Joint Learning for Legal Text Retrieval and Textual Entailment: Leveraging the Relationship between *Relevancy* and *Affirmation*

Hai-Long Nguyen[1], **Thi-Hai-Yen Vuong**[1], **Ha-Thanh Nguyen**[2] and **Xuan-Hieu Phan**[1]
[1] *VNU University of Engineering and Technology*, Hanoi
[2] *National Institute of Informatics*, Tokyo, Japan
{long.nh, yenvth, hieupx}@vnu.edu.vn
nguyenhathanh@nii.ac.jp

Abstract

In legal text processing and reasoning, one normally performs information retrieval to find relevant documents of an input question, and then performs textual entailment to answer the question. The former is about *relevancy* where as the latter is about *affirmation* (or *conclusion*). While relevancy and affirmation are two different concepts, there is obviously a connection between them. That is why performing retrieval and textual entailment sequentially and independently may not make the most of this mutually supportive relationship. This paper, therefore, propose a multi–task learning model for these two tasks to improve their performance. Technically, in the COLIEE dataset, we use the information of Task 4 (conclusions) to improve the performance of Task 3 (searching for legal provisions related to the question). Our empirical findings indicate that this supportive relationship truly exists. This important insight sheds light on how leveraging relationship between tasks can significantly enhance the effectiveness of our multi-task learning approach for legal text processing.

1 Introduction

In legal text processing and reasoning, legal document retrieval and textual entailment are two important tasks. Given an input legal question, the former helps to narrow down and locates a subset of most relevant documents while the latter attempts to give a yes/no answer to the question by analyzing those relevant documents. Legal document retrieval and textual entailment have several challenges, such as intricate legal language structures (Nguyen et al., 2022b), scarcity of annotated legal data (Nguyen et al., 2022a; Yoshioka et al., 2021), rich vocabulary with multiple meanings (Nguyen et al., 2021), and high dependency and the interrelationship between legal statutes (Vuong et al., 2022).

In this paper, we focus on these two tasks within the context of the Competition on Legal Information Extraction/Entailment (COLIEE) (Rabelo et al., 2021b, 2022). In COLIEE, the retrieval task (a.k.a Task 3) involves the retrieval of relevant statutes from a database of Japanese civil code statutes regarding an input yes/no legal question. The textual entailment task (a.k.a Task 4) aims to confirm the entailment of a yes/no answer based on the anlsysis of the retrieved civil code statutes. Traditionally, these two tasks have been addressed independently as two separate steps without considering their interdependence. This formulation approach should ignore a mutually supportive relationship between them.

In this work, our main idea is that the joint and simultaneous learning and inference of these two tasks can make the most of the mutual relationship between *relevancy* and *affirmation*. In some cases, truly relevant statutes that confirm an input question may not relevant in terms of having common vocabulary in traditional information retrieval. Our hypothesis is that this joint and "bi–directional" learning will make more accurate decisions in each task. Especially, legal entailment information (Task 4) can help to improve the performance of the retrieval task (Task 3).

Technically, we propose a multi–task learning framework that integrate and learn Task 3 and Task 4 simultaneously. This should allow the exploitation of shared features and relevant information that may be ignored in the independent and separate training of these tasks. Our method aims to leverage the strengths of the mentioned techniques to deal with the challenges of legal text processing as well as make use of the relationship between *relevancy* and *affirmation* to improve the performance of legal text retrieval and entailment tasks.

The remaining of the paper is organized as follows. Section 2 mentions related work. Section 3 addresses the problem. Section 4 presents our multi–task learning method. The experiments, results, and anslysis are described in Section 5. Finally, Section 6 draws our conclusions.

Proceedings of the Natural Legal Language Processing Workshop 2023, pages 192–201
December 7, 2023 ©2023 Association for Computational Linguistics

2 Related Work

Various methods have been proposed to tackle legal text retrieval and textual entailment in the COLIEE competition. Researchers have identified BM25 as a useful baseline for statute retrieval, and several studies have focused on combining lexical and neural ranking models (Rosa et al., 2021; Askari et al., 2021). With the COLIEE 2020 dataset, Vuong et al. observed an equivalence between the Entailment relation and the relation between the topic sentence and other sentences in a passage. Consequently, the research team employed a weak-labeling method to generate additional data and trained a supporting model. Using this supporting model as a pre-trained model helped the research team achieve the highest performance in tasks 1 and 2 of COLIEE 2020 (Vuong et al., 2023). A Graph-Augmented Dense Statue Retriever (G-DSR) which leverages legislation's structural aspects using a graph neural network surpassed robust retrieval benchmarks when evaluated on a real-world SAR dataset expertly annotated by professionals (Louis et al., 2023). For improving the retrieval performance in task 1 of COLIEE 2021, Abolghasemi et al. used multi–task learning which combined document–level representation learning and ranking objective (Abolghasemi et al., 2022).

For Legal Textual Entailment, approaches involve IR-based systems, BERT ensemble methods, and transformer-based techniques combined with textual similarity (Kim et al., 2019; Rabelo et al., 2021a). Yoshioka et al. employed an systematic method for creating training data for syntatic structure understanding (Yoshioka et al., 2021). Multi–task learning has been employed to overcome data scarcity in the domain, addressing tasks such as translation, summarization, multi–label classification, and legal case retrieval and entailment (Elnaggar et al., 2018).

3 Problem Statement

3.1 Legal Document Retrieval

Legal document retrieval refers to the task of finding relevant legal documents given a query, which could be a legal question or a legal statement. In this work, the set of legal documents that need to be retrieved is the statute law. Let $D = d_1, d_2, ..., d_n$ be a collection of legal documents. Given a query q (i.e., a legal question or statement), this task is to retrieve a subset documents $D_q \subset D$ such that these documents are semantically related to q, that is, they could be used to answer or explain q. The problem can be described using a relevance evaluation function 1. This problem is particularly challenging due to the complexity of legal documents and the vast amount of available legal materials (Ruhl et al., 2017; Katz et al., 2020).

$$f(query, d_i) = \mathbf{R} \tag{1}$$

where $\mathbf{R}$ having two values, 0 and 1, corresponding to relevance (YES_R) and non-relevance (NO_R) respectively.

3.2 Legal Textual Entailment

Legal textual entailment involves the identification of entailment relationship between a set of relevant articles and a given query. Two types of relationship is pre–defined including "*documents entail the query*" and "*documents entail the negative form of the query*". In other words, the goal of this task is to answer the question: *arcording to the relevant documents, whether the query is true or false?*.

Specifically, let q be the input query and D_q be a set of relevant articles of q. The goal is to determine which form of entailment holds: $Entail(D_q, q)$ or $Entail(D_q, \textbf{not } q)$. The problem can be described using expression 1.

$$g(q, D_q) = \mathbf{E} \tag{2}$$

where $\mathbf{E}$ having two values including:

- $Entail(q, D_q)$ denotes YES_T, corresponding to the content of the document set D_q proving the query q to be true.
- $Entail(-q, D_q)$ is denoted as NO_T, corresponding to the content of the document set D_q proving the negation of the query q to be true.

The entailment relationship identification is a complicated task due to several reasons. First, recent NLP deep learning models are mostly based on the semantic similarity between sentences or documents. This approach is not suitable because this task requires the logical analysis and reasoning based on the actual content of the query and documents. Second, the document length is significantly long and the combination of multiple relevant documents in lexical–level can create a large document which exceeds the input limit of normal NLP deep learning model. Therefore, building a model which

can efficiently determines the entailment relationship is challenging and requires logical reasoning that is beyond semantic similarity.

3.3 Legal Outcome-based Retrieval (*LOR*) Problem

When additional navigational information is added at the output stage, the conventional Retrieval problem typically transforms into an Outcome-based Retrieval problem. Addressing the Outcome-based Retrieval problem often yields significantly improved results compared to solely tackling the conventional Retrieval problem, as the additional navigational information could profoundly influences the model's outcome. In this section, we present a methodology to transition the Legal Retrieval problem discussed in Section 3.1 into a Legal Outcome-based Retrieval *(LOR)* problem, with the expectation that addressing the LOR problem will enhance the results of the Retrieval problem.

3.3.1 Decomposing the Textual Entailment Relationship

The essence of the relationship $g(query, D_q)$ signifies that the correctness of the query will be determined based on the synthesized content of all relevant documents $D_q = d_{i_1}, d_{i_2}, ..., d_{i_n}$. However, the task of combining legal texts poses a considerable challenge due to the extensive length of each legal document, which surpasses the capabilities of current mid-level language models. Furthermore, determining the correctness of the query necessitates a logical combination of the content from all relevant legal documents.

Therefore, in this research, we propose simplifying the original entailment relationship using smaller sub-relationships. Specifically, the decomposition process is represented by expression (3).

$$g(q, [a_1, a_2, ..., a_n]) = \mathbf{E} \Leftrightarrow \begin{cases} g(q, a_1) & = \mathbf{E} \\ g(q, a_2) & = \mathbf{E} \\ ... & \\ g(q, a_n) & = \mathbf{E} \end{cases} \quad (3)$$

In this study, we propose *IIA* (Insufficient Information to Answer) label when legal texts cannot definitively answer the query as either yes or no. Therefore, in addition to $\mathbf{E}$ - the Textual Entailment Relationship having two values, YES_T and NO_T, it also includes *IIA*.

The textual-entailment relationships decomposition as described above may lead to cases where the relationships $g(query, a_i)$ become inaccurate in certain situations. The first scenario occurs when, within the set of related legal documents $[d_{i_1}, d_{i_2}, ..., d_{i_n}]$, only a small subset of documents contain content that can answer the query, while the remaining, although related, do not provide usable content to answer the query. Decomposing the relationship according to expression (3) in this case can result in changing the relationship between the query and related but insufficiently informative documents from *IIA* to YES_T or NO_T. The second scenario arises when any two documents, although containing content related to the query, have content that conditionally negates each other in specific cases. Therefore, simplifying the relationship according to expression 3 can lead to a reversal of the actual relationship between the query and each document. Examples of two mentioned scenarios are provided in the Figure 1.

The two scenarios mentioned occur rarely, and their impact on our research approach is insignificant. In our preliminary analysis of the 782 legal articles in the corpus, there are 34 instances that fall into the scenario 2, determined by using key terms indicative of conditional negation within legal language With the scenario 1, the queries that involve more than two related legal documents account for about 21.39% in the dataset. Despite the presence of these scenarios, they have a limited impact on the training phase and the overall effectiveness of our method.

3.3.2 The Transformation of Legal Retrieval into Legal Outcome-based Retrieval

The utilization of outcome-based retrieval often yields better results than conventional retrieval due to the added navigational information, resulting in more focused and targeted searches. Therefore, in this study, we propose the use of Decomposed Textual Entailment information to enhance the performance of the Retrieval phase in finding relevant legal documents.

Through the decomposition of the Entailment relationship, each pair $(query, a_i)$ are not only associated with the Relevant relationship ($\mathbf{R}$) denoted by two labels, YES_R and NO_R, but also with the Textual Entailment relationship ($\mathbf{E}$) featuring three labels, namely YES_T, NO_T, and *IIA*. Based on the observation that legal experts tend to seek legal documents with relevant content to support

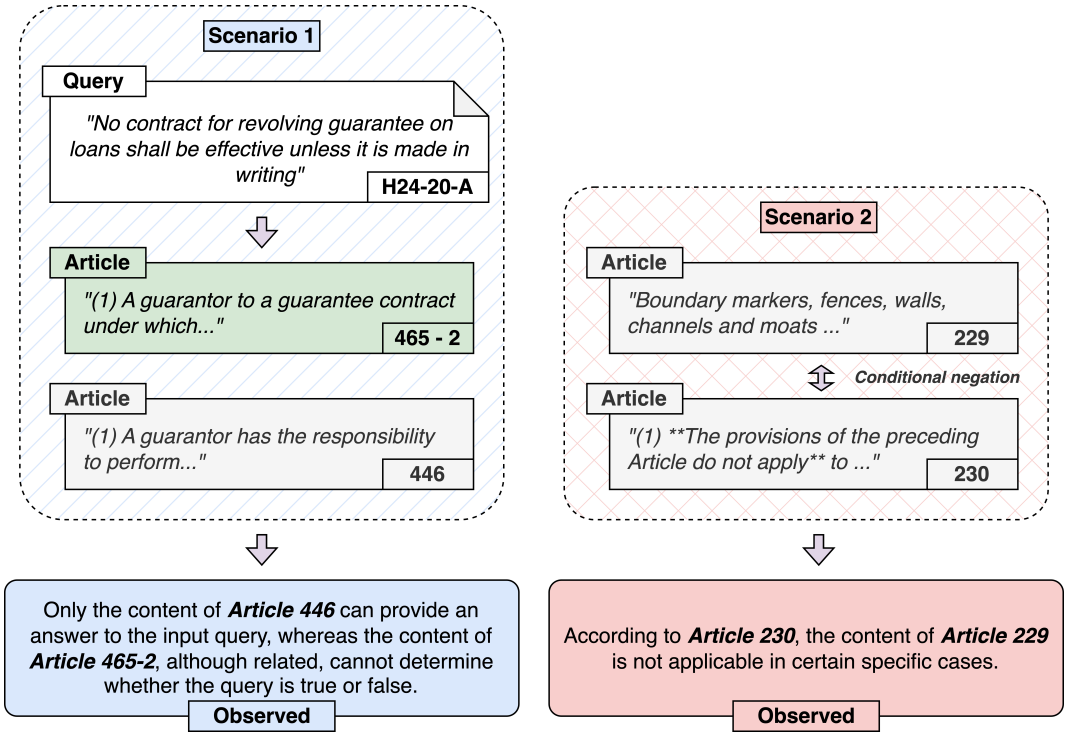

Figure 1: Two scenarios that affect the Textual Entailment Decomposition

their stance on whether a statement/query is true or false, we assume that the correctness of a query plays a crucial role in determining whether a legal document is related to that query or not. With this assumption, the Multi-Task Learning model that combines two streams of information, $\mathbf{R}$ and $\mathbf{E}$ is proposed. Specifically, the Decomposed Textual Entailment relationship is introduced as one of the two outputs of the model, alongside the Relevant relationship. The aim is that the Entailment information will impact the back-propagation process during training phase, allowing the model to learn more effectively and achieve better optimization. The architecture of the proposed Multi-Task model which is present in Figure 2 is based on the BERT architecture, with the addition of two Classification layers that output two corresponding labels for Retrieval and Entailment.

4 Legal Document Retrieval and Textual Entailment with Multi–Task Model

4.1 Pre–ranking Phase by BM25 Model

Due to the large number of documents in the corpus, directly using deep learning models here would consume a significant amount of resources and be inefficient. Therefore, the pre-ranking step is essential to quickly filter candidates from the corpus that are lexically relevant to the query. This speeds up both the training and inference phases without significant information loss. The BM25 model is chosen for its statistical-based approach, fast runtime, and effectiveness in retrieval tasks. Furthermore, the correlation score computed by the BM25 model can also be utilized to integrate with the output of the multi-task model, thereby compensating for the recall aspect of the retrieval system. With a set of document $D = \{d_1, d_2, ..., d_m\}$ and a query q which contains n terms $q = [t_1^q, t_2^q, ..., t_n^q]$, the relevant score between the query q and document d_j computed by BM25 (Equation 6) is the mutiplication of TF score (Equation 5) and IDF score (Equation 4).

$$IDF(t_i^q) = \ln\left(1 + \frac{m - f(t_i^q) + 0.5}{f(t_i^q) + 0.5}\right) \quad (4)$$

$$TF(t_i^q, d_j) = \frac{g(t_i^q, d_j) * (k1 + 1)}{g(t_i^q, d_j) + k1 * [1 - b + b * R(d_j)]} \quad (5)$$

$$score_{q,d_j} = \sum_{i=1}^{n} IDF(t_i^q) * TF(t_i^q, d_j) \quad (6)$$

where:

- $f(t)$ represents the number of documents in D containing the token t.
- $g(t, d)$ represents the frequency of token t occurring in document d.
- $k1$ is a parameter that determine the term frequency saturation which ensures that words

with high occurrence frequencies do not significantly influence the final score.

- b implies the impact of the document length on the final score.

- $R(d) = \frac{lenOf(d)}{averageLen}$ is the ratio between the length of document d and the average length of all documents in D.

4.2 Re–ranking Phase by Multi–Task Model

In the second phase of the retrieval system, the multi-task model is utilized to calculate correlation scores based on the semantics of the query and the legal documents. Specifically, the model adopts a pre-trained model for encoding the semantic of the input sequence and contains two corresponding outputs for the retrieval and textual-entailment tasks. Through this architectural design, the model parameters will be updated based on the information from both loss functions: one originating from the retrieval task and the other from the textual entailment task. In other words, instead of solely basing on their contents for determining whether the query and the related document are relevant, the model will learn how to integrate the information about the correctness of the query and vice versa, determining the correctness of the query based on the content provided in the document. Figure 2 illustrates the detailed architecture of the multi-task model used in the re-ranking phase.

In this architecture, both the query and the legal text are simultaneously input into the model. The output vector representing the special token [CLS] is used as a semantic representation for the query-text pair. Subsequently, this representation vector is passed through two separated linear layers corresponding to each task (retrieval task and textual-entailment task). The retrieval head of the model consists of two nodes representing the outcomes of YES_R or NO_R. Meanwhile, the textual-entailment output of the model comprises three nodes corresponding to the three labels: YES_T, NO_T and *IIA*.

The output calculation formula for each task is described by Equation 7. Supposed that the input sequence is $T = [t_1, t_2, ..., t_n]$. The language model, denoted as the *LM* function, embeds the semantic information of the input sequence into a d-dimensional vector.

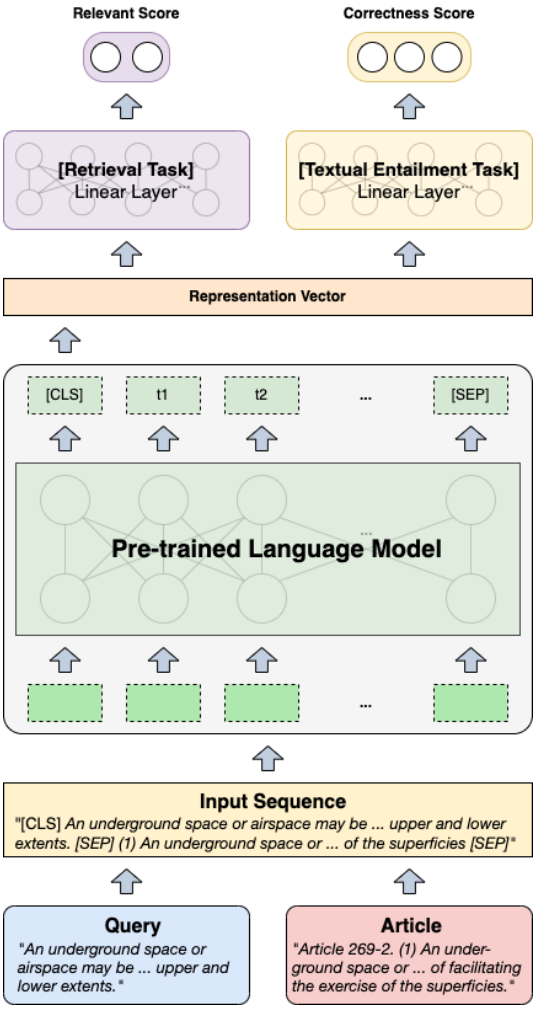

Figure 2: Bert-based multi-task model architecture

$$\begin{aligned} \mathbf{V}_{CLS} &= LM([t_1, t_2, ..., t_n]) \\ \mathbf{H}_{retrieval} &= softmax(\mathbf{V}_{CLS} * \mathbf{W}_R) \\ \mathbf{H}_{entail} &= softmax(\mathbf{V}_{CLS} * \mathbf{W}_T) \end{aligned} \quad (7)$$

where:

- $\mathbf{V}_{CLS} \in \mathcal{R}^d$ is the representation vector of special [CLS] token with d is the hidden dimension of the language model.

- $\mathbf{W}_R \in \mathcal{R}^{d \times 2}$, $\mathbf{W}_T \in \mathcal{R}^{d \times 3}$ are the matrix weights of retrieval head and textual-entailment head, respectively.

- $\mathbf{H}_{retrieval} \in \mathcal{R}^2$ is a vector containing the probabilities of two labels: relevant and irrelevant, for the retrieval task.

- $\mathbf{H}_{entail} \in \mathcal{R}^3$ is a vector containing the probabilities of three labels: irrelevant, entails, and does not entail, for the textual entailment task.

Due to the model having two output heads, there will be two loss functions: one for the retrieval head and one for the textual entailment head. Both loss functions utilize the Cross Entropy formula; however, due to the data imbalance in the retrieval task, the loss function for the retrieval head will be weighted for each label. Equation 8 and 9 represent the loss function for the retrieval head and the textual entailment head, respectively.

$$\mathcal{L}_{re} = -\sum_{c=0}^{1} w_c \log \frac{\exp(pr_c)}{\sum_{i=1}^{c} \exp(pr_i)} yr_c \quad (8)$$

$$\mathcal{L}_{en} = -\sum_{c=0}^{2} \log \frac{\exp(pe_c)}{\sum_{i=1}^{c} \exp(pe_i)} ye_c \quad (9)$$

where:

- $w \in \mathcal{R}^2$ is the weight of each label in retrieval task.
- $pr \in \mathcal{R}^2$, $pe \in \mathcal{R}^3$ are the vector containing the probability for each class of retrieval task and entailment task, respectively.
- $yr \in \mathcal{R}^2$, $ye \in \mathcal{R}^3$ are the one-hot label vector of retrieval task and entailment task, respectively.

Two loss functions $\mathcal{L}_{re}$ and $\mathcal{L}_{en}$ are combined through an addition operation. The equal weighting of the combination reflects the equal importance of the two tasks in model training procedure. The final loss used in training phase is present in Equation 10.

$$\mathcal{L} = \mathcal{L}_{re} + \mathcal{L}_{en} \quad (10)$$

5 Experiments and Results

5.1 COLIEE Dataset

We use two datasets for our evaluation, COLIEE 2022[1] and COLIEE 2023[2], with $F2$ as the main metric (Rabelo et al., 2022). The statue law corpus in both datasets is sourced from the Japanese Civil Code, encompassing a collection of 782 legal provisions. Table 1 provides an overview of basic statistics on the statue law corpus. The statistical analysis shows that the corpus contains a maximum of 655 words for English and 755 words for Japanese. These figures exceed the maximum input length supported by BERT-based models, which may lead to the loss of semantic information when truncating the text during model input.

[1] https://sites.ualberta.ca/~rabelo/COLIEE2022/
[2] https://sites.ualberta.ca/~rabelo/COLIEE2023/

Table 1: Statute Law Corpus Statistic

Statue Law Corpus		
#word per legal article		
	en	jp
Min	1	6
Max	655	755
Average	71.83	93.34

Table 2: Data Statistics

		COLIEE 2022		**COLIEE 2023**	
		Train	**Test**	**Train**	**Test**
# query case		887	109	996	101
# word per query					
Min	en	6	13	6	11
	jp	10	18	10	15
Max	en	149	91	149	87
	jp	202	125	202	109
Average	en	39	42	39	42
	jp	51	52	51	54
# relevant article per query					
Max		6	5	6	2
Average		1.28	1.20	1.28	1.29

The COLIEE 2022 dataset comprises 996 labeled instances, while the COLIEE 2023 dataset consists of 996 labeled instances as well. The dataset includes both English and Japanese versions, where Japanese is the original language of the dataset, and the English dataset is a translation version from Japanese through a translation process. Table 2 provides basic statistics on the COLIEE 2022 and 2023 datasets for both the Japanese and English versions. Based on the table, the maximum number of words in a civil law provision is 655 (for the English version) and 755 (for the Japanese version). This poses challenges for the re-ranking phase with BERT-based models, as their maximum input length is limited to 512 tokens.

5.2 Pre–ranking by BM25 Model

BM25 is employed as the first retrieval phase to extract highly relevant legal documents based on textual similarity. The pre-ranking phase aims to maintain a high recall score with an appropriate number of candidate documents within a reasonable time frame. We experiment with multiple thresholds of 30, 100, 200, and 500 and with English version of the dataset.Table 3 exhibits the statistics of the experimented top-k thresholds and their corresponding recall scores.

Top-k threshold selection balances time and recall score. A low top-k decreases recall scores but reduces retrieval time while a high top-k increases

Table 3: Recall score of corresponding top-k

Top-k candidates	Recall score			
	COLIEE 2022		COLIEE 2023	
	Train	Test	Train	Test
30	0.7590	0.9359	0.7784	0.8465
100	0.8394	0.9482	0.8513	0.9208
200	0.8829	0.9706	0.8926	0.9604
500	0.9441	0.9862	0.9487	0.9801

recall scores at the expense of longer retrieval time. As per Table 3, a top-k value of *30* is used to generate training data for re-ranking model, which helps mitigate data imbalance and reduce training time. For the inference process, a top-k value of *500* optimizes the recall score and minimizes the removal of relevant legal documents in the re-ranking phase.

5.3 Implementation Details

The multi-task model in Section 4.2 is used for filtering candidate documents obtained from the pre-ranking phase. While the pre-ranking phase prioritizes processing time and recall score, the re-ranking phase aims to improve the precision score.

To train this multi-task model, the candidate documents from the BM25 model are pairwise combined with the query, forming document-query sample pairs. Each pair has two labels: a retrieval label (either YES_R or NO_R) and a textual entailment label (YES_T, NO_T, or *IIA*).

The dataset consists of English and Japanese versions, with the English version being machine-translated from the original Japanese. To compare the performance of the model on both versions, the pre-trained *Multilingual BERT* parameters are used to initialize the model's backbone.

In addition, the pre-trained parameters of the *monot5-base-msmarco* (Nogueira et al., 2020) model are utilized, since the task belongs to the retrieval domain. However, as this pre-trained model is entirely in English, the multi-task model with the T5 backbone is only experimented on the English dataset of the COLIEE competition.

To maintain generalization and simplicity in deployment, the SGD optimizer (Mikolov et al., 2011) is used (Keskar and Socher, 2017). Two versions of the model corresponding to each language are trained for a total of 20 epochs.

5.4 Results on Retrieval Tasks with The COLIEE Dataset

With the parameter settings as described in the previous section, the multi-task model based on the pre-trained *Multilingual BERT* and *Mono-T5-base* was experimented with and evaluated on the COLIEE datasets of 2022 and 2023 using the F2-measure. In addition, to compare the effectiveness of the proposed multi-task approach in improving retrieval performance, a single-task models were also experimented. The investigated models include: single-task model using pre-trained Multilingual BERT, multi-task model using pre-trained *Multilingual BERT* - both architectures were tested with English and Japanese datasets. The pre-trained *Mono-T5-base* parameters were also used for evaluating, but only with the English dataset.

Finally, an ensemble process was used to optimize the strengths of the individual models and improve retrieval performance on the private test set. The ensemble method involved combining the relevant score of the BM25 model and the six aforementioned models. Two ensemble strategies were implemented: weighted ensemble and voting ensemble. In the weighted ensemble strategy, the final relevant score will be determined by calculating the weighted sum of each individual model whose weights will be determined through a grid search process on the validation set. Equation 11 represents the formula for combining the retrieval results of all six experimented models.

$$\begin{aligned} \textit{relevant\_score} &= \sum_{i=1}^{n} w_i * s_i \\ s.t &: \sum_{i=1}^{n} w_i = 1 \end{aligned} \tag{11}$$

where:

- n is the total number of models that are ensembled.
- w_i represents the weight of model i.
- s_i represents the relevant score calculated by model i.

In the voting strategy, each model's prediction is treated as a vote. All the votes are counted and the final decision is made based on the majority. If the majority of models predict relevance, the pair of question-legal document is deemed relevant; otherwise, it is considered as non-relevant.

Table 4 and Table 5 respectively present the best results achieved by participating teams in the COLIEE competition of 2022 and 2023, along with the results of the proposed models in this study. The

Table 4: F2-measure of all models and other teams (Kim et al., 2022) evaluated on COLIEE 2022 dataset

Model	F2	P	R
HUKB (Yoshioka et al., 2022)	0.820	**0.818**	0.841
OVGU (Wehnert et al., 2022)	0.779	0.778	0.805
JNLP (Bui et al., 2022)	0.770	0.687	0.838
UA	0.764	0.807	0.764
LLNTU	0.642	0.674	0.639
(1) Single BERT JP	0.786	0.667	**0.897**
(2) Single BERT EN	0.776	0.658	0.879
(3) Single MonoT5 EN	0.783	0.706	0.845
(4) Multi-Task BERT JP	0.827	0.755	**0.897**
(5) Multi-Task BERT EN	0.781	0.716	0.833
(6) MultiTask MonoT5 EN	0.799	0.723	0.871
(7) Voting Ensemble	0.815	0.768	0.856
(8) Weighted Ensemble	**0.829**	0.792	0.865

Table 5: F2-measure of all models and other teams evaluated on COLIEE 2023 dataset

Model	F2	P	R
CAPTAIN	0.757	**0.726**	0.792
JNLP	0.745	0.645	**0.822**
NOWJ	0.727	0.682	0.767
HUKB	0.673	0.628	0.708
LLNTU	0.654	0.733	0.644
UA	0.564	0.621	0.564
(1) Single BERT JP	0.707	0.600	0.807
(2) Single BERT EN	0.579	0.566	0.614
(3) Single MonoT5 EN	0.689	0.627	0.762
(4) Multi-Task BERT JP	0.731	0.670	0.782
(5) Multi-Task BERT EN	0.688	0.623	0.777
(6) MultiTask MonoT5 EN	0.695	0.598	0.797
(7) Voting Ensemble	0.727	0.682	0.767
(8) Weighted Ensemble	**0.773**	0.723	**0.822**

result tables clearly shows that the models trained on the Japanese dataset consistently outperform the models trained on the English dataset. As previously analyzed, the English dataset undergoes translation process, resulting in significant loss of important information. This loss adversely affects the retrieval effectiveness of the models, leading to a significant decrease in performance.

The multi-task model achieves higher F2 scores than single models on the COLIEE 2022 and 2023 datasets. For 2022 dataset, *(4) Multi-Task BERT JP* outperforms *(1) Single-BERT JP* by 4.1%, *(5) Multi-Task BERT EN* by 0.5%, and *(6) Multi-Task MonoT5 EN* by 1.6%. In 2023, *(4) Multi-Task BERT JP* has a higher F2 score by 2.4%, *(5) Multi-Task BERT EN* by 10.9%, and *Multi-Task MonoT5 EN* by 0.6%.

Furthermore, the multi–task model achieves remarkable results when compared to other teams in the competition. Specifically, on the COLIEE 2022 dataset, the *(4) Multi-Task BERT JP* model achieves an F2 score of 0.827, surpassing the highest-scoring team, HUKB2, with an F2 score of 0.820. As for the COLIEE 2023 dataset, the *Multi-Task BERT JP* model achieves an F2 score of 0.731, surpassing the third-place team, HUKB1, with an F2 score of 0.673.

Two ensemble methods, voting ensemble and weighted ensemble, were used to combine retrieval score from six single models (1 to 6). The voting ensemble increased the average precision score but harmed the average recall score, decreasing the overall F2 score. The weighted ensemble achieved F2 scores surpassing the top-performing team on both COLIEE 2022 and 2023 datasets without using large language models. On the 2022 dataset, the F2 score was 0.829, surpassing the top team by 0.9%. On the 2023 dataset, the F2 score was 0.773, surpassing the top team by 1.6%. This shows the effectiveness of the weighted ensemble method in utilizing the strengths of individual models and the importance of ensemble methods for improving retrieval results.

6 Conclusions

In this study, we proposed a multi-task learning framework for legal document retrieval and textual entailment within the COLIEE dataset, introducing Legal Outcome-based Retrieval (LOR) to enhance traditional legal retrieval. The approach decomposes textual entailment relationships and transforms retrieval into an outcome-based problem, leveraging the relationship between relevancy and affirmation. The experimental results highlight the effectiveness of multi-task learning and reveal potential future research directions, such as expanding to other legal domains, languages, and utilizing advanced deep learning models to improve legal document retrieval and textual entailment tasks.

Limitations

Although integrating information from the entailment task significantly improves the effectiveness of retrieval, the results of the entailment task itself are not yet highly accurate. This can be attributed to several possible reasons. The first reason is that determining the correctness of a query is a non-trivial task that requires incorporating the semantics of all relevant documents to reach a final conclusion. However, the proposed model only utilizes the semantic of a single (query, document) pair to

perform this task. Second, in this study, the correctness of the query is determined by a rule stating that if any of the (query, document) pairs have the label as *correct*, the query will be considered *correct*. However, this approach is logically incorrect in many cases where one (query, document) pair is labeled as *correct* while another (query, document) pair is labeled as 'incorrect'. In the future, research on integrating the semantics of all relevant legal documents to determine the correctness of a query will need to be carefully considered and pursued.

Acknowledgements

This work was supported by the AIP challenge funding related with JST, AIP Trilateral AI Research, Grant Number JPMJCR20G4.

Hai-Long Nguyen was funded by the Master, PhD Scholarship Programme of Vingroup Innovation Foundation (VINIF), code VINIF.2022.ThS.050.

References

Amin Abolghasemi, Suzan Verberne, and Leif Azzopardi. 2022. Improving bert-based query-by-document retrieval with multi-task optimization. In *European Conference on Information Retrieval*, pages 3–12. Springer.

AA Askari, SV Verberne, O Alonso, S Marchesin, M Najork, and G Silvello. 2021. Combining lexical and neural retrieval with longformer-based summarization for effective case law retrieva. In *Proceedings of the second international conference on design of experimental search & information REtrieval systems*, pages 162–170. CEUR.

MQ Bui, C Nguyen, DT Do, NK Le, DH Nguyen, and TTT Nguyen. 2022. Using deep learning approaches for tackling legal's challenges (coliee 2022). In *Sixteenth International Workshop on Juris-informatics (JURISIN)*.

Ahmed Elnaggar, Christoph Gebendorfer, Ingo Glaser, and Florian Matthes. 2018. Multi-task deep learning for legal document translation, summarization and multi-label classification. In *Proceedings of the 2018 Artificial Intelligence and Cloud Computing Conference*, pages 9–15.

Daniel Martin Katz, Corinna Coupette, Janis Beckedorf, and Dirk Hartung. 2020. Complex societies and the growth of the law. *Scientific reports*, 10(1):18737.

Nitish Shirish Keskar and Richard Socher. 2017. Improving generalization performance by switching from adam to sgd. *arXiv preprint arXiv:1712.07628*.

Mi-Young Kim, Juliano Rabelo, and Randy Goebel. 2019. Statute law information retrieval and entailment. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, pages 283–289.

Mi-Young Kim, Juliano Rabelo, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2022. Coliee 2022 summary: Methods for legal document retrieval and entailment. In *JSAI International Symposium on Artificial Intelligence*, pages 51–67. Springer.

Antoine Louis, Gijs van Dijck, and Gerasimos Spanakis. 2023. Finding the law: Enhancing statutory article retrieval via graph neural networks. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2761–2776, Dubrovnik, Croatia. Association for Computational Linguistics.

Tomas Mikolov, Stefan Kombrink, Anoop Deoras, Lukar Burget, and Jan Cernocky. 2011. Rnnlm-recurrent neural network language modeling toolkit. In *Proc. of the 2011 ASRU Workshop*, pages 196–201.

Ha-Thanh Nguyen, Minh-Phuong Nguyen, Thi-Hai-Yen Vuong, Minh-Quan Bui, Minh-Chau Nguyen, Tran-Binh Dang, Vu Tran, Le-Minh Nguyen, and Ken Satoh. 2022a. Transformer-based approaches for legal text processing: Jnlp team-coliee 2021. *The Review of Socionetwork Strategies*, 16(1):135–155.

Ha-Thanh Nguyen, Manh-Kien Phi, Xuan-Bach Ngo, Vu Tran, Le-Minh Nguyen, and Minh-Phuong Tu. 2022b. Attentive deep neural networks for legal document retrieval. *Artificial Intelligence and Law*, pages 1–30.

Ha-Thanh Nguyen, Vu Tran, Phuong Minh Nguyen, Thi-Hai-Yen Vuong, Quan Minh Bui, Chau Minh Nguyen, Binh Tran Dang, Minh Le Nguyen, and Ken Satoh. 2021. Paralaw nets–cross-lingual sentence-level pretraining for legal text processing. *arXiv preprint arXiv:2106.13403*.

Rodrigo Nogueira, Zhiying Jiang, Ronak Pradeep, and Jimmy Lin. 2020. Document ranking with a pre-trained sequence-to-sequence model. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 708–718, Online. Association for Computational Linguistics.

Juliano Rabelo, Randy Goebel, Mi-Young Kim, Yoshinobu Kano, Masaharu Yoshioka, and Ken Satoh. 2022. Overview and discussion of the competition on legal information extraction/entailment (coliee) 2021. *The Review of Socionetwork Strategies*, 16(1):111–133.

Juliano Rabelo, Mi-Young Kim, and Randy Goebel. 2021a. The application of text entailment techniques in coliee 2020. In *New Frontiers in Artificial Intelligence: JSAI-isAI 2020 Workshops, JURISIN, LENLS 2020 Workshops, Virtual Event, November 15–17,*

2020, Revised Selected Papers 12, pages 240–253. Springer.

Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2021b. Coliee 2020: methods for legal document retrieval and entailment. In *New Frontiers in Artificial Intelligence: JSAI-isAI 2020 Workshops, JURISIN, LENLS 2020 Workshops, Virtual Event, November 15–17, 2020, Revised Selected Papers 12*, pages 196–210. Springer.

Guilherme Moraes Rosa, Ruan Chaves Rodrigues, Roberto Lotufo, and Rodrigo Nogueira. 2021. Yes, bm25 is a strong baseline for legal case retrieval. *arXiv preprint arXiv:2105.05686*.

JB Ruhl, Daniel Martin Katz, and Michael J Bommarito. 2017. Harnessing legal complexity. *Science*, 355(6332):1377–1378.

Yen Thi-Hai Vuong, Quan Minh Bui, Ha-Thanh Nguyen, Thi-Thu-Trang Nguyen, Vu Tran, Xuan-Hieu Phan, Ken Satoh, and Le-Minh Nguyen. 2022. Sm-bert-cr: a deep learning approach for case law retrieval with supporting model. *Artificial Intelligence and Law*, pages 1–28.

Yen Thi-Hai Vuong, Quan Minh Bui, Ha-Thanh Nguyen, Thi-Thu-Trang Nguyen, Vu Tran, Xuan-Hieu Phan, Ken Satoh, and Le-Minh Nguyen. 2023. Sm-bert-cr: a deep learning approach for case law retrieval with supporting model. *Artificial Intelligence and Law*, 31(3):601–628.

Sabine Wehnert, Libin Kutty, and Ernesto William De Luca. 2022. Using textbook knowledge for statute retrieval and entailment classification. In *JSAI International Symposium on Artificial Intelligence*, pages 125–137. Springer.

Masaharu Yoshioka, Yasuhiro Aoki, and Youta Suzuki. 2021. Bert-based ensemble methods with data augmentation for legal textual entailment in coliee statute law task. In *Proceedings of the eighteenth international conference on artificial intelligence and law*, pages 278–284.

Masaharu Yoshioka, Youta Suzuki, and Yasuhiro Aoki. 2022. Hukb at the coliee 2022 statute law task. In *JSAI International Symposium on Artificial Intelligence*, pages 109–124. Springer.