Result Diversification for Legal Case Retrieval

Ruizhe Zhang
Quan Cheng Laboratory,
Dept. of CS&T, Institute for Internet Judiciary, Tsinghua University
Beijing, China

Qingyao Ai
Quan Cheng Laboratory,
Dept. of CS&T, Institute for Internet Judiciary, Tsinghua University
Beijing, China

Yueyue Wu
Quan Cheng Laboratory,
Dept. of CS&T, Institute for Internet Judiciary, Tsinghua University
Beijing, China

Yixiao Ma
Quan Cheng Laboratory,
Dept. of CS&T, Institute for Internet Judiciary, Tsinghua University
Beijing, China

Yiqun Liu *
Quan Cheng Laboratory,
Dept. of CS&T, Institute for Internet Judiciary, Tsinghua University
Beijing, China
yiqunliu@tsinghua.edu.cn

ABSTRACT

Legal case retrieval has received considerable attention in the last decade. As more and more legal documents are collected and stored in digital form, the need for efficient and reliable access to relevant information in large-scale legal databases continues to grow. While most existing studies have focused on differentiating relevant documents from irrelevant ones based on their similarity to the query case, user studies have revealed that similarity is not the sole concern in legal case retrieval. In many instances, users require not only cases that are similar in content but also cases that encompass a broad range of subtopics (i.e., charges) related to the query case. In contrast to open-domain retrieval, such as web search, our research has found that search diversification in legal case retrieval involves a smaller number of highly correlated subtopics. To address this issue, we have constructed a Diversity Legal Retrieval dataset (DLR-dataset) that includes query-charge labels and charge-level relevance labels between the query case and candidate cases. Additionally, we propose a Diversified Legal Case Retrieval Model (DLRM) that simultaneously considers topical relevance and subtopic relationships using a legal relationship graph. Experimental results demonstrate that DLRM outperforms existing diversified search baselines in the field of legal retrieval.

CCS CONCEPTS

• **Information systems** → *Retrieval models.*

KEYWORDS

diversification, legal search,datasets,retrieval model

ACM Reference Format:
Ruizhe Zhang, Qingyao Ai, Yueyue Wu, Yixiao Ma, and Yiqun Liu [1]. 2023. Result Diversification for Legal Case Retrieval. In *Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region (SIGIR-AP '23), November 26–28, 2023, Beijing, China.* ACM, New York, NY, USA, 11 pages. https://doi.org/10.1145/3624918.3625319

SIGIR-AP '23, November 26–28, 2023, Beijing, China

ACM ISBN 979-8-4007-0408-6/23/11.
https://doi.org/10.1145/3624918.3625319

1 INTRODUCTION

The domain of legal retrieval has witnessed considerable growth in importance in recent years, as highlighted by several studies [22, 30, 37]. This trend can be attributed to the common practice among legal practitioners of searching for and analyzing cases similar to their own from extensive legal document databases, especially when dealing with a specific case, referred to as the query case [2]. In this context, a robust legal case retrieval system becomes an essential tool, enabling efficient and effective analysis of the query case. By doing so, it contributes significantly to the advancement of modern legal systems [14].

The majority of existing research on legal case retrieval has predominantly focused on ranking and evaluating legal documents based on their individual relevance to the query [46]. This is akin to traditional information retrieval challenges such as ad-hoc retrieval and web search, where relevant documents in legal case retrieval are generally lexically or semantically similar to the query. As a result, early methodologies frequently adapted web search retrieval models for legal case retrieval [20].

For example, Wen and Huang [49] utilized the BM25 algorithm [35] to retrieve pertinent documents in legal case retrieval. More recent developments in neural and language modeling, such as BERT [9] and BERT-PLI [42], have been incorporated into legal case search frameworks. Empirical results from previous research have shown that these techniques can deliver state-of-the-art performance in retrieving relevant and similar documents from extensive legal case retrieval benchmarks.

In real-world applications, what legal practitioners need extends beyond merely analogous documents. To make judicious decisions on a case, they frequently examine cases that not only bear similar content to the current case but also cover a wide array of potential subtopics[2] associated with the case.

[1] Corresponding authors.
[2] In our work, we use the term 'charges' to denote subtopics. According to the principle of criminal law, every criminal case must incorporate one or more charges as a cause of action (presented in the indictment). Each charge relates to a specific law in the criminal law. Henceforth in this paper, the terms 'subtopic(s)' and 'charge(s)' are used interchangeably.

Previous research on legal case retrieval has underscored the necessity of addressing such diversity requirements [21]. Therefore, enhancing search diversity in legal case retrieval systems is crucial for improving system quality and user satisfaction. Koniaris et al. [21] attempted to directly implement methods from open-domain searches, like web search, to legal case retrieval. However, we argue that such strategies may not be optimal for several reasons.

Primarily, the underlying motivations for diversified search intents significantly differ between legal case retrieval and open-domain search. In open-domain search, queries typically comprise shorter words, which may convey less information and be more susceptible to ambiguity. Without additional information concerning the query intent, diversifying the search results to cater to a wider user base is the most effective strategy.

In contrast, legal case retrieval frequently incorporates longer and more detailed queries, with users providing extensive specifics about the case of interest. Our study shows that users typically describe the case using an average of 7.07 sentences (refer to Section 2 for more details). However, empirical findings suggest that users often aim to investigate various types of cases with different yet related charges to inform their final judgment decisions, a unique feature of the legal case retrieval task.

Therefore, relying exclusively on document-query similarity, without considering the user's exploration needs, may not be the most effective approach for legal search in practical applications. Moreover, diversification in legal case retrieval poses unique challenges and opportunities. On one hand, the restricted domain of legal case retrieval narrows the scope of search diversification to a finite number of query subtopics, specifically the charges involved in the cases within the data collection. On the other hand, unlike in open-domain search where query subtopics are typically independent of each other, subtopics in legal case retrieval sessions often exhibit interconnections. For example, if a query is relevant to the charge of "abandonment", the user may also require information on the related charge of "abuse". Consequently, the logical associations between charges can provide invaluable insights in legal case retrieval. Without leveraging the relationships between subtopics, diversity models designed for open-domain search that treat query subtopics as independent entities [1, 3, 18, 36] may yield suboptimal performance in legal case retrieval.

This paper presents a comprehensive study on search diversification specifically tailored for legal case retrieval.

Initially, we construct a new legal case retrieval dataset, particularly focusing on search diversity. The dataset comprises 106 criminal cases, all written in Chinese. Unlike previous studies [21], which evaluated search diversity using pseudo aspect relevance labels generated through latent topics extracted by topic modeling approaches, our dataset is the first to contain explicit human annotations on query aspects (i.e., charges) and aspect-level document relevance judgments.

Furthermore, based on insights derived from user studies, we introduce a search diversification algorithm specifically tailored for legal case retrieval, known as the Diversified Legal Case Retrieval Model (DLRM). DLRM goes beyond mere document dissimilarities to model search diversity, and includes explicit modeling of subtopic relationships using a legal relationship graph. The final ranking of results is established by considering both text similarity between query-document pairs and the relationships among diverse charges in legal case retrieval.

Through our experiments, we demonstrate that DLRM significantly outperforms both non-diversity baselines and state-of-the-art search diversification methods within the context of legal case retrieval.

In conclusion, this study makes significant contributions in the following areas:

- We create the first legal diversification dataset, integrating human annotations for both query-subtopic relationships and subtopic-level relevance judgments. This dataset is an essential tool for the evaluation and development of search diversification in the legal field.
- We present the DLRM, which improves legal case retrieval by leveraging the links between queries and related charges. Through explicit modeling of these relationships, DLRM enhances the quality of search results in the legal sector.

2 RELATED WORK

We present a brief review of associated work in two distinct classifications: legal case retrieval and diversification. The first classification encapsulates conventional methods utilized in the field of legal case retrieval, while the second classification explores diversification techniques typically employed in web search.

2.1 Legal case retrieval

The digitization of legal judgments has surged in recent years, escalating legal case retrieval as a pivotal research issue in both Information Retrieval (IR) and legal fields [40, 44, 46]. As a result, several strategies have adapted web search retrieval models for legal case retrieval, leading to the development of dedicated legal case retrieval models [24–26, 41, 42].

Unlike web retrieval, legal case retrieval often involves users inputting comprehensive case descriptions to find similar cases. Web search utilizes various ranking methods such as BM25 [35], TF-IDF [39], and Learning To Rank (LTR) [28]. Deep learning techniques like Deep Structured Semantic Models (DSSM) [15], Convolutional Neural Networks (CNN) [43], Recurrent Neural Networks (RNN) [32], and Match-SRNN [47] are also employed to improve ranking performance. However, these established methods often fall short when dealing with long query cases, unlike their effectiveness with short query terms in web retrieval.

For extensive query cases, additional user requirement information can be analyzed and incorporated to optimize the ranking method. Within the legal case retrieval domain, Van Opijnen and Santos [46] examined the definition of relevance in law, and Bench-Capon et al. [2] proposed diverse approaches to legal case retrieval. Shao et al. [42] suggested BERT-PLI to enhance legal case retrieval, following the method in PACRR [16]. However, these methods neglected the impact of charges, despite charges associated with cases being crucial in determining the relevance of results.

2.2 Diversification

In web search, it is a standard practice to construct a diverse ranking list. Oftentimes, users input query text replete with ambiguity and redundancy. To cater to diverse user intents and satisfy their

information requirements, numerous methods have been proposed [36].

Carbonell and Goldstein [3] introduced the Maximal Marginal Relevance (MMR) method to construct a novelty ranking list. Other methods such as Risk Minimisation (RM) [51], Conditional Relevance (CR) [7], Mean-Variance Analysis (MVA) [48], Quantum Probability Ranking Principle (QPRP) [55], Absorbing Random Walk (ARW) [53], and Sparse Spatial Selection Diversification (SSSD) [12] were proposed as alternative ways to optimize the novelty of the result list. Coverage-based approaches like Ranked-armed Bandits (RAB) [33], Facet Modelling (FM) [4], and Score Difference (SD) [18] have also proven effective in enhancing the diversification of Search Engine Results Pages (SERPs).

Hybrid methods, such as Weighted Word Coverage (WWC) [45], Diversification Perceptron (DP) [34], Relational Learning to Rank (RLTR) [54], Diversified Data Fusion (DDF) [27], and IA-select [1] have been proposed, marrying the concepts of novelty and coverage. Later, diversification algorithms based on reinforcement learning, such as M2DIV [10], were proposed and considered state-of-the-art. These methods consider both novelty and coverage to improve the diversification of the ranking list.

Diversification in legal case retrieval presents unique challenges compared to web retrieval. Users typically input a full legal case, leading to queries that are less ambiguous but more redundant. Furthermore, when a query refers to a specific charge, users may also seek cases pertaining to different charges, unlike web searches. This is because users frequently need to compare, reference, discern, and contrast various cases.

3 LOG ANALYSIS ON LEGAL CASE RETRIVAL

In this section, we provide a log analysis to elucidate the diverse intents of users in legal case retrieval. The data we analyzed were sourced from a commercial criminal legal case search engine[1]. On this platform, users can submit keywords related to potential charges of candidate cases to the search box, and the system returns a list of cases highly correlated with the input charges.

We collected real search logs from the search engine, amassing a total of 281 search sessions. All of these sessions pertained to the search for criminal cases. In this paper, our focus is solely on the criminal cases, and we treat the types of charges as subtopics for queries. Henceforth, we will use the terms "subtopic" and "charge" interchangeably.

Existing research on Web search diversification [1, 13] posits that:

- Should a query incorporate a certain word (or phrase), it is probable that the user's intent is connected to this word.
- If a user selects a result associated with a specific word or phrase, their intent is likely connected to this word or phrase.

By analogy, in the realm of legal case retrieval, we propose that:

- If a query includes a specific charge, the user's intent is likely connected to this charge.
- If a user selects a result tied to a particular charge, their intent is likely connected to this charge.

[1]https://xszk.chineselaw.com/case

Building on the aforementioned assumptions, we aim to address the following research question—**RQ1:** *Do users incorporate different charges in their queries within a single search session?*

In legal case retrieval, users often submit multiple queries within a single search session [40]. This legal search engine allows users to enter a text-only query term. In addition, the engine allows users to enter (or not to do so) a crime of the case, and this input(crime) will be added to the search engine's filters, assisting the system in providing results to the user. We consider queries from the same user within 30 minutes to be the same session, and queries longer than 30 minutes will be cut into different sessions. Our study aims to determine if diverse intents are exhibited within a single session. We calculate the number of sessions involving varying quantities of different charges in the queries (#*Charges in Queries per Session*) and present these findings in Table 1. The table indicates that over 46% of sessions directly incorporate more than one charge in user queries, while over 20% involve at least three charges.

Table 1: The table displays the distribution of sessions with a specific count of Charges per Session(in Query Terms), denoted briefly as $\#C/S\ (queryterms)$. The data indicates that users include more than one charge in their query text in over 46% of sessions, suggesting that nearly half of the users articulate their interest in multiple subtopics directly.

$\#C/S\ (query\ terms)$	1	2	3	4	5	≥ 6
Percentage	54%	24%	9%	3%	1%	9%

Hence, it is justifiable to conclude that a significant proportion of users search for queries with multiple charges within a single search session.

The second research question we aim to explore is as follows: **RQ2:** *Do users interact with results encompassing different charges within a single legal case search session?* Prior user studies in Web search [1, 6] reveal that users may not consistently express their search intents in their queries. In such scenarios, the documents clicked in a search can serve as a crucial signal reflecting user's intents.

Consequently, we quantify the number of distinct charges encompassed in the documents clicked during each session, a metric we refer to as #*Charges in clicked documents per Session*. The findings are illustrated in Table 2. As demonstrated in the table, 89% of

Table 2: The table illustrates the distribution of sessions with a particular count of Charges per Session(as reflected in clicked documents), encapsulated as $\#C/S(clicked)$.

$\#C/S\ (clicked)$	1	2	3	4	5	≥ 6
Percentage	11%	14%	10%	14%	17%	34%

the sessions involve clicked documents that encompass a range of charges. Over half of the users interact with documents containing as many as five charges within a singular session. This suggests that in the context of legal case retrieval, users frequently desire

results that address a substantial number of charges in practical applications.

One potential limitation of quantifying distinct charges in the clicked documents is the oversight that, within the realm of legal case retrieval, a single document could encompass multiple charges. Drawing the conclusion that users possess diverse intents when all clicked documents cover the exact identical set of charges can be precarious.

To mitigate this issue, we further quantify the distinct sets of charges within the clicked documents for each session. For instance, assume a user has clicked on two documents, X and Y, where X includes charges $\{A, B\}$ and Y includes charges $\{B, C\}$. In this case, the number of distinct charge sets is considered to be 2 ($\{A, B\}$ and $\{B, C\}$)[2].

We then plot the distribution of *#Charge sets in clicked documents per Session* in Table 3. Observations from Table 2 and Table 3 are

Table 3: The table presents the distribution of sessions with a specific count of Charge Sets per Session(as represented by clicked results), denoted as $\#CS/S(clicked)$.

$\#CS/S\,(clicked)$	1	2	3	4	5	≥ 6
Percentage	16%	44%	19%	10%	3%	8%

remarkably consistent. Around 84% of sessions display varied user intents, suggesting users seek documents encompassing multiple charges. Approximately 22% of users require more than three distinct charge sets, with 44% needing two distinct sets. This suggests that most users engaged in legal case retrieval are interested in a diverse range of search results.

Considering the distinctive nature of legal retrieval, the need for search diversification is anticipated. Unlike web searches, legal practitioners often require comprehensive investigations. They routinely delve into a variety of cases and charges to support their decisions, a practice reflected in their search behaviors.

Therefore, the significance of search diversity in legal retrieval is substantial. This observation prompts us to further develop datasets and algorithms tailored for legal search diversification.

4 DATASET CONSTRUCTION

In this section, we detail our laboratory study and the creation of a novel legal case retrieval dataset that emphasizes search diversity. We have designated this dataset as the Diversity Legal Case Retrieval Dataset (DLR-dataset).

4.1 Overview

Securing reliable and reusable datasets is a crucial step in the construction of effective retrieval models. Numerous datasets [8, 23] have been proposed for Web search diversification, leading to the development of a wide array of successful algorithms. However, in the realm of legal retrieval, no public dataset currently exists for search diversification.

Previous studies have suggested extending existing legal search datasets with pseudo aspect-level relevance labels, treating the latent topics (extracted by topic models) of each document as the subtopics of each query [50]. Regrettably, subtopics extracted in this manner are neither reusable nor reliable, as the outputs of latent topic modeling approaches are typically unstable. Consequently, experiments based on such datasets are challenging to reproduce.

To circumvent these challenges, we construct the DLR-dataset using direct human annotations. Generally, the creation of a dataset for search diversification involves two primary tasks: the identification of query subtopics and the annotation of document relevance at the subtopic level. Therefore, our laboratory study primarily focuses on two objectives:

- Comprehending the distribution of user intents (on charge levels) within a particular query case.
- Determining whether a candidate document can satisfy a user's search intent regarding a specific charge.

Our lab study is based on a legal case retrieval dataset for the Chinese law system [29]. This dataset, written in Chinese, consists of 107 criminal cases, with each case providing 100 judgments as candidate documents.

We utilize 106 query cases from this dataset, excluding one query case due to its length (29 sentences, which is significantly longer than others). For these 106 query cases, we evaluated the number of sentences in each query case. The results presented in Table 4 demonstrate that the query terms in legal case retrieval are significantly longer than those in web search.

Table 4: The number of sentences per query case varies, with most cases comprising between 5 to 10 sentences. The shortest case contains 2 sentences, while the longest extends to 20 sentences. On average, there are 7.07 sentences per query case.

#Sentences	≤ 5	$(5, 10]$	$(10, 15]$	> 15
Percentages	35.85%	39.62%	19.81%	5.66%

For each query case, we utilize the top 30 candidate documents retrieved by BM25 from the top-100 candidate documents in this dataset. Specifically, we first employ THULAC (THU Lexical Analyzer for Chinese) [31] for word segmentation, followed by the use of BM25 to calculate the relevance score between the query case and each candidate case.

The construction of the DLR-dataset is a two-step process. Initially, we annotate the relevance between queries and potential charges. Subsequently, we label the relevance of each query-charge-document triplet based on the results derived from the first step. Detailed information about the specific notations utilized in this paper is provided in Table 5.

4.2 Query-charge Relevance Annotation

In the first step, our objective is to discern potential user intentions related to charges (i.e., $\{I_k\}$) within a specific query case Q_i. Specifically, we aim to comprehend the distribution of requirements for varying charges amongst users who have submitted the query.

[2]We regard the subset of a specific set as distinct ones.

Table 5: Definitions of Notations

Variable	Description
$n = 107$	#query cases
$m = 30$	#candidate cases/query cases
$s = 272$	#charges
$Q = \{q_i\}\ (i \in [1, n])$	query cases
$I = \{I_k\}\ (k \in [1, s])$	intents on charges
$D = \{d_{ij}\}\ (i \in [1, n], j \in [1, m])$	candidate documents

4.2.1 Annotation Process. Due to the dataset's immense size, annotating all query-charge pairs is impractical. We adopt a two-step procedure to create a candidate charge pool for each query. Initially, we extract charges from the query string using regular expressions. Then, we employ a legal judgement prediction model to forecast the five most pertinent charges to the query case. These charges are merged with those extracted initially to form the final Candidate Charge Set.

We engaged eight experienced legal practitioners as annotators for the actual annotation process. They were directed to review the query cases' descriptions and the Candidate Charge Set (CCS), followed by the selection and ranking of relevant charges. For a given query, annotators selected pertinent charges from the CCS, ranking them by relevance. For example, given a CCS of $I_1, I_2, I_3, I_4, I_5, I_6$, an annotator might produce a ranked list like $I_2 = I_3 > I_1 > I_5 = I_6$, disregarding I_4 as irrelevant.

Our created CCS may not cover all relevant charges for each query. Annotators were asked to submit any additional relevant charge(s) to the dataset, but none were provided, indicating the effectiveness of our charge candidate collection method.

4.2.2 Result Analysis. In this section, we briefly discuss the results obtained from the query-charge relevance annotation. Initially, we present the number of unique relevant charges identified for each query in Table 6. We merely amalgamate the annotated relevant charges from all annotators to formulate the final charge set for each query. As depicted in the figure, every query in our dataset possesses at least two relevant charges. Moreover, over 90% of the queries contain more than three relevant charges. These findings suggest that legal search users frequently exhibit a strong need to inspect documents referring to multiple relevant charges.

Table 6: Upon analyzing the labeling results, we can make the following observations: 1. Participants exhibit diverse intents for a specific query case. 2. In the majority, the number of potential intents a user may have ranges from 3 to 5.

Size of intent(s) set	1	2	3	4	5	6
Percentage	0.0%	6.5%	35.5%	33.6%	21.5%	2.8%

Furthermore, we scrutinize the distribution of the significance of each relevant charge within each query. Through the select-and-sort annotation process, we have gathered the annotators' preferences regarding relevant charges in each query. This data can subsequently be utilized to construct multi-level relevance labels for charges. Table 7 indicates the number of relevance levels the annotators have designated for each query. For instance, a sorted list $I_2 = I_3 > I_1 > I_5 = I_6$ derived from a CCS= $\{I_1, I_2, I_3, I_4, I_5, I_6\}$ signifies four relevance levels for charges, namely, *perfect* for $\{I_2, I_3\}$, *excellent* for $\{I_1\}$, *fair* for $\{I_5, I_6\}$, and *irrelevant* for $\{I_4\}$.

Table 7: The participants differentiated the levels of importance (LoI) in distinct ways. In 48.1% of the cases, participants bifurcated the results into only 2 LoI. Conversely, in 41.9% of instances, participants divided the results into 3 LoI. Notably, less than 1% of participants divided the intents into more than 5 LoI.

LoI	2	3	4	5	6
percentage	48.1%	41.9%	8.9%	0.9%	0.1%

Table 7 illustrates that 48.1% of the queries exhibit two-level relevance judgments, namely *relevant* and *irrelevant*, while 41.9% of the queries display three-level judgments. Approximately 10% of the queries demonstrate more than three levels. This significant variance in the number of relevance levels among different queries suggests that a relevance grading method with a predetermined number of possible levels is not apt for query-charge annotation.

Beyond the importance of each charge, legal system designers may be more concerned with the distribution of user intents within each query, specifically $P(I_k|Q_i)$. Regrettably, there is no straightforward solution to acquire such information without conducting a large-scale user survey. In this paper, we employ a simplistic strategy to calculate intent distributions based on the annotated query-charge pairs.

Initially, for each sorted intent list, if the annotator has divided the results into k levels, we select k value(s) uniformly within the range $[0, 1]$ to represent the probability of each intent appearing in the query. For instance, if a sorting list is $I_2 = I_3 > I_1 > I_5 = I_6$ with I_4 unselected, the probability of each charge being the query intent is computed as 1 for $\{I_2, I_3\}$, $\frac{2}{3}$ for $\{I_1\}$, $\frac{1}{3}$ for $\{I_5, I_6\}$, and 0 for $\{I_4\}$.

Subsequently, we average the intent distribution gathered from all annotators to obtain the final $P(I_k|Q_i)$.

4.3 Charge-level Relevance Annotation for Query-document Pairs

Considering the relevance annotation on query-charge pairs, the subsequent segment of our laboratory study aims to gather detailed relevance information for each query-charge-document triplet, denoted as $(Q_i, I_k, d_{i,j})$.

4.3.1 Annotation Process. For the actual annotation process, we enlisted nine annotators, including the previous eight, all of whom are legal practitioners with substantial legal knowledge.

We randomly and evenly divided the nine annotators into three groups. Each group was tasked with annotating the documents in 35 (or 36) queries. We collected all labels from the three annotators and recorded them in the dataset. We then used the median of the scores within each group as the final relevance labels for the $(Q_i, I_k, d_{i,j})$ triplets in subsequent experiments.

However, we realized that the task of annotating all the triplets $(Q_i, I_k, d_{i,j})$ would be overwhelmingly labor-intensive and unmanageable. The numbers of Q, I, and $d_{i,j}$ are 106, 272, and 100, respectively. Annotating a single triplet could take an annotator between 2 to 4 minutes (with an average of 3 minutes). The total time cost would be $106 \cdot 272 \cdot 100 \cdot 3 = 8,649,600$ minutes, approximately 16 years. This is entirely impractical.

We hypothesize that a candidate judgment 'may' satisfy the users' information need for intent I_k if and only if the user has an information need for intent I_k when searching the query case Q_i, and the document $d_{i,j}$ is relevant to the intent I_k.

Consequently, in this step, we only required annotators to label a portion of the triplets. A triplet $(Q_i, I_k, d_{i,j})$ would be labeled by annotators if and only if all three conditions were met:

- (1) The probability of observing the charge in the user intent of the query ($P(I_k|Q_i)$) is above zero (as labeled in step 1);
- (2) The document $d_{i,j}$ is relevant to the query Q_i (labels derived from LeCard, which we based on [29]);
- (3) The document $d_{i,j}$ is relevant to the charge I_k (filtered from 272 subtopics).

To filter charges under condition 3, we employed the same filtering process as in step 1. We combined the top-5 relevant charges from the LJP models and charges extracted by regular expressions as relevant charges for document $d_{i,j}$. In the legal domain, word usage is stringent, especially for crimes where the terminology used in criminal law is the only correct one. Hence, the validity of using regular expressions for matching.

This methodology significantly reduced the annotators' workload. On average, each annotator was required to annotate about 1858.3 triplets. Consequently, this streamlined process meant each annotator spent approximately $1858.3 \cdot 3 = 5574.9$ minutes (about 92 hours) on this step of the annotation process.

A triplet $(Q_i, I_k, d_{i,j})$ would only have a non-zero relevance label if: (1) the probability of observing the charge in the user intent of the query ($P(I_k|Q_i)$) is above zero; (2) the document $d_{i,j}$ is relevant to the query Q_i; and (3) the document $d_{i,j}$ is relevant to the charge I_k. In this study, the documents we consider are legal cases with judgments, implying that we can directly extract the relevant charges of a document from its content. After extracting each document's relevant charges, we enlisted nine annotators (all of whom are legal practitioners with substantial legal knowledge) to assign a four-level relevance label (i.e., *perfect*, *excellent*, *fair*, and *irrelevant*) for each triplet $(Q_i, I_k, d_{i,j})$.

Table 8: Label Quality Analysis. The participants were stratified into three groups (A, B, C), with each group consisting of three individuals assigned to the same labeling tasks. The findings indicated a significant degree of concordance among participants, underscoring the superior quality of the labeling results.

Measure	All	Group A	Group B	Group C
Krippendorff's α	0.448	0.471	0.403	0.469
avg(Fleiss' Kappa)	0.461	0.437	0.513	0.433
avg(Kendall's τ)	0.740	0.707	0.767	0.747

Table 9: The basic information of the DLR-dataset.

	Training	Test
#Querys	70	36
#Candidate / Query	30	30
#Queries' Relevant charge(s)	3.54	3.50
#Sentences / Query(Avg.)	7.78	7.56
#Sentences / Document(Avg.)	188.12	181.69

4.3.2 Basic Information and quality of label results. In this study, we employed statistical measures such as average Fleiss' kappa [11], the average Kendall rank correlation coefficient [17], and Krippendorff's α [5] to evaluate the quality of the label results. These computations and visualizations are presented in Table 8. Overall, the high degree of agreement amongst annotators suggests the reliability of the labels derived from the annotation process.

Subsequently, the queries were randomly divided into a training set and a test set at a ratio of 2:1. Details in Table 9.

5 DIVERSIFIED LEGAL CASE RETRIEVAL MODEL

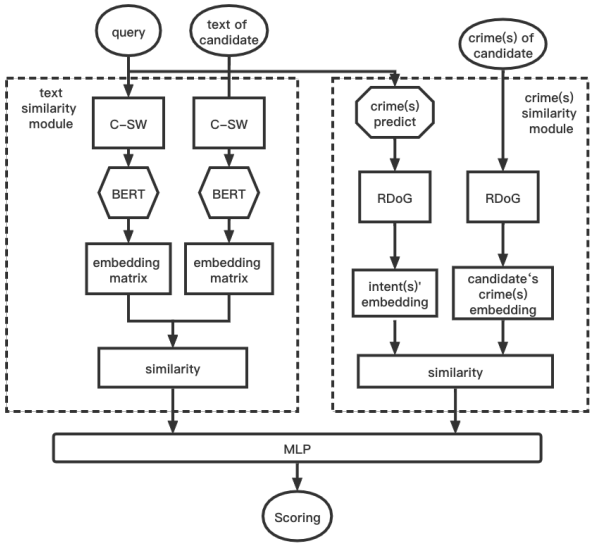

Figure 1: An overview of the DLRM. DLRM is a sophisticated system that integrates a text similarity module and a charge similarity module. The results generated by these modules are subsequently synthesized using a Multi-Layer Perceptron (MLP) model. Within the text similarity module, 'C-SW' denotes the 'Cut with Sliding Windows' technique, a process that enhances the accuracy of text analysis. Conversely, in the charge similarity module, 'RWoG' stands for the 'Random Walk on Graph' process, a method employed to bolster the precision of charge-related evaluations. This intricate combination of modules and processes ensures the DLRM's superior performance in legal case retrieval.

Diversified Legal Case Retrieval Model (DLRM) is a novel model aimed at optimizing legal case retrieval. DLRM amalgamates a text similarity module, a charge similarity module, and a multi-layer perceptron model (MLP), as shown in Figure 1. DLRM operates by processing a query case (Q_i) and a candidate document (d_{ij}), each associated with multiple charges (Section 4.3.1). This processing yields a ranking score for the candidate document, which is essential for the retrieval and ranking of relevant legal cases, thus effectively addressing the initial query.

5.1 Text Similarity Module

Textual similarity is a vital determinant of document relevance. In the DLRM, we have incorporated a text similarity module to capture both lexical and semantic similarities between the query and document text. This module uses raw text from a query case and a candidate document as input, and outputs an embedding, T_{sim}, encoding semantic information from the raw text, thereby providing a comprehensive representation of textual similarities.

Due to their extensive length, legal documents pose a challenge for direct processing using neural language models. This paper addresses this issue by introducing a strategy called Cut with Sliding Windows (C-SW) module, which segments lengthy text into smaller, overlapping passages.

Formally, the input to the C-SW module is a text segment composed of l sentences, represented as $\{s_1, s_2, ..., s_l\}$. Given a sliding window of size w and step d, the first output passage from the C-SW module would be $\{s_1, s_2 ... s_w\}$, whereas the second would be $\{s_{1+d}, s_{2+d} ... s_{w+d}\}$, and so forth. To ensure each passage contains w sentences, we pad the output passage with empty strings. In our approach, we set w and d to be 3 and 1 for each input query, and 13 and 5 for an input document, respectively.

For each output passage derived from the C-SW module, we employ a pre-trained BERT model to encode and construct an embedding representation of the passage (i.e., the 768-dimensional vector of [CLS] in BERT). Let n and m denote the number of passages extracted for a query and a document, respectively. We then compute a similarity matrix $M \in \mathbb{R}^{n \times m}$, where $M_{i,j}$ represents the cos-similarity of the i-th passage of the query and the j-th passage of the document.

Subsequently, we apply a max-pooling layer over the query passages (i.e., rows in M) to derive a similarity vector $T_s \in \mathbb{R}^n$. To generate an input vector of fixed length for the MLP model, we concatenate $\{T_s, \phi, T_s\}$, where ϕ is a sequence of zeros, to form a vector T_{sim} of fixed length (for instance, 54 in our experiments).

5.2 Charge Similarity Module

As elaborated in Section 4, legal search users often require documents pertinent to multiple charges, and the number of potential charges within a legal corpus is finite. Unlike in Web search, we observe strong correlations between documents associated with specific charges in our user study. In other words, the relevance of query-charge-document triples $(Q_i, I_k, d_{i,j})$ with varying charges is not mutually independent. Neglecting such information could lead existing search diversification algorithms to yield suboptimal results in legal case retrieval.

To address this issue, we suggest the creation of a charge-similarity module that encapsulates charge relationships for legal case retrieval. We begin by building a legal relationship graph using charges extracted from each query and document. We then apply a random walk algorithm, known as Random Walk on the Graph (RWoG), to encapsulate the semantic relationships between query charges and document charges. Finally, we calculate the charge similarity between the query and the document using their embeddings, which is then used as the module's output.

Our legal relationship graph is built using judgments from different case trials. While most documents contain single-trial cases, some have multiple trials with overturned judgments. Charge reversals between judgments indicate strong connections between charges. This implies that presenting both charges to users in legal retrieval could be advantageous. Based on this insight, we construct a legal charge graph that incorporates charge reversal information.

Initially, we count the frequency of a charge i being reversed by a charge j. Let this frequency matrix be $G \in \mathbb{N}^{s \times s}$, where s is the total number of possible charges in our dataset (i.e., $s = 272$). We then treat each charge as a node and construct directional edges among them in the following manner:

- If $\forall j\ G_{ij} = 0$, we add a self-loop for node i with weight 1.
- If $\exists j\ G_{ij} > 0$, we add an edge from i to j with weight E_{ij}.

$$E_{ij} = \begin{cases} \alpha & \text{if } i = j \\ (1-\alpha) \cdot \frac{G_{ij}}{\sum_{k=1}^{s} G_{ik}} & \text{if } i \neq j \end{cases} \tag{1}$$

Utilizing the relationship graph, we implement a Random Walk on the Graph (RWoG) to derive the charge-based embedding representations of both the query and the document. Let $C_{qo} \in \mathbb{R}^s$ and $C_{do} \in \mathbb{R}^s$ denote binary vectors that represent the relevance of a charge to the query and document, respectively.

For each query, we construct C_{qo} using the query's charge candidate set, as discussed in Section 4.2.1. The ith dimension of C_{qo} is set to the output of the Legal Judgment Prediction (LJP) model, an end-to-end model with case descriptions as inputs. The output of the LJP model is a probability vector of $1 \cdot 272$, indicating the corresponding crime(s) for the case description.

For each document, we construct C_{do} using the document's charge candidate set, as discussed in Section 4.3.1. The ith dimension of C_{do} is set to 1 if charge i is relevant to the query (or document), and 0 otherwise. Both C_{qo} and C_{do} are normalized to ensure the sum of their elements equals 1.

Following this, we set the initial node probabilities separately with C_{qo} and C_{do}, and execute RWoG twice to extract new node probability vectors $C_q \in \mathbb{R}^s$ and $C_d \in \mathbb{R}^s$, which serve as the final charge representations of the query and the document, respectively.

Given that the number of charges is fixed and the number of nodes in the graph is 272, the time complexity of the random walk is $O(Tn^2)$, where T represents the number of rounds and $n = 272$ is the number of nodes. For graphs of this scale, this complexity is entirely manageable.

The final output of the charge similarity module, denoted as the charge similarity embedding C_{qd} between the query and the document, is calculated as follows: $C_{qd} = C_q \bigotimes C_d$. In this equation, C_{qd} represents the Kronecker product of C_q and C_d.

5.3 Ranking Prediction and Model Training

To establish a final ranking for the candidate documents, we combine the outputs of the text similarity module (T_{qd}) and the charge similarity module (C_{qd}) to form an input vector. This vector undergoes processing by a Multi-Layer Perceptron network (MLP) to predict each document's ranking score. The documents are then sorted based on their respective scores to produce the result list, adopting the methodology of ColBERT [19].

It's worth noting that, despite not explicitly modeling document novelty in the ranking process, the DLRM's charge similarity module intrinsically aids in estimating ranking scores based on query intent distribution. As a result, our DLRM inherently encapsulates search diversity within its results. DLRM's effectiveness in search diversification is further showcased in Section 6.

Considering the limited size of our training data and a designated ranking metric (e.g., NDCG-IA@10), we use the following approach to train our model. Initially, we randomly select a query Q_i and associated candidate documents $\{d_i\}$ from the training set. For a particular document d_{ij}, we randomly select a position k (i.e., $k \in [1, 10]$) and place d_{ij} at k. The remaining positions are randomly filled with documents chosen (without replacement) from $\{d_i\}$, and we calculate the expected metric rewards (e.g., NDCG-IA@10), denoted as $\mathbb{E}(R(k, d_{ij}))$, of the randomly sampled ranked list.

The final label $l(k, d_{ij})$ assigned to the document d_{ij} is computed as follows:

$$l(k, d_{ij}) = \frac{\mathbb{E}(R(k, d_{ij})) - \min_a \mathbb{E}(R(k, d_{ia}))}{\max_a \mathbb{E}(R(k, d_{ia})) - \min_a \mathbb{E}(R(k, d_{ia}))} \tag{2}$$

Let the ranking score of d_{ij} be γ_{ij} (i.e., the output of MLP). Our model is trained by minimizing the mean square errors between $l(k, d_{ij})$ and γ_{ij}. To ensure the reliability of the entire training process, we repeat this process one million times to produce the final model.

6 EXPERIMENT

6.1 Experimental setup

We carry out our experiments using the proposed DLR-dataset, as described in Section 4, and we employ two widely recognized evaluation metrics for search diversification. The first one is α-NDCG [38]. The computation of α-NDCG operates under the assumption that all search intents are distributed evenly. Accordingly, we filter out charge I_k with $P(I_k|Q_i) \leq 0.5$ for each query Q_i and utilize the remaining intents as relevant ones for Q_i in the computation of α-NDCG.

Additionally, within α-NDCG, each document can only be classified as *relevant* or *irrelevant* to a query intent. Consequently, we convert the four-level relevance labels of each query-document-charge triple into a binary label (2,3 as 1, and 1,0 as 0) for the calculation of α-NDCG.

The second metric we employ is NDCG-IA [1]. Specifically, we use $P(I_k|Q_i)$ from the ground truths as the weight of each intent in NDCG-IA:

$$\text{NDCG-IA}(Q_i) = \sum_{I_k} P(I_k|Q_i) \cdot NDCG(Q_i|I_k) \tag{3}$$

Here, $NDCG(Q_i|I_k)$ is computed using the four-level relevance labels of each query-document-charge triple.

For the sake of comparison, we incorporate five established baselines into our experiments:

- **BM25** [35]: This is a traditional retrieval model that employs the BM25 function to assess the relevance between query cases and candidate cases.
- **MMR** [3]: This renowned diversification algorithm ranks documents based on both their relevance scores and novelty. Specifically, MMR calculates the ranking score of a document as a linear combination of its relevance score to the query and its novelty in relation to the previously selected documents.
- **IA-select** [1]: IA-select ranks documents according to their relevance to distinct query intents. It separately calculates the relevance scores of a document for each query intent and establishes the final ranking by harmonizing documents relevant to various query intents. In this instance, we employ the initial intent distribution extracted by LJP (i.e., C_{qo}) to represent the intents for each query.
- **exIA-select**: An extended version of IA-select that employs the query intent vector learned by RWoG in the charge similarity module of DLRM (i.e., C_q) as the intent distributions of each query.
- **M2DIV** [10]: This is a state-of-the-art diversification algorithm that constructs a policy-value network with reinforcement learning to diversify search results.

For the implementation of MMR, IA-select, and exIA-select, we adhere to the experimental design proposed by Devlin et al. [9], utilizing a BERT model to encode both the query and the document into latent vectors. The relevance score is calculated as the cosine similarity between the BERT vectors of the query and the document.

For the MMR algorithm, we calculate a document's novelty by averaging its cosine similarity with the selected documents. We fine-tune the hyperparameter of the linear combination function from 0 to 0.1. As for other baselines, we perform a grid search to find the optimal hyperparameters. Only the performance of baselines with the best discovered hyperparameter settings is reported. Regarding M2DIV, we retrain it using our training set.

Regarding the DLRM, all parameters, with the exception of those for the MLP network, are fixed after the pre-training process. We solely train the MLP network based on the training data. We use the Adam optimizer with a learning rate of 10^{-5} to train the MLP model. The three hidden layers in the MLP have sizes of 128, 32, and 4. The α in RWoG is set at 0.4. The LJP module in our experiment is the TopJudge [52]. To smooth the initial intent distribution predicted by LJP, we add 0.3 to the LJP outputs of the top 5 charges for each query, followed by normalization as described in Section 5.2.

6.2 Overall Results

Table 10 presents the performance of DLRM and all baseline methods. As the table illustrates, DLRM surpasses all baselines. Notably, the improvement of DLRM over the most proficient existing search diversification baseline (i.e., exIA-select) is 20% or more in terms of NDCG-IA. This clearly underscores the efficacy of DLRM as a search diversification model for legal case retrieval.

Table 10: Experiment result on DLR-dataset. $N-IA$ stands for $NDCG-IA$ and $\alpha-N$ stands for α-NDCG. */ denote significant differences with respect to the best baseline (exIA-select) at $p < 0.05/p < 0.01$ level using the pairwise t-test, respectively.**

	$N-IA@1$	$N-IA@3$	$N-IA@5$	$N-IA@10$	$\alpha-N@1$	$\alpha-N@3$	$\alpha-N@5$	$\alpha-N@10$
BM25	0.4537	0.4783	0.4921	0.5278	0.5448	0.4970	0.5085	0.5621
MMR	0.4537	0.4769	0.4978	0.5181	0.5448	0.5053	0.5302	0.5621
IA-select	0.4951	0.5070	0.5194	0.5548	0.5686	0.4727	0.4520	0.4443
exIA-select	0.6023	0.5971	0.6069	0.6370	0.7419	0.6291	0.6185	0.6286
M2DIV	0.5569	0.5505	0.5611	0.5778	0.6238	0.5485	0.5586	0.5858
DLRM	**0.7199****	**0.7389****	**0.7753****	**0.8747****	**0.8143***	**0.6786***	**0.6521***	**0.6450**
improve.	19.5%	23.7%	27.7%	37.3%	9.8%	7.9%	5.4%	2.6%

DLRM outperforms the baseline model across all metrics, including NDCG-IA@1. This suggests that DLRM not only enhances the quality of the ranking result list but also improves the top 1 results, as it takes into account diverse information needs.

To further highlight the benefits of charge similarity modeling in legal case retrieval, we compare the results of IA-select and exIA-select. In exIA-select, we replace the intent (charge) distribution (C_{qo}) used in IA-select with one derived from our legal relationship graph (C_q). As illustrated in Table 10, exIA-select surpasses IA-select on all metrics, with statistically significant differences. This implies that our DLRM's charge similarity module can effectively discern the distribution of query intents, thus yielding more effective search diversification models.

Our experiments reveal an interesting finding: MMR and BM25 exhibit similar performance. Despite MMR being implemented with a more advanced model (BERT), the addition of document novelty did not enhance its overall effectiveness. Surprisingly, we observed that lower weights assigned to document novelty in MMR generally resulted in better outcomes. This suggests that search diversification algorithms that solely rely on document differences may not meet the needs of legal case retrieval users.

6.3 Ablation study

In this paper, we devise the DLRM, constituted by two modules: the text similarity module and the charge similarity module. To further demonstrate the effectiveness of each module, we conduct an ablation study.

Specifically, we design three variations of the DLRM, as follows:

- **Text Only**: Text similarity module + MLP.
- **Charge Only**: Charge similarity module + MLP.
- **None (Random)**: MLP only. The input of the MLP network is randomly initialized for each query-document pair.

Table 11 presents the ranking performance of each model. As illustrated in the table, all variations of the DLRM underperformed in comparison to the complete DLRM. Among all variations, the charge-only model performed the best. This underscores the importance of charge similarity modeling in legal case retrieval.

Table 11: An ablation study reveals each DLRM component's contribution to enhancing NDCG-IA@k performance. Boldface indicates the most effective setting. */ denote significant performance differences compared to the full model at $p < 0.05/0.01$ levels, as determined by a pairwise t-test. N-IA is an abbreviation for NDCG-IA.**

Model	N-IA@1	N-IA@3	N-IA@5	N-IA@10
None(Random)	0.4441**	0.4463**	0.4496**	0.4694**
Text Only	0.5637*	0.5707**	0.5790**	0.6147**
Charge Only	0.6340	0.6587*	0.6877*	0.7620**
DLRM	0.7199	0.7389	0.7753	0.8747

7 CONCLUSION AND FUTURE WORK

This paper explores the necessity of search diversity in legal case retrieval. We conduct a comprehensive analysis, including the development of a novel dataset with human-annotated intent-level document relevance. We investigate and model the relationships between charges in queries and documents, introducing DLRM as a superior search diversification algorithm for legal case retrieval.

Our research initiates a novel exploration into the potential of search diversification in legal retrieval. Going forward, we aim to further investigate the necessity of search diversity and the integration of diverse domain knowledge to improve legal retrieval models. Further exploration will include substituting our relationship graphs with knowledge graphs.

Notably, our experimental evaluation employs standard diversity metrics, which are not tailored for legal retrieval. Consequently, developing effective evaluation methods and satisfaction prediction techniques for legal case retrieval is another crucial aspect we intend to address in future work.

ACKNOWLEDGMENTS

This work is supported by Quan Cheng Laboratory (Grant No. QCLZD202301) and the Natural Science Foundation of China (Grant No. 62002194).

REFERENCES

[1] Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying search results. In *Proceedings of the second ACM international conference on web search and data mining*. 5–14.

[2] Trevor Bench-Capon, Michał Araszkiewicz, Kevin Ashley, Katie Atkinson, Floris Bex, Filipe Borges, Daniele Bourcier, Paul Bourgine, Jack G Conrad, Enrico Francesconi, et al. 2012. A history of AI and Law in 50 papers: 25 years of the international conference on AI and Law. *Artificial Intelligence and Law* 20, 3 (2012), 215–319.

[3] Jaime Carbonell and Jade Goldstein. 1998. The use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proceedings of the 21st annual international ACM SIGIR conference on Research & Development in Information Retrieval*. 335–336.

[4] Ben Carterette and Praveen Chandar. 2009. Probabilistic models of ranking novel documents for faceted topic retrieval. In *Proceedings of the 18th ACM conference on Information and knowledge management*. 1287–1296.

[5] Santiago Castro. 2017. Fast Krippendorff: Fast computation of Krippendorff's alpha agreement measure. https://github.com/pln-fing-udelar/fast-krippendorff.

[6] Danqi Chen, Weizhu Chen, Haixun Wang, Zheng Chen, and Qiang Yang. 2012. Beyond ten blue links: enabling user click modeling in federated web search. In *Proceedings of the fifth ACM international conference on Web search and data mining*. 463–472.

[7] Harr Chen and David R Karger. 2006. Less is more: probabilistic models for retrieving fewer relevant documents. In *Proceedings of the 29th annual international ACM SIGIR conference on Research & Development in Information Retrieval*. 429–436.

[8] C. L Clarke, N Craswell, and I Soboroff. 2009. Overview of the TREC 2009 Web Track. In *Proceedings of The Eighteenth Text REtrieval Conference, TREC 2009, Gaithersburg, Maryland, USA, November 17-20, 2009*.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[10] Yue Feng, Jun Xu, Yanyan Lan, Jiafeng Guo, Wei Zeng, and Xueqi Cheng. 2018. From Greedy Selection to Exploratory Decision-Making: Diverse Ranking with Policy-Value Networks. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval* (Ann Arbor, MI, USA) *(SIGIR '18)*. Association for Computing Machinery, New York, NY, USA, 125–134. https://doi.org/10.1145/3209978.3209979

[11] Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin* 76, 5 (1971), 378.

[12] Veronica Gil-Costa, Rodrygo LT Santos, Craig Macdonald, and Iadh Ounis. 2011. Sparse spatial selection for novelty-based search result diversification. In *International Symposium on String Processing and Information Retrieval*. Springer, 344–355.

[13] Fan Guo, Chao Liu, and Yi Min Wang. 2009. Efficient multiple-click models in web search. In *Proceedings of the second acm international conference on web search and data mining*. 124–131.

[14] Hanjo Hamann. 2019. The German Federal Courts Dataset 1950–2019: From Paper Archives to Linked Open Data. *Journal of Empirical Legal Studies* 16, 3 (2019), 671–688.

[15] Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in neural information processing systems*. 2042–2050.

[16] Kai Hui, Andrew Yates, Klaus Berberich, and Gerard de Melo. 2017. PACRR: A Position-Aware Neural IR Model for Relevance Matching. arXiv:1704.03940 [cs.IR]

[17] Maurice G Kendall. 1938. A new measure of rank correlation. *Biometrika* 30, 1/2 (1938), 81–93.

[18] Sadegh Kharazmi, Mark Sanderson, Falk Scholer, and David Vallet. 2014. Using score differences for search result diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & Development in Information Retrieval*. 1143–1146.

[19] Omar Khattab and Matei Zaharia. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction over BERT. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 39–48. https://doi.org/10.1145/3397271.3401075

[20] Marios Koniaris, Ioannis Anagnostopoulos, and Yannis Vassiliou. 2016. Multi-dimension diversification in legal information retrieval. In *International Conference on Web Information Systems Engineering*. Springer, 174–189.

[21] Marios Koniaris, Ioannis Anagnostopoulos, and Yannis Vassiliou. 2017. Evaluation of diversification techniques for legal information retrieval. *Algorithms* 10, 1 (2017), 22.

[22] Carol Collier Kuhlthau and Stephanie L Tama. 2001. Information search process of lawyers: a call for'just for me'information services. *Journal of documentation* (2001).

[23] D. D. Lewis. 1997. The TREC-5 Filtering Track. (1997).

[24] Haitao Li, Qingyao Ai, Jia Chen, Qian Dong, Yueyue Wu, Yiqun Liu, Chong Chen, and Qi Tian. 2023. SAILER: Structure-aware Pre-trained Language Model for Legal Case Retrieval. arXiv:2304.11370 [cs.IR]

[25] Haitao Li, Weihang Su, Changyue Wang, Yueyue Wu, Qingyao Ai, and Yiqun Liu. 2023. THUIR@COLIEE 2023: Incorporating Structural Knowledge into Pre-trained Language Models for Legal Case Retrieval. arXiv:2305.06812 [cs.IR]

[26] Haitao Li, Changyue Wang, Weihang Su, Yueyue Wu, Qingyao Ai, and Yiqun Liu. 2023. THUIR@COLIEE 2023: More Parameters and Legal Knowledge for Legal Case Entailment. arXiv:2305.06817 [cs.CL]

[27] Shangsong Liang, Zhaochun Ren, and Maarten De Rijke. 2014. Fusion helps diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & Development in Information Retrieval*. 303–312.

[28] Tie-Yan Liu. 2011. Learning to rank for information retrieval. (2011).

[29] Yixiao Ma, Yunqiu Shao, Yueyue Wu, Yiqun Liu, Ruizhe Zhang, Min Zhang, and Shaoping Ma. 2021. LeCaRD: A Legal Case Retrieval Dataset for Chinese Law System. *Information Retrieval (IR)* 2 (2021), 22.

[30] Stephann Makri, Ann Blandford, and Anna L Cox. 2008. Investigating the information-seeking behaviour of academic lawyers: From Ellis's model to design. *Information Processing & Management* 44, 2 (2008), 613–634.

[31] Sun Maosong, Chen Xinxiong, Zhang Kaixu, Guo Zhipeng, and Liu Zhiyuan. 2016. THULAC: An Efficient Lexical Analyzer for Chinese. 2016.. In *Proceedings of EMNLP*.

[32] Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 30.

[33] Filip Radlinski, Robert Kleinberg, and Thorsten Joachims. 2008. Learning diverse rankings with multi-armed bandits. In *Proceedings of the 25th international conference on Machine learning*. 784–791.

[34] Karthik Raman, Pannaga Shivaswamy, and Thorsten Joachims. 2012. Online learning to diversify from implicit feedback. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*. 705–713.

[35] Stephen E Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR'94*. Springer, 232–241.

[36] LT Rodrygo, Craig Macdonald, and Iadh Ounis. 2015. Search result diversification. *Foundations and Trends in Information Retrieval* 9, 1 (2015), 1–90.

[37] Tony Russell-Rose, Jon Chamberlain, and Leif Azzopardi. 2018. Information retrieval in the workplace: A comparison of professional search practices. *Information Processing & Management* 54, 6 (2018), 1042–1057.

[38] Tetsuya Sakai. 2018. α-nDCG. Springer New York, New York, NY, 98–99. https://doi.org/10.1007/978-1-4614-8265-9_80619

[39] Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information processing & management* 24, 5 (1988), 513–523.

[40] Yunqiu Shao. 2020. Towards Legal Case Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research & Development in Information Retrieval*. 2485–2485.

[41] Yunqiu Shao, Haitao Li, Yueyue Wu, Yiqun Liu, Qingyao Ai, Jiaxin Mao, Yixiao Ma, and Shaoping Ma. 2023. An Intent Taxonomy of Legal Case Retrieval. arXiv:2307.13298 [cs.IR]

[42] Yunqiu Shao, Jiaxin Mao, Yiqun Liu, Weizhi Ma, Ken Satoh, Min Zhang, and Shaoping Ma. 2020. BERT-PLI: Modeling Paragraph-Level Interactions for Legal Case Retrieval.. In *IJCAI*. 3501–3507.

[43] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A latent semantic model with convolutional-pooling structure for information retrieval. In *Proceedings of the 23rd ACM international conference on conference on information and knowledge management*. 101–110.

[44] Vu Tran, Minh Le Nguyen, Satoshi Tojo, and Ken Satoh. 2020. Encoded summarization: summarizing documents into continuous vector space for legal case retrieval. *Artificial Intelligence and Law* 28, 4 (2020), 441–467.

[45] Ioannis Tsochantaridis, Thorsten Joachims, Thomas Hofmann, Yasemin Altun, and Yoram Singer. 2005. Large margin methods for structured and interdependent output variables. *Journal of machine learning research* 6, 9 (2005).

[46] Marc Van Opijnen and Cristiana Santos. 2017. On the concept of relevance in legal information retrieval. *Artificial Intelligence and Law* 25 (2017), 65–87.

[47] Shengxian Wan, Yanyan Lan, Jun Xu, Jiafeng Guo, Liang Pang, and Xueqi Cheng. 2016. Match-srnn: Modeling the recursive matching structure with spatial rnn. *arXiv preprint arXiv:1604.04378* (2016).

[48] Jun Wang and Jianhan Zhu. 2009. Portfolio theory of information retrieval. In *Proceedings of the 32nd international ACM SIGIR conference on Research & Development in Information Retrieval*. 115–122.

[49] Miao Wen and Xiangji Huang. 2006. York University at TREC 2006: Legal Track.. In *TREC*.

[50] Chaojun Xiao, Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Zhiyuan Liu, Maosong Sun, Tianyang Zhang, Xianpei Han, Zhen Hu, Heng Wang, et al. 2019. Cail2019-scm: A dataset of similar case matching in legal domain. *arXiv preprint arXiv:1911.08962* (2019).

[51] ChengXiang Zhai and John Lafferty. 2006. A risk minimization framework for information retrieval. *Information Processing & Management* 42, 1 (2006), 31–55.
[52] Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal Judgment Prediction via Topological Learning. In *Proceedings of EMNLP*.
[53] Xiaojin Zhu, Andrew B Goldberg, Jurgen Van Gael, and David Andrzejewski. 2007. Improving diversity in ranking using absorbing random walks. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*. 97–104.
[54] Yadong Zhu, Yanyan Lan, Jiafeng Guo, Xueqi Cheng, and Shuzi Niu. 2014. Learning for search result diversification. In *Proceedings of the 37th international ACM SIGIR conference on Research & Development in Information Retrieval*. 293–302.
[55] Guido Zuccon and Leif Azzopardi. 2010. Using the quantum probability ranking principle to rank interdependent documents. In *European Conference on Information Retrieval*. Springer, 357–369.