P@1 measures the capacity of placing a highlight sentence at the top of the ranked list, P@R is a precision metric suitable for queries with a diverse number of relevant answers, MAP balances equally important metrics precision and recall and MRR is an indicator of the effort of the system's user, in relation to the rank of the first relevant sentence.

We also consider ROUGE [25], a family of metrics for summary evaluation. For each opinion, from the list of ranked sentences, we consider the hypothesis summary to be made of the top 5 sentences. The reference summary of an opinion is made of the highlight sentences in the opinion. ROUGE will score favorably a ranker which identified sentences lexically similar to the reference sentences.

4.3 Data

We use a sample out of VerbCL Highlights as training material. We make use of a random sample of 88k opinions, from the pool of opinions that have at least one sentence marked as highlight. The data is split between training material and test material. The train collection contains circa 70k opinions.

As we have to respect the sequence length limitations of the deep neural models, we restricted the test collection to the opinions for which there is at least one highlighted sentence in the first $N = 512$ words of the opinion, and we consider the original text as made of these N first words. This test collection contains circa 1,000 opinions. Since the data is split based on opinions, the data in the test set is unseen for a model trained on the train set.

As we focus on identifying highlights, we make use of documents for which we observe that they contain highlights. We leave apart in this work the task of deciding whether or not an opinion contains highlight. From the perspective of the highlight extraction task, we consider having an oracle that has already stated the existence of highlights in the dataset we use.

4.4 Baselines

Considering the highlight task as a sentence ranking task, we propose two different types of baseline using

- binary classification at a sentence level;
- extractive summarization at a document level.

While it makes sense to consider the binary classification task, we argue that this approach will fail in this case, as most of the signals

that a sentence in a legal opinion might be of interest for citation lie outside of the sentence itself. We produce this experiments to verify our assumptions that the sentence itself does not contain the signal of its importance.

An extractive summarization model will have to take the whole document into account in order to rank the sentences. On the one hand, deterministic methods will focus on one aspect for the importance of a sentence within a document. On the other hand, trainable models can be fine-tuned towards golden summaries, based on sentence-level annotations. We will use both types of models with our annotated data from VerbCL Highlights, and observe whether our annotations for importance can be learned this way. The assumption we want to confirm is that the document contains more information about the important sentence than the sentence itself, but still not enough information to lead to an accurate extraction.

We select the following models as our baselines:

- **DistilBERT** [36]: a generalized language model that we fine-tune for the binary classification task at sentence level. We use the `transformers` library from HuggingFace [10]. We derive a sentence ranker from this model by considering the estimated probability of belonging to the positive class as the sentence score.
- **TextRank** [1]: an extractive summarization deterministic algorithm. Although it can not be considered as state of the art, TextRank has proven to be a reliable baseline for summarization with a reasonable computation footprint. Considering the task of highlight detection as an extractive summarization task, we evaluate how well the standard summarization approach that ranks sentences by their importance is able to capture what is considered salient from the perspective of citing an opinion. Using the Python implementation of `PyTextRank[30]`, we retrieve the score of each sentence.
- **PreSumm** [27]: an extractive summarization model, based on Bert [10]. PreSumm has the capacity to be fine-tuned on any dataset annotated at sentence level. PreSumm implementation [11] has been adapted to our dataset, and we retrieve the score of each sentence after inference. The model was trained for 50k steps, which took approximatively 14 hours with 4 nVidia GPUs.

4.5 Results

We present our results in Table 6. They confirm the initial assumptions with regard to the difficulty of the task. We formulate the highlight extraction as a task where the relevant context is the context of the current jurisprudence, therefore we make the assumption that this task can only be solved at corpus level, and not at document level. We observe from our experiences that using summarizers improves on the performance, but only to a certain degree.

The baselines we selected are introduced in their order of complexity. The sentence ranking we derive from a binary relevance classifier based on DistilBERT performs similarly to a random ranker. This validates our assumption that highlights can not be discovered at sentence level only.

---

[10]https://huggingface.co/transformers/
[11]https://github.com/nlpyang/PreSumm

Model	P@1	MAP	P@R	MRR	Rouge1-F	Rouge2-F	RougeL-F
DistilBERT	0.05	0.11	0.05	0.12	0.31	0.20	0.26
TextRank	0.14	0.23	0.14	0.24	0.41	0.36	0.38
PreSumm	0.31	0.37	0.30	0.38	0.52	0.48	0.48

Table 6: Results for the Highlight Extraction Baselines

On the other hand, the improvements shown with the summarizing models emphasizes that some of the highlights can be captured at the document level. We hypothesize that this is linked to the opinion drafter's awareness of the importance of the sentence, and its potential to be cited later. An opinion drafter is a person skilled in the legal domain, they are in position to have this intuition. They would reflect the importance of the sentence at the text level, and therefore be detected by models from the BERT lineage, who excel at isolating signals from the text [9, 26, 41].

The P@1 = 0.31 observed on PreSumm is a promising result, but we also have to consider that it is restricted to short documents, while a majority of documents are longer than the maximum input length. Despite the observed improvements, we conclude that none of these systems has solved the task sufficiently.

5   RELATED WORK

In this section, we present an overview of existing work in relation with the dataset and the tasks we want to support, from within the legal domain as well as outside of this legal domain. We provide a brief overview of the previously proposed datasets, and describe the recent advancements on the tasks of highlight extraction, citation-based summarization and case law retrieval.

It is important to note that similar tasks often arise also in patent retrieval [15, 34, 37, 39, 40] and academic document retrieval domains [23, 24, 38, 44].

When drawing parallels to the tasks outside of the legal domain, the following dimensions are of a special importance:

- emergence of a directed citation network;
- time-based dependencies (it is possible to cite only previously published sources);
- highlight extraction for document summarization;
- predicting citations and text reuse.

We observe recent work in the field of citation analysis for scientific literature, enabled by S2ORC [28], which we consider a similar dataset to VerbCL Citation Graph and VerbCL Highlights in content and intent. Applied to academic literature, rating novelty [16], identifying contributions [14] or summarizing [5] share similarities with the highlight extraction task. We depart from the work done on academic literature, as we argue that the novelty is the core motivation of academic publication, while court opinions are motivated by the obligations of a public service which has to serve justice whenever it is requested. In that matter, novelty is expected to be sparser, while abundant redundancy should be the norm.

5.1   Case Law Datasets
CourtListener was also the original source of CaseLaw [29], for the purpose of testing case law retrieval models and systems, while the

main contribution was a test collection of 2,500 relevance assessments and an annotation tool. We observe similarly sized datasets in other languages, such as CAIL2019-SCM [43] in Chinese. Recent meta-reviews of the field [45] provide pointers to existing datasets[12], we observe that none of them has the size required for training a complex model on a specific task.

COLIEE [18, 33] is also a relevant competition, task and dataset for legal purposes, of reduced size. It is made of multiple tasks, tackling case law retrieval with tasks 1 and 2, within the legal domain of Immigration and Citizenship in canadian courts.

Task 1 describes case law retrieval as the task of identifying which past cases should be mentioned in a query case from which citations have been removed. The rationale behind the relevance judgment, the reason why cases have been cited in the past is left unknown. Task 2 will focus on identifying relevancy at paragraph level, which is a step forward to clarifying relevance for the human reader, although the dataset does not include annotations with regard to relevance arising from either situation and facts or reasoning.

5.2   Highlight Extraction
Given a long document, the task of highlight extraction aims at finding the snippets of text that contain the information that a user will find useful in this document, making it a task ripe for user query biased extraction. For example, [19], [7] and [32] show the importance of highlighting for learning. Highlighting is described as a manual annotation task, for the benefit of meeting future information needs. The concept is put in practice also by [13] and [8], where highlights support the generation or the evaluation of text summaries. We observe only few attempts at creating an automated highlighting system for text, in contrast to the flourishing field of video highlight generation.

The key concept we articulate our task around is meeting future information needs. Although we have plenty of historical data available, we can only speculate now on what is important in any opinion that is published, and what will be picked up in cases that courts will litigate in the near or far future.

The highlight extraction task we present in this work focus on identifying the novel legal content in a new case that would potentially be cited in future cases, and evaluate how this could stem from text level analysis of the opinion, rather than identifying how an opinion would suit an argument made in another one. We refer to the Section 4 for more details.

---

[12]https://github.com/thunlp/LegalPapers

5.3 Case Law Retrieval

Our work is also related to the previous work on the task of case law retrieval. In this task, a legal practitioner retrieves a list of past cases that are legally relevant to a case at hand.

COLIEE [18, 33] has hosted a yearly competition including such a retrieval task, restricted to cases from the Immigration and Citizenship Courts of Canada. The target is to identify cases that could be cited in a query case.

The task is reduced to identifying the similarities between court opinions, similarities in details of the case, in the underlying storylines and reasoning. Participants reformulated the retrieval task as a ranking task based on a pairwise relevancy classifier, with various text representation techniques applied, from lexical representations [20, 42] to deep representations [11, 35]. It is observed that the scarcity of data is hindering the use of advanced data-driven models. This domain relevant task has shown the difficulty to work at text level to evaluate a relevancy that professionals had annotated.

We see a potential application of our dataset in a case law retrieval task guided by arguments, where the query is an argument, and the search results a list of past opinions that could support the argument, based on the highlights that were extracted.

5.4 Citation-based Summarization

The task of highlight extraction finds its roots in the task of summarizing long documents. We observed many efforts in the field of summarizing by using citations, in the domain of academic literature summarization, for example. Citation-based summarization was first introduced in the context of scholarly data processing for summarizing scientific articles [17]. In citation-based summarization, citation anchors (called "catchphrases") for one target document over multiple citing document form a summary for this target document. The CL-SciSumm shared task [6] aims at producing summaries of academic papers guided by citation anchors (called "citances"), by identifying spans of text relevant to the anchor. The approaches are evaluated against a set of golden summaries, using standard metrics of the ROUGE family [25].

The court opinions will show a predetermined structure, as academic papers do, although the articulation in different parts or chapters will not be signaled explicitly in the text. Our task definition differs as we focus on actual verbatim quotes from cited documents which are less biased towards the writer's view of the cited document than paraphrases. Our work considers these verbatim quotes to be the target summary, as opposed to considering them as a way to build a summary similar to the golden summary.

We consider that the spans of an opinion which are later used in citations form a summary of this opinion, which a specific type of summary guided by the search for novel and important legal aspects addressed in the opinion document. This summary characterizes what the change an opinion makes in terms of the legal argumentation, i.e., its network effect (global context), not just a summary of all the elements present in the opinion (local context).

6 CONCLUSION

In this paper we introduced the task of highlight extraction from court opinions and the large dataset VerbCL of annotated court opinions aimed at supporting training and evaluation for the highlight extraction task. Our dataset focuses on citations made in a court opinions by quoting verbatim preceding court opinions in support of a legal argument. VerbCL is sufficiently large and can be used for training machine learning models on the highlight extraction task. We also see the potential of VerbCL for other legal information retrieval tasks that can be informed by the citation network.

We also demonstrated the difficulty of the highlight extraction task, which escapes the reduction to a sentence-level or document-level task. Note that the citation network is the result of a massively distributed expert work from the date of the first citation until now. Every citation is the result of an expert's retrieval of a relevant opinion to support the argument in the context of a specific case at hand. However, we can verify the importance of every specific text span from an opinion only post-hoc, i.e., not at the time of the publication but later on as we observe how the same arguments made in the opinion are subsequently re-used within other opinion documents in our dataset. Only this reuse over time indicates the document's importance in terms of the impact it made to help shape other documents. Future work should aim to address such network and historical effects, distilling knowledge of the previously published documents into the processing of a new one. We believe that similar approaches will also prove useful for patent retrieval and academic search scenarios.

Future work should also consider extending the analysis to abstractive anchors, which requires a text generation task setup instead of purely retrieval-based one considered here. Finally, considering the full document length remains a challenge for the current neural ranking approaches that should be addressed to be applicable in the legal domain.

REFERENCES

[1] Federico Barrios, Federico López, Luis Argerich, and Rosa Wachenchauzer. 2016. Variations of the Similarity Function of TextRank for Automated Summarization. arXiv:1602.03606 [cs.CL]

[2] Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.".

[3] Ulrik Brandes. 2001. A faster algorithm for betweenness centrality. *Journal of mathematical sociology* 25, 2 (2001), 163–177.

[4] Ulrik Brandes and Christian Pich. 2007. Centrality estimation in large networks. *International Journal of Bifurcation and Chaos* 17, 07 (2007), 2303–2318.

[5] Isabel Cachola, Kyle Lo, Arman Cohan, and Daniel S Weld. 2020. Tldr: Extreme summarization of scientific documents. *arXiv preprint arXiv:2004.15011* (2020).

[6] Muthu Kumar Chandrasekaran, Michihiro Yasunaga, Dragomir Radev, Dayne Freitag, and Min-Yen Kan. 2019. Overview and Results: CL-SciSumm Shared Task 2019. *CEUR Workshop Proceedings* 2414 (jul 2019), 153–166. arXiv:1907.09854 http://arxiv.org/abs/1907.09854

[7] X. Che, H. Yang, and C. Meinel. 2018. Automatic Online Lecture Highlighting Based on Multimedia Analysis. *IEEE Transactions on Learning Technologies* 11, 1 (2018), 27–40. https://doi.org/10.1109/TLT.2017.2716372

[8] Jianpeng Cheng and Mirella Lapata. 2016. Neural Summarization by Extracting Sentences and Words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers).* Association for Computational Linguistics, Berlin, Germany, 484–494. https://doi.org/10.18653/v1/P16-1046

[9] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What Does BERT Look At? An Analysis of BERT's Attention. arXiv:1906.04341 [cs.CL]

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers).* Association for Computational Linguistics, Minneapolis, Minnesota, 4171–4186. https://doi.org/10.18653/v1/N19-1423

[11] Baban Gain, Dibyanayan Bandyopadhyay, Tanik Saikh, and Asif Ekbal. 2019. IITP in COLIEE@ICAIL 2019: Legal Information Retrieval using BM25 and BERT. (06 2019). https://doi.org/10.13140/RG.2.2.28887.32161

[12] Aric A. Hagberg, Daniel A. Schult, and Pieter J. Swart. 2008. Exploring Network Structure, Dynamics, and Function using NetworkX. In *Proceedings of the 7th Python in Science Conference*, Gaël Varoquaux, Travis Vaught, and Jarrod Millman (Eds.). Pasadena, CA USA, 11 – 15.

[13] Hardy Hardy, Shashi Narayan, and Andreas Vlachos. 2019. HighRES: Highlight-based Reference-less Evaluation of Summarization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics, Florence, Italy, 3381–3392. https://doi.org/10.18653/v1/P19-1330

[14] Hiroaki Hayashi, Wojciech Kryściński, Bryan McCann, Nazneen Rajani, and Caiming Xiong. 2020. What's New? Summarizing Contributions in Scientific Literature. arXiv:2011.03161 [cs.CL]

[15] Sebastian Hofstätter, Navid Rekabsaz, Mihai Lupu, Carsten Eickhoff, and Allan Hanbury. 2019. Enriching word embeddings for patent retrieval with global context. In *European Conference on Information Retrieval.* Springer, 810–818.

[16] Bolin Hua and YoungKug Shin. 2021. Extraction of Sentences Describing Originality from Conclusion in Academic Papers. (2021).

[17] Nouf Ibrahim Altmami and Mohamed El Bachir Menai. 2020. Automatic summarization of scientific articles: A survey. https://doi.org/10.1016/j.jksuci.2020.04.020

[18] Yoshinobu Kano, Miyoung Kim, M. Yoshioka, Yao Lu, J. Rabelo, Naoki Kiyota, R. Goebel, and K. Satoh. 2018. COLIEE-2018: Evaluation of the Competition on Legal Information Extraction and Entailment. In *JSAI-isAI Workshops.*

[19] David Y. J. Kim, Adam Winchell, Andrew E. Waters, Phillip J. Grimaldi, Richard G. Baraniuk, and Michael C. Mozer. 2020. Inferring student comprehension from highlighting patterns in digital textbooks: An exploration of an authentic learning platform. In *Proceedings of the Second International Workshop on Intelligent Textbooks.*

[20] Moemedi Lefoane, Tshepho Koboyatshwene, and Lakshmi Narasimhan. 2018. KNN clustering approach to legal precedence retrieval.

[21] V. I. Levenshtein. 1966. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Soviet Physics Doklady* 10 (Feb. 1966), 707.

[22] Loet Leydesdorff, Caroline S Wagner, and Lutz Bornmann. 2018. Betweenness and diversity in journal citation networks as measures of interdisciplinarity—A tribute to Eugene Garfield. *Scientometrics* 114, 2 (2018), 567–592.

[23] Xinyi Li and Maarten de Rijke. 2017. Do topic shift and query reformulation patterns correlate in academic search?. In *European Conference on Information*

[24] Xinyi Li, Bob JA Schijvenaars, and Maarten de Rijke. 2017. Investigating queries and search failures in academic search. *Information processing & management* 53, 3 (2017), 666–683.

[25] Chin-Yew Lin. 2004. *ROUGE: A Package for Automatic Evaluation of Summaries.* Technical Report. 74–81 pages. https://www.aclweb.org/anthology/W04-1013

[26] Yongjie Lin, Yi Chern Tan, and Robert Frank. 2019. Open Sesame: Getting inside BERT's Linguistic Knowledge. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP.* Association for Computational Linguistics, Florence, Italy, 241–253. https://doi.org/10.18653/v1/W19-4825

[27] Yang Liu and Mirella Lapata. 2019. Text Summarization with Pretrained Encoders. *CoRR* abs/1908.08345 (2019). arXiv:1908.08345 http://arxiv.org/abs/1908.08345

[28] Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The Semantic Scholar Open Research Corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics, Online, 4969–4983. https://doi.org/10.18653/v1/2020.acl-main.447

[29] Daniel Locke and Guido Zuccon. 2018. A test collection for evaluating legal case law search. In *41st International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2018.* Association for Computing Machinery, Inc, New York, NY, USA, 1261–1264. https://doi.org/10.1145/3209978.3210161

[30] Paco Nathan. 2016. *PyTextRank, a Python implementation of TextRank for phrase extraction and summarization of text documents.* https://doi.org/10.5281/zenodo.4637885

[31] José Luis Ortega. 2014. Influence of co-authorship networks in the research impact: Ego network analyses from Microsoft Academic Search. *Journal of Informetrics* 8, 3 (2014), 728–737. https://doi.org/10.1016/j.joi.2014.07.001

[32] Héctor R. Ponce, Richard E. Mayer, Verónica A. Figueroa, and Mario J. López. 2018. Interactive highlighting for just-in-time formative assessment during whole-class instruction: effects on vocabulary learning and reading comprehension. *Interactive Learning Environments* 26, 1 (2018), 42–60. https://doi.org/10.1080/10494820.2017.1282878 arXiv:https://doi.org/10.1080/10494820.2017.1282878

[33] Juliano Rabelo, mi-young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. *A Summary of the COLIEE 2019 Competition.* 34–49. https://doi.org/10.1007/978-3-030-58790-1_3

[34] Julien Rossi and Evangelos Kanoulas. 2018. Query Generation for Patent Retrieval with Keyword Extraction Based on Syntactic Features.. In *JURIX.* 210–214.

[35] Julien Rossi and Evangelos Kanoulas. 2019. Legal Information Retrieval with Generalized Language Models. *Proceedings of the 6th Competition on Legal Information Extraction/Entailment. COLIEE* (2019).

[36] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2020. DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv:1910.01108 [cs.CL]

[37] Serhad Sarica, Binyang Song, En Low, and Jianxi Luo. 2019. Engineering knowledge graph for keyword discovery in patent search. In *Proceedings of the Design Society: International Conference on Engineering Design*, Vol. 1. Cambridge University Press, 2249–2258.

[38] Philipp Schaer, Johann Schaible, and Leyla Jael Garcia Castro. 2020. Overview of LiLAS 2020–Living Labs for Academic Search. In *International Conference of the Cross-Language Evaluation Forum for European Languages.* Springer, 364–371.

[39] Walid Shalaby and Wlodek Zadrozny. 2018. Toward an interactive patent retrieval framework based on distributed representations. In *The 41st International ACM SIGIR conference on research & development in information retrieval.* 957–960.

[40] Walid Shalaby and Wlodek Zadrozny. 2019. Patent retrieval: a literature review. *Knowledge and Information Systems* (2019), 1–30.

[41] Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT Rediscovers the Classical NLP Pipeline. arXiv:1905.05950 [cs.CL]

[42] Vu Tran, Son Truong Nguyen, and Minh Le Nguyen. 2018. JNLP group: legal information retrieval with summary and logical structure analysis. In *Twelfth International Workshop on Juris-informatics (JURISIN), COLLEE.*

[43] Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Tianyang Zhang, Xianpei Han, Zhen Hu, Heng Wang, and Jianfeng Xu. 2019. CAIL2019-SCM: A Dataset of Similar Case Matching in Legal Domain. *CoRR* abs/1911.08962 (2019). arXiv:1911.08962 http://arxiv.org/abs/1911.08962

[44] Chenyan Xiong, Russell Power, and Jamie Callan. 2017. Explicit semantic ranking for academic search via knowledge graph embedding. In *Proceedings of the 26th international conference on world wide web.* 1271–1279.

[45] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How Does NLP Benefit Legal System: A Summary of Legal Artificial Intelligence. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.* Association for Computational Linguistics, Online, 5218–5230. https://doi.org/10.18653/v1/2020.acl-main.466