PROVBENCH: A Benchmark of Legal Provision Recommendation for Contract Auto-Reviewing

Xiuxuan Shen[1] **Zhongyuan Jiang**[1,5*] **Junsan Zhang**[2] **Junxiao Han**[3*] **Yao Wan**[4]
Chengjie Guo[1] **Bingcheng Liu**[1] **Jie Wu**[2] **Renxiang Li**[1] **Philip S. Yu**[6]
[1] Xidian University, [2] China University of Petroleum (East China),
[3] School of Computer and Computing Science, Hangzhou City University,
[4] Huazhong University of Science and Technology, [5] Purple Mountain Laboratories
[6] University of Illinois Chicago
shenxiuxuan@stu.xidian.edu.cn, zyjiang@xidian.edu.cn, hanjx@hzcu.edu.cn

Abstract

Contract review is a critical process to protect the rights and interests of the parties involved. However, this process is time-consuming, labor-intensive, and costly, especially when a contract faces multiple rounds of review. To accelerate the contract review and promote the completion of transactions, this paper introduces a novel benchmark of legal provision recommendation and conflict detection for contract auto-reviewing (PROVBENCH), which aims to recommend the legal provisions related to contract clauses and detect possible legal conflicts. Specifically, we construct the first Legal Provision Recommendation Dataset: PROVDATA, which covers 8 common contract types. In addition, we conduct extensive experiments to evaluate PROVBENCH on various state-of-the-art models. Experimental results validate the feasibility of PROVBENCH and demonstrate the effectiveness of PROVDATA. Finally, we identify potential challenges in the PROVBENCH and advocate for further investigation[1].

1 Introduction

Contract review (Hermalin et al., 2007) is an essential step in completing a transaction, where parties carefully assess each contract clause to ensure compliance, fairness, and effective risk management (Wang and Chen, 2022; Schuhmann and Eichhorn, 2015). However, contract review is typically labor-intensive, time-consuming, and costly (Zheng et al., 2020). For enterprises that manage numerous contracts with multiple review rounds, it is necessary to hire legal professionals and commit significant time, leading to a substantial increase in financial pressure (Hu et al., 2018).

To alleviate this issue, various methods (Leivaditi et al., 2020; Wang et al., 2023) have been developed to improve efficiency in contract review through tasks such as classifying the contract clause (Lippi et al., 2019) or assessing their importance levels (Hendrycks et al., 2021). These methods primarily focus on analyzing contracts while ignoring their association with relevant legal provisions, which limits their ability to provide clear evidence or specific recommendations to ensure legal compliance.

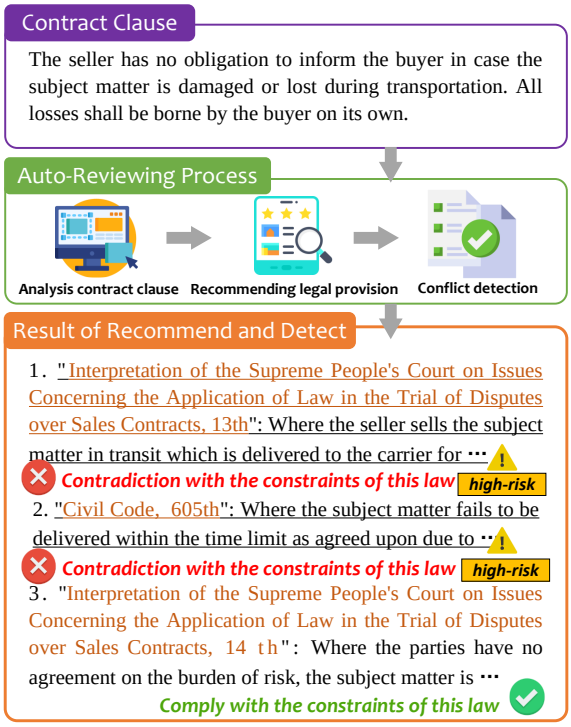

Figure 1: An illustration of contract review: recommending relevant legal provisions as clues and detecting potential conflicts.

In this paper, we aim to bridge this gap by proposing a framework that not only provides explicit legal provisions as evidence but also evaluates potential legal contradictions. As shown in Figure 1, the process begins with the input of a contract clause for review. Then, it is analyzed to recommend relevant legal provisions, followed by detecting whether logical contradictions exist between the clause and the recommended provisions.

*Corresponding authors.
[1] https://github.com/labpaper/ProvBench

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6240–6254
July 27 - August 1, 2025

The results of recommendation and contradiction detection are returned to the lawyer for assistance in contract review.

Given the difficulty of understanding contract clauses and accurately completing the above process, this paper introduces a novel task for legal provision recommendation in the context of contract auto-reviewing: PROVREC, aimed at mitigating the limitations of existing methods.

To begin formulating and benchmarking the PROVREC task, a primary challenge lies in the absence of a dataset that associates contract clauses with relevant legal provisions through logical relationships. Inspired by the MSCOCO (Lin et al., 2014) dataset, which provides a set of textual descriptions for each image (Lee et al., 2018; Wu et al., 2021), we align each contract clause with relevant legal provisions and specified logical relationships, to construct the first Legal Provision Recommendation Dataset (PROVDATA).

In the PROVDATA, we manually assign each contract clause with a set of relevant legal provisions. Similar to the textual entailment task (Parikh et al., 2016; Pàmies et al., 2023), we consider each contract clause and its corresponding legal provisions as text pairs, labeling them as "entailment" if the provision supports the clause or "contradiction" if it conflicts. This manual annotation process spanned over 4 months, resulting in 3,550 contract clauses and more than 24,850 annotations. This provides a high-quality dataset that establishes a critical benchmark for advancing research in contract auto-reviewing.

Furthermore, we establish PROVBENCH as a benchmark for the PROVREC task, which includes three parts: (1) *Legal Text Learner*, (2) *Legal Provision Recommender*, and (3) *Conflict Detector*. The legal provision recommender recommends relevant provisions for contract clauses, while the conflict detector evaluates the logical relationships to identify potential conflicts. Besides, to benchmark the PROVBENCH under different learning paradigms, we explore two types of strategies: (1) supervised learning methods, where models are trained on PROVDATA dataset, and (2) zero-shot methods, including retrieval-based and prompting-based approaches that require no additional training.

To conclude, the main contributions of this paper can be summarized as follows:

- We propose a novel benchmark PROVBENCH for legal provision recommendation and conflict detection, assisting contract review by providing clear legal support or refuting evidence.
- We construct the PROVDATA dataset, consisting of 3,550 contract clauses and over 24,850 annotations, providing a solid data foundation for benchmarking.
- We benchmark several models for PROVBENCH and conduct extensive experiments to evaluate their performance.

We believe this benchmark provides a valuable tool for contract auto-reviewing. It offers clearer legal references, which enhance efficiency and accuracy in the review process.

2 Problem Formulation

Given a set $C = \{C_1, C_2, \ldots, C_N\}$ consisting of N contract clauses, the goal of PROVBENCH is to assist in recommending clause C_i with relevant legal provisions from candidate legal provisions $L = \{L_1, L_2, \ldots, L_M\}$ and detecting potential conflicts. This involves two key steps:

Legal Provision Recommendation. Empirically, this step recommends the top 3 most relevant legal provisions from L based on semantic similarity to the input contract clause.

> ▶**Input:** A contract clause C_i and candidate legal provisions L.
> ▶**Process:** Recommend 3 legal provisions $L^i = \{L^i_1, L^i_2, L^i_3\}$ by measuring semantic similarity between C_i and each L_j from L.
> ▶**Output:** A ranked set of 3 recommend legal provisions $L^i = \{L^i_1, L^i_2, L^i_3\}$ that are relevant to C_i.

Conflict Detection. Once the relevant legal provisions are recommended, the next step is to evaluate the logical relationship between the contract clause C_i and recommended legal provision L^i by checking for potential conflicts. The output consists of recommended legal provisions and corresponding logical labels $label^i_j$, indicating whether the contract clause entails or contradicts each provision. The process can be formulated as:

> ▶**Input:** A contract clause C_i and a set of recommended legal provisions L^i.
> ▶**Process:** For each L^i_j, determine the logical relationship between C_i and L^i_j by checking if C_i conforms to or contradicts L^i_j.
> ▶**Output:** A set of binary labels $label^i_j \in \{0, 1\}$, where 0 indicates entailment (no conflict), and 1 indicates contradiction.

3 Dataset: PROVDATA

3.1 Data Construction

Constructing a dataset that associates contract clauses with relevant legal provisions and logical relationships is a challenging task. Unlike datasets that can be constructed automatically, ours is entirely built through manual annotation by 5 annotators trained in legal data processing, each with extensive experience in labeling large-scale legal and contractual texts. Such manual construction requires significant human effort and time.

Concretely, we first focus on the Chinese legal scenario and gather a collection of publicly available Chinese contracts through web search, primarily using Baidu [2] and from the Contract Demonstration Text Library[3], covering 8 common contract types: (1) *sales*, (2) *lease*, (3) *technology*, (4) *service*, (5) *loan*, (6) *donation*, (7) *transportation*, and (8) *intellectual property*. From the collected contracts, we select clauses that are typically associated with elevated legal risk or demand closer scrutiny during contract review. A subset of suboptimal clauses is manually revised to enhance quality and enrich the coverage of representative scenarios. Subsequently, we construct a legal provision library based on 5 legal codes, including 663 legal provisions relating to contract, as shown in Table 1.

Data Expansion To expand the contract clause data, we employ ChatGPT-4o (OpenAI, 2024) to generate synthetic data using prompt-based method. The final dataset comprises approximately 60% manually contract clauses and 40% synthetic contract clauses.

▶**Definition**: Generate contract clauses with potential contradictions based on specified violation scenarios.
▶**Input**: Clauses that provide for penalty, force majeure or extension conditions, and vague definitions of "reasonable delay" and penalty clauses.
▶**Output**: 即使交付期延迟,标的物的价格不受政府定价波动的影响,买受人仍应按照合同约定的价格支付款项,出卖人不承担价格调整风险.
▶**Output translate to English**: Even if the delivery period is delayed, the price of the subject matter is not affected by fluctuations in government pricing. The buyer should still pay the amount according to the price stipulated in the contract. The seller does not bear the risk of price adjustment.

As shown in the above prompt, we select a scenario with potential contradiction as a representative case, annotators construct the prompt by specifying relevant clauses and potential conflicts based on common contractual contexts.

Additionally, the expanded data are manually evaluated and filtered for legal logic validity, adherence to contract drafting conventions, and clarity in the expression of contradictions, ensuring that the retained data are logically sound, legally compliant, and semantically clear. The process mentioned above effectively expands the dataset while maintaining high annotation quality.

Legal Code	Num
The Contract section of Civil Code of the People's Republic of China.	505
Copyright Law of the People's Republic of China.	55
Several Opinions on the People's Courts' Handling of Loan Cases.	33
The Interpretation of the Supreme People's Court on the Applicable Law in the Trial of Disputes over Sale Contracts.	46
The Supreme People's Court's Interpretation on Several Issues concerning the Application of Law in the Trial of Cases Involving Disputes over Sales Contracts of Commodity Houses.	24

Table 1: Our collected legal provision library.

Data Annotation. First, each item in the Legal Provision Library is annotated with an average of 4 labels: (1) *The name of the legal code*, (2) *The serial number within the legal code*, (3) *Applicable contract type*, (4) *Applicable contractual element type*. Then, for each contract clause, the top 3 relevant provisions are selected, and their logical labels are annotated, resulting in an average of 7 labels per clause. The complexity of legal texts and the demand for precise semantic interpretation result in each manual annotation requiring thoughtful consideration. The structure of the annotated data is as follows:

▶**Data Structure Example:**
contract clause: C_i,
for $j \in \{1, 2, 3\}$:{
  law$_j$: {
    legal provision: L_j^i,
    conflict label: $label_j^i$,
    legal label: law_j^i,
    legal category: $cate_j^i$,
    legal provision number: num_j^i,
  }
}

Furthermore, to enhance future scalability in contract review automation, the annotators also categorized the legal provisions with 15 tags based on contract elements, including *subject, quality, price, delivery and performance, obligations, con-*

[2] https://www.baidu.com
[3] https://htsfwb.samr.gov.cn/

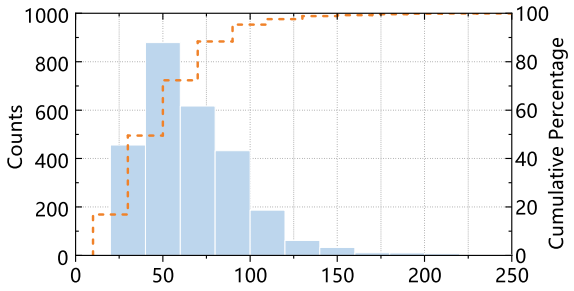

Figure 2: Length statistics of contract clauses.

fidentiality, breach, interest, termination, validity, amendments and transfer, parties, property, litigation, and guarantees.

To conclude, the construction of PROVDATA dataset includes 24,850 manual annotations. It required significant effort, with 5 annotators dedicating approximately 4 months to complete this process. We believe that PROVDATA dataset can provide a valuable data foundation for contract auto-reviewing.

Besides, detailed annotation procedures and data examples are outlined in the Appendix A.

3.2 Data Quality Assessment

To ensure the quality and consistency of annotations in the PROVDATA dataset, we use a cross-validation approach. Initially, each annotator independently labels each contract clause and its corresponding legal provision. We then randomly sample 10% of the annotated pairs and assign them to a second set of annotators, who verify the relevance and logical relationship (entailment or contradiction) between the pairs. If discrepancies arise between the original and second annotations, a third party facilitates a discussion to resolve the inconsistencies. This process ensures the accuracy and logical consistency of the annotations, which is essential for high-quality contract review.

3.3 Data Statistics

Length Statistics of Contract Clause. Figure 2 presents the length distribution of contract clauses in the PROVDATA training set. As shown, fewer than 10% of the clauses exceed 128 words, indicating that the vast majority of clauses fall well within this threshold. Based on this observation, we set the input sequence length to 128 in subsequent experiments to ensure adequate coverage while maintaining computational efficiency.

Statistic	Train	Val	Test
Contract clause	2,698	404	448
Contradiction	2,797	589	568
Entailment	5,297	704	776

Table 2: Statistics of the PROVDATA.

Train, Validation, and Test Splits. The basic statistics of each split in the PROVDATA are shown in Table 2. Following Krishna et al. (2024), we split the data roughly into 75% for training, 12.5% for validation, and 12.5% for testing.

4 Benchmarking: PROVBENCH

As shown in Figure 3, we first adopt a Legal Text Learner to capture the legal features of contract clauses and legal provisions. Next, we compute the interaction information between each contract clause and all legal provisions and use cosine similarity (Lee et al., 2018; Wu et al., 2021; Fu et al., 2023) to rank the relevance of each contract clause with the legal provisions. After filtering out irrelevant legal provisions, we assess the logical relationship between the relevant legal provisions and the contract clause, ultimately providing a clear legal basis for reviewing the contract clause and detecting any potential legal conflicts.

Legal Text Learner. Our input consists of two components: the text of the contract clauses C and legal provisions L. Initially, we load vocabulary and generate token sequences for both C and L based on vocabulary. Then, C and L are encoded into the legal feature $\mathbf{C}$ and $\mathbf{L}$. The process is formulated as:

$$\begin{cases} \mathbf{C} = f_{encoder}(C) \\ \mathbf{L} = f_{encoder}(L) \end{cases} \quad (1)$$

where $f_{encoder}(\cdot)$ refers to the Text Encoder, $\mathbf{C}$ and $\mathbf{L}$ represent the feature representations of the complex legal semantics of C and L, which will be used in subsequent tasks such as legal provision recommendation and conflict detection.

Legal Provision Recommender. Following previous approaches (Lee et al., 2018; Wu et al., 2021), we begin by calculating the interaction between the $\mathbf{C}$ and the $\mathbf{L}$ using a variant of Dot Product Attention (Luong, 2015). The core of this module lies in enhancing the focus on relevant legal provisions from the perspective of the contract clause, which is achieved by optimizing the weights of the legal provision features based on the characteristics

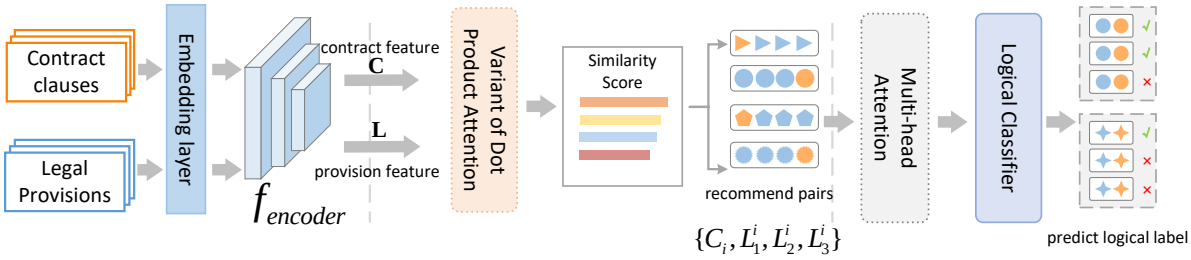

Figure 3: An overview of the PROVBENCH.

of the current contract clause. An extended ablation study on the effectiveness of this module is presented in Appendix B.

Subsequently, the cosine similarity is used to measure the semantic relationship for each $\mathbf{C}_i$ and $\mathbf{L}_j$ as follows:

$$s(\mathbf{C}_i, \mathbf{L}_j) = \frac{\mathbf{C}_i^T \cdot \mathbf{L}_j}{||\mathbf{C}_i|| \cdot ||\mathbf{L}_j||} \quad (2)$$

where $s(\mathbf{C}_i, \mathbf{L}_j)$ is cosine similarity score. Then, the top 3 results correspond to the highest similarity values are selected as the recommended legal provisions $\{\mathbf{L}_1^i, \mathbf{L}_2^i, \mathbf{L}_3^i\}$ associated with the current contract clause $\mathbf{C}_i$.

Conflict Detector. We model the conflict detector by treating each $\mathbf{C}_i$ and relevant legal provisions $\{\mathbf{L}_1^i, \mathbf{L}_2^i, \mathbf{L}_3^i\}$ recommended by Legal Provision Recommender as 3 recommend pairs$(\mathbf{C}_i, \mathbf{L}_j^i)$, where $j \in \{1, 2, 3\}$.

To capture the logical interaction between a contract clause and its corresponding legal provision, we apply a Multi-head Attention Mechanism (Vaswani et al., 2017). Concretely, the legal provision $\mathbf{L}_j^i$ is used as the query Q, while the contract clause $\mathbf{C}_i$ serves as both the key K and value V. Since the two representations are independently generated by the legal text learner, attention facilitates their alignment across multiple semantic subspaces, enabling the conflict detector to learn more diverse logical relationships between the clause and the provision.

$$\hat{y} = \text{Softmax}\left(W_{logic} \cdot \text{Pool}(\text{Multi}(\mathbf{C}_i, \mathbf{L}_j^i)\right) \quad (3)$$

where W_{logic} represents the weight matrix of the logical classifier, $\hat{y}$ indicates whether the relationship is "entailment" or "contradiction", helping to detect logical conflicts between $\mathbf{C}_i$ and $\mathbf{L}_j^i$.

Model Learning. Following previous approaches related to similarity matching (Kiros et al., 2014; Karpathy and Fei-Fei, 2015), we adopt the hinge-based triplet ranking loss to optimize the legal provision recommender as follows:

$$loss(\mathbf{C}, \mathbf{L}) = \sum_{\widehat{\mathbf{L}}} [\alpha - \text{S}(\mathbf{C}, \mathbf{L}) + \text{S}(\mathbf{C}, \widehat{\mathbf{L}})]_+ + \sum_{\widehat{\mathbf{C}}} [\alpha - \text{S}(\mathbf{C}, \mathbf{L}) + \text{S}(\widehat{\mathbf{C}}, \mathbf{L})]_+ \quad (4)$$

where $\widehat{\mathbf{L}}$ and $\widehat{\mathbf{C}}$ refers to negative examples, $[x]_+ \equiv max(x, 0)$ and the margin α is set to 0.2. Through the above loss function, the relevance of matching contract clauses and legal provision pairs is maximized, while the relevance of non-matching pairs is minimized.

5 Experimental Evaluation

5.1 Benchmarked Models

▷**Mamba** (Gu and Dao, 2023) As a State Space Model (SSM) based model, Mamba delivers excellent performance with lower computational complexity and high efficiency. (Wang et al., 2024; Patro and Agneeswaran, 2024). In this study, we apply Mamba to the PROVBENCH and set the SSM state expansion factor to 16, local convolution width to 4, and block expansion factor to 2.

▷**BERT** (Devlin et al., 2019) based models are widely recognized in NLP for their exceptional performance. Therefore, we evaluate our experiments on BERT (bert-base-chinese[4]), Legal-BERT (Chalkidis et al., 2020), and RoBERTa (Liu, 2019) provided by Zhao et al. (2019), to capture the complex semantic features of legal texts. In addition, we employ LaBSE (Feng et al., 2022) as a zero-shot baseline for the legal provision recom-

[4] https://huggingface.co/google-bert/bert-base-chinese

mendation task, evaluating its performance without task-specific fine-tuning.

▷**T5** (Raffel et al., 2020) is a pre-trained language model trained on a multilingual corpus and designed to handle NLP tasks within a unified framework. We conduct experiments on legal text feature learning using T5-base as the legal text encoder and observe strong performance after fine-tuning.

▷**ChatGPT-o1.** (OpenAI, 2024) Recent advancements in Large Language Models (LLMs), particularly ChatGPT-o1, have shown exceptional ability in reasoning through complex texts. Therefore, we are motivated to use ChatGPT-o1 as the conflict detector in a zero-shot manner via prompt-based inference.

▷**DeepSeek-R1.** (DeepSeek-AI, 2025) As a recently released LLMs, DeepSeek-R1 demonstrates strong capabilities in text understanding and reasoning. We employ it for conflict detection in a zero-shot setting via API-based prompt inference.

▷**BM25** (Robertson et al., 1995; Crestani et al., 1998) is a probabilistic ranking model based on term frequency and inverse document frequency, designed to estimate document relevance in information retrieval tasks.

5.2 Evaluation Metrics

Inspired by Lee et al. (2018), we adopt Recall at K (R@K) metrics, specifically R@1, R@3, and R@5, to evaluate whether at least one relevant legal provision appears among the top 1, top 3, and top 5 results. Furthermore, motivated by Oh et al. (2022), we use the Exact Match (EM) metric, denoted as Top-3 EM in this paper, to evaluate whether the top-3 recommended legal provisions match the gold set exactly.

Additionally, in the conflict detection task, we evaluate the accuracy of both entailment and contradiction classes separately and the overall F1 score. To ensure stable and targeted evaluation, we utilize the gold-standard top-3 legal provisions as the reference for computation, rather than relying on model-predicted results. This design facilitates more consistent supervision and enables more effective optimization during the training phase. This multi-dimensional evaluation offers a rigorous assessment of both recommendation and contradiction detection capabilities.

5.3 Implementation Details

Our model is implemented on Ubuntu 24.04 and trained using 2 NVIDIA GeForce RTX 4090 GPU (24GB) with CUDA 11.7. We train for 80 epochs with a batch size of 32. Adam is used as the optimizer with a learning rate of 5×10^{-5}. The input sequence length is set to 128, with embedding dimensions of 512 for Mamba and 384 for BERT-based models.

TextEncoder	R@1	R@3	R@5	Top-3 EM
T5	86.38	95.09	97.10	60.94
BERT	90.63	96.86	98.21	**76.79**
RoBERTa	89.96	95.31	97.54	70.31
LegalBERT	**92.19**	97.77	98.88	73.88
Mamba	91.29	**97.99**	**99.33**	57.37

Table 3: Experimental results of different text encoders on the Legal Provision Recommendation

5.4 Results Analysis

Performance on Legal Provision Recommend. As shown in Table 3, all supervised learning text encoders achieve R@1 scores above 85% and R@5 scores above 97%, demonstrating their ability to retrieve relevant provisions with high recall. The BERT-based models yield Top-3 EM scores exceeding 70%, indicating a relatively stronger capacity to identify accurate provision sets within the top-ranked results. This performance may be attributed to their pre-training on Chinese or legal-domain corpora, which likely enhances their alignment with domain-specific linguistic patterns and legal reasoning structures.

TextEncoder	Accuracy		F1
	Entailment	Contradictory	
T5	97.88	90.43	90.21
BERT	98.16	**91.62**	**90.81**
RoBERTa	97.68	91.36	90.51
LegalBERT	**98.20**	90.99	90.20
Mamba	98.06	89.96	90.17

Table 4: Experimental results of different text encoders on the Conflict Detection

Performance on Conflict Detect. As shown in Table 4, all evaluated models achieve entailment accuracy above 97%, contradictory accuracy above 89%, and overall F1 scores exceeding 90%, demonstrating the feasibility of applying supervised text encoders to potential conflict detection. BERT-based models exhibit comparatively higher accuracy on both entailment and contradiction cases,

Model	R@1	R@3	R@5	Top-3 EM
BM25	**50.89**	**67.63**	**75.22**	**3.57**
LaBSE	42.63	61.83	70.09	0.89

Table 5: Experimental results of different zero-shot baselines on the Legal Provision Recommendation.

Model	Accuracy		F1
	Entailment	Contradictory	
ChatGPT-o1	**81.44**	62.15	72.98
DeepSeek-R1	80.54	**70.60**	**75.66**

Table 6: Experimental results of different zero-shot baselines on the Conflict Detection.

Figure 4: Visualization of the weight distribution of legal provisions.

which may be attributed to their pre-training on Chinese or legal-domain corpora. The generally lower performance on conflict detection may be attributed to the greater logical complexity of the task and the relatively lower proportion of contradiction instances in the dataset, which may limit the ability of models to distinguish this class accurately.

5.5 Results on Zero-shot Baseline

To assess the applicability of our proposed PROVBENCH without relying on task-specific training, we introduce zero-shot baselines for both legal provision recommendation and conflict detection. These baselines are used to evaluate task performance without any additional training, relying solely on direct inference.

Zero-shot for Legal Provision Recommendation. As shown in Table 5, BM25 and LaBSE achieve R@5 scores of 75.22% and 70.09%, respectively. This suggests that the legal provision recommendation task demonstrates a certain level of feasibility when approached with these models. However, their Top-3 EM scores remain comparatively low, with both models scoring below 5%. This indicates that while the models can retrieve partially relevant legal provisions, accurately identifying the complete and correct set remains challenging. This limitation can be attributed to the lack of domain-specific adaptation, as these models have not been trained on Chinese legal corpora and are not optimized for the semantic alignment between contract clauses and legal provisions.

Zero-shot for Conflict Detection. We also evaluate the performance of ChatGPT-o1 and DeepSeek-R1 as the conflict detector. Unlike our constructed conflict detector, ChatGPT-o1 and DeepSeek-R1 directly process the original legal text without learning domain-specific features. As shown in Table 6, ChatGPT-o1 achieves an entailment accuracy of 81.44%, a contradictory accuracy of 62.15%, and an F1 score of 72.98%. DeepSeek-R1 achieves an entailment accuracy of 80.54%, a contradictory accuracy of 70.60%, and an F1 score of 75.66%. These results highlight the potential of large language models to detect contradictions without legal text pre-training. However, their performance remains lower than that of our trained conflict detector, highlighting the effectiveness of domain-specific pre-training on legal corpora for improving contradiction detection. Besides, the prompt template used by ChatGPT-o1 is provided in Appendix C.

Visualization of Recommending Weights. For the following contract clause:

Content of contract clause

If there are natural disasters such as earthquakes and typhoons or force majeure events such as wars that affect the performance of the contract, the relevant parties must notify the other party as soon as possible and provide a formal force majeure certificate issued by the local government agency or notary department within 15 days. The two parties will jointly decide whether to suspend, adjust, or terminate the contract based on the impact of the force majeure event.

Figure 4 visualizes the attention weights between a contract clause and its most relevant legal provision, where the text encoder is Mamba, highlighting key terms like "force majeure," "reasonable

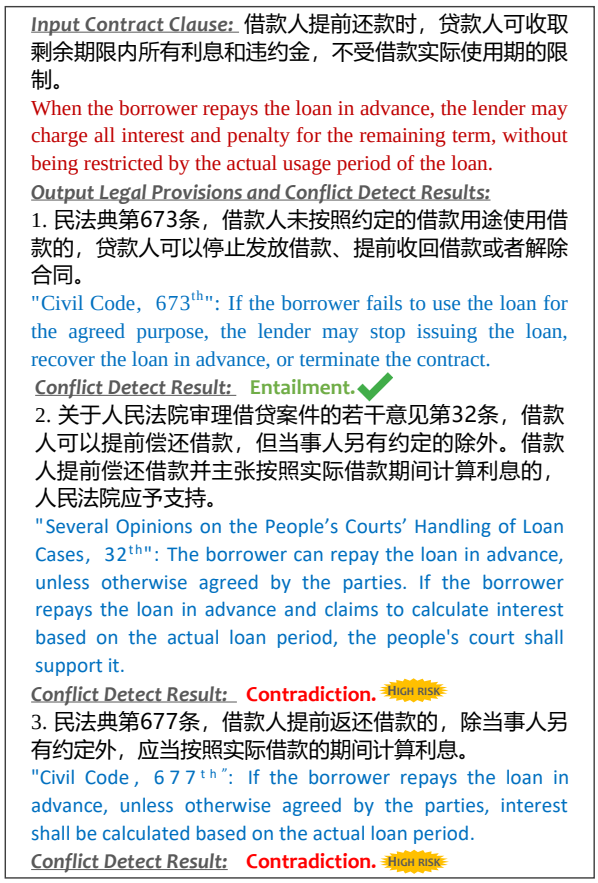

Figure 5: An example of PROVBENCH to recommend related legal provisions and conflict detection results.

period of time," and "loss." These terms closely match the content of the clause, demonstrating the ability of PROVBENCH to focus on critical elements when determining the relationship between the contract and the legal provision.

5.6 Case Study and Error Analysis

Case Study. As shown in Figure 5, we present an example of a loan contract review using PROVBENCH. This case demonstrates that PROVBENCH accurately matches the legal provisions related to "early repayment". Additionally, it identifies the unreasonable aspect of the "interest calculation period" in the contract clause and detects which specific legal provision this clause violates.

Error Analysis. The effectiveness of PROVBENCH has been demonstrated in the previous sections, but some limitations remain. We conduct an error analysis to classify incorrect predictions into two categories, as shown in Figure 6: (1) Incorrect Provision Recommendation: PROVBENCH sometimes recommends provisions associated with clauses that are similar in description but differ in meaning within the same contract type. (2) Incorrect Conflict Detection: PROVBENCH fails to accurately determine the logical relationship between clauses and provisions, leading to misidentified conflicts.

These conflicts often arise from implicit logical details embedded in the clauses, making them challenging to analyze and detect. Future work will focus on strengthening the ability of PROVBENCH for precise semantic interpretation and comprehensive analysis of logical relationships, enhancing its accuracy and reliability in supporting lawyers during contract review.

6 Related Work

Legal NLP. In recent years, Legal NLP has gained increasing attention, with research advancing tasks such as legal text retrieval (Feng et al., 2024) and understanding (Paul et al., 2022). For instance, El Jelali et al. (2015) retrieved relevant court decisions with respect to the disputant case description. Ma et al. (2022); Sampath and Durairaj (2022); Li et al. (2023a) focused on matching descriptions from similar cases to support legal decisions. Moreover, Askari et al. (2024) utilized retrieval for legal question answering. Joshi et al. (2024) proposed IL-TUR, a multilingual benchmark for evaluating legal text understanding and reasoning in the Indian legal system. These methods advance law and artificial intelligence integration but focus little on linking legal recommendations to contract review.

Legal Text Entailment. Textual entailment aims to determine whether the meaning of one text logically follows from another (Parikh et al., 2016). This concept has been adopted in the legal domain for tasks like Legal Case Entailment (Goebel et al., 2023, 2024), which identifies supporting paragraphs from cases that justify a query decision. To illustrate, pre-trained models have been widely employed to enhance performance (Li et al., 2023b; Nguyen et al., 2024). Effective data augmentation (Aoki et al., 2022; Yoshioka et al., 2021) also plays an important role. In this paper, we extend this concept to evaluate the logical compliance of contract clauses with legal provisions, thus assisting in determining the legality of the given contract clause.

Incorrect Provision Recommendation	***Contract Clause:*** 逾期超过60日，买受人愿意继续履行合同的，经出卖人同意，合同继续履行，自约定的应付款期限届满之日起至实际全额支付应付款之日止，买受人按日计算向出卖人支付逾期应付款万分之七的违约金。 ***Translate to English:*** If the overdue period exceeds 60 days and the buyer is willing to continue to perform the contract, with the consent of the seller, the contract shall continue to be performed. From the expiration of the agreed payment deadline to the actual full payment date, the buyer shall pay the seller a penalty of 0.07% of the overdue payment per day. ***Ground Truth：*** The Supreme People's Court's Interpretation on Several Issues concerning the Application of Law in the Trial of Cases Involving Disputes over Sales Contracts of Commodity Houses. 14th (entailment), 11th (entailment) ,15th (entailment). ***Recommend Results：*** The Supreme People's Court's Interpretation on Several Issues concerning the Application of Law in the Trial of Cases Involving Disputes over Sales Contracts of Commodity Houses. **13th (entailment)**, 11th (entailment) ,15th (entailment).
Incorrect Conflict Detection	***Contract Clause:*** 在施工过程中，若承揽人认为使用替代材料对施工周期、工艺效果或预算控制有正面影响，承揽人有权无需定作人同意便可实施该调整。定作人不得因材料替代影响而要求赔偿或重新协商合同条款。 **Translate to English:** During the construction process, if the contractor believes that the use of alternative materials has a positive impact on the construction period, process effect, or budget control, the contractor has the right to implement the adjustment without the consent of the ordering party. The client shall not demand compensation or renegotiate the terms of the contract due to the impact of material substitution. ***Ground Truth：*** The Contract section of the People's Republic of China Civil Code. 774th (contradictory), 775th (contradictory) ,784th (entailment). ***Recommend Results：*** The Contract section of the People's Republic of China Civil Code. **774th (entailment)**, 775th (contradictory) ,784th (contradictory).

Figure 6: Case studies of error cases.

Contract Review. Previous methods have explored various approaches to assist contract review. For example, Fenech et al. (2009) designed keyword-based categories for risk detection, while Leivaditi et al. (2020); Tecuci et al. (2020) identified contract entities and generated risk prompts based on their types. Checking through the consistency of keywords before and after (Zhang et al., 2021) has also been applied by extracting key transaction values to assess contract risks. Additionally, Hendrycks et al. (2021) differentiated importance levels based on clause types. Wang et al. (2023) understand legal texts to extract transaction points.

7 Conclusion

In this paper, we propose a novel and meaningful task, PROVREC, aimed at providing clear evidence for contract auto-reviewing. Furthermore, we establish a benchmark PROVBENCH to support this task. We construct the PROVDATA dataset, which contains a diverse set of contract clauses, legal provisions, and their annotated logical relationships. Additionally, we design the framework for PROVBENCH through two subtasks: recommending relevant legal provisions and detecting potential legal conflicts between contract clauses and legal provisions. Extensive experiments demonstrate the effectiveness of PROVBENCH in both legal provision recommendation and conflict detection. We hope that the PROVBENCH established in this paper can promote the development of contract auto-reviewing.

8 Limitations

One limitation of our research lies in the use of contract clauses primarily derived from publicly available templates, which are structurally consistent and rarely exhibit substantial conflicts. Although we enrich data diversity through data expansion, more complex and realistic scenarios remain underexplored. In particular, implicit unfairness arising from interactions among multiple clauses, especially when vague or inconsistent terms collectively affect obligations, poses a significant challenge for automated contract review.

Another limitation is that our dataset focuses on 8 representative contract types. The contract clauses within the same category often correspond to overlapping legal provisions, which reflects the structure of real-world legal practice, where specific legal norms consistently regulate similar types of transactions. However, such a design may constrain the capacity of the model to generalize to previously unseen contract categories or legal domains. While variation in clause phrasing promotes semantic understanding beyond surface-level pattern matching, we aim to expand the benchmark to cover a wider range of contract scenarios and legal systems in future work.

Acknowledgements

This work is supported by the National Key R&D Project of China (Grant No. 2022YFB2701800), and the Zhejiang Provincial Natural Science Foundation of China (No. LQ24F020019).

References

Yasuhiro Aoki, Masaharu Yoshioka, and Youta Suzuki. 2022. Data-augmentation method for bert-based legal textual entailment systems in coliee statute law task. *The Review of Socionetwork Strategies*, 16(1):175–196.

Arian Askari, Zihui Yang, Zhaochun Ren, and Suzan Verberne. 2024. Answer retrieval in legal community question answering. In *European Conference on Information Retrieval*, pages 477–485. Springer.

Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, Nikolaos Aletras, and Ion Androutsopoulos. 2020. LEGAL-BERT: The muppets straight out of law school. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2898–2904, Online. Association for Computational Linguistics.

Fabio Crestani, Mounia Lalmas, Cornelis J Van Rijsbergen, and Iain Campbell. 1998. "is this document relevant?... probably" a survey of probabilistic models in information retrieval. *ACM Computing Surveys (CSUR)*, 30(4):528–552.

DeepSeek-AI. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Soufiane El Jelali, Elisabetta Fersini, and Enza Messina. 2015. Legal retrieval as support to emediation: matching disputant's case and court decisions. *Artificial Intelligence and Law*, 23:1–22.

Stephen Fenech, Gordon J Pace, and Gerardo Schneider. 2009. Clan: A tool for contract analysis and conflict discovery. In *International Symposium on Automated Technology for Verification and Analysis*, pages 90–96. Springer.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2022. Language-agnostic bert sentence embedding. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 878–891.

Yi Feng, Chuanyi Li, and Vincent Ng. 2024. Legal case retrieval: A survey of the state of the art. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6472–6485.

Zheren Fu, Zhendong Mao, Yan Song, and Yongdong Zhang. 2023. Learning semantic relationship among instances for image-text matching. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15159–15168.

Randy Goebel, Yoshinobu Kano, Mi-Young Kim, Juliano Rabelo, Ken Satoh, and Masaharu Yoshioka. 2023. Summary of the competition on legal information, extraction/entailment (coliee) 2023. In *Proceedings of the Nineteenth International Conference on Artificial Intelligence and Law*, pages 472–480.

Randy Goebel, Yoshinobu Kano, Mi-Young Kim, Juliano Rabelo, Ken Satoh, and Masaharu Yoshioka. 2024. Overview of benchmark datasets and methods for the legal information extraction/entailment competition (coliee) 2024. In *JSAI International Symposium on Artificial Intelligence*, pages 109–124. Springer.

Albert Gu and Tri Dao. 2023. Mamba: Linear-time sequence modeling with selective state spaces. *arXiv preprint arXiv:2312.00752*.

Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. Cuad: An expert-annotated nlp dataset for legal contract review. *NeurIPS*.

Benjamin E. Hermalin, Avery W. Katz, and Richard Craswell. 2007. Chapter 1 contract law. volume 1 of *Handbook of Law and Economics*, pages 3–138. Elsevier.

Qiao Hu, Juan Du, Ruilin Li, and Bibin Leng. 2018. Study on the model of financial centralized management in the large-scale construction enterprises. *IOP Conference Series: Materials Science and Engineering*, 439(3):032040.

Abhinav Joshi, Shounak Paul, Akshat Sharma, Pawan Goyal, Saptarshi Ghosh, and Ashutosh Modi. 2024. Il-tur: Benchmark for indian legal text understanding and reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11460–11499.

Andrej Karpathy and Li Fei-Fei. 2015. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3128–3137.

Ryan Kiros, Ruslan Salakhutdinov, and Richard S Zemel. 2014. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*.

Akhila Krishna, Ravi Kant Gupta, Pranav Jeevan, and Amit Sethi. 2024. Advancing gene selection in oncology: A fusion of deep learning and sparsity for precision gene selection. *arXiv preprint arXiv:2403.01927*.

Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked cross attention for

image-text matching. In *Proceedings of the European conference on computer vision (ECCV)*, pages 201–216.

Spyretta Leivaditi, Julien Rossi, and Evangelos Kanoulas. 2020. A benchmark for lease contract review. *arXiv preprint arXiv:2010.10386*.

Haitao Li, Qingyao Ai, Jia Chen, Qian Dong, Yueyue Wu, Yiqun Liu, Chong Chen, and Qi Tian. 2023a. Sailer: structure-aware pre-trained language model for legal case retrieval. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1035–1044.

Haitao Li, Changyue Wang, Weihang Su, Yueyue Wu, Qingyao Ai, and Yiqun Liu. 2023b. Thuir@ coliee 2023: more parameters and legal knowledge for legal case entailment. *arXiv preprint arXiv:2305.06817*.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Marco Lippi, Przemysław Pałka, Giuseppe Contissa, Francesca Lagioia, Hans-Wolfgang Micklitz, Giovanni Sartor, and Paolo Torroni. 2019. Claudette: an automated detector of potentially unfair clauses in online terms of service. *Artificial Intelligence and Law*, 27:117–139.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Minh-Thang Luong. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.

Yixiao Ma, Qingyao Ai, Yueyue Wu, Yunqiu Shao, Yiqun Liu, Min Zhang, and Shaoping Ma. 2022. Incorporating retrieval information into the truncation of ranking lists for better legal search. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 438–448.

Chau Nguyen, Phuong Nguyen, Thanh Tran, Dat Nguyen, An Trieu, Tin Pham, Anh Dang, and Le-Minh Nguyen. 2024. Captain at coliee 2023: Efficient methods for legal information retrieval and entailment tasks.

Geunseob Oh, Rahul Goel, Chris Hidey, Shachi Paul, Aditya Gupta, Pararth Shah, and Rushin Shah. 2022. Improving top-k decoding for non-autoregressive semantic parsing via intent conditioning. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 310–322.

OpenAI. 2024. Learning to reason with llms.

Marc Pàmies, Joan Llop, Francesco Multari, Nicolau Duran-Silva, César Parra-Rojas, Aitor Gonzalez-Agirre, Francesco Alessandro Massucci, and Marta Villegas. 2023. A weakly supervised textual entailment approach to zero-shot text classification. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 286–296, Dubrovnik, Croatia. Association for Computational Linguistics.

Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.

Badri N Patro and Vijay S Agneeswaran. 2024. Simba: Simplified mamba-based architecture for vision and multivariate time series. *arXiv preprint arXiv:2403.15360*.

Shounak Paul, Pawan Goyal, and Saptarshi Ghosh. 2022. Lesicin: A heterogeneous graph-based approach for automatic legal statute identification from indian legal documents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 11139–11146.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, and 1 others. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Kayalvizhi Sampath and Thenmozhi Durairaj. 2022. Prelcap: precedence retrieval from legal documents using catch phrases. *Neural Processing Letters*, 54(5):3873–3891.

Ralph Schuhmann and Bert Eichhorn. 2015. From contract management to contractual management. *European Review of Contract Law*, 11(1):1–21.

Dan G Tecuci, Ravi Palla, Hamid R Motahari Nezhad, Nishchal Ahuja, Alex Monteiro, Tigran Ishkhanov, and Nigel Duffy. 2020. Dicr: Ai assisted, adaptive platform for contract review. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13638–13639.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Feng Wang, Jiahao Wang, Sucheng Ren, Guoyizhe Wei, Jieru Mei, Wei Shao, Yuyin Zhou, Alan Yuille, and Cihang Xie. 2024. Mamba-r: Vision mamba also needs registers. *arXiv preprint arXiv:2405.14858*.

Guiling Wang and Yimin Chen. 2022. [retracted] enabling legal risk management model for international corporation with deep learning and self data mining. *Computational Intelligence and Neuroscience*, 2022(1):6385404.

Steven H Wang, Antoine Scardigli, Leonard Tang, Wei Chen, Dimitry Levkin, Anya Chen, Spencer Ball, Thomas Woodside, Oliver Zhang, and Dan Hendrycks. 2023. Maud: An expert-annotated legal nlp dataset for merger agreement understanding. *arXiv preprint arXiv:2301.00876*.

Jie Wu, Chunlei Wu, Jing Lu, Leiquan Wang, and Xuerong Cui. 2021. Region reinforcement network with topic constraint for image-text matching. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(1):388–397.

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.

Shuo Zhang, Junzhou Zhao, Pinghui Wang, Nuo Xu, Yang Yang, Yiting Liu, Yi Huang, and Junlan Feng. 2021. Learning to check contract inconsistencies. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 14446–14453.

Zhe Zhao, Hui Chen, Jinbin Zhang, Xin Zhao, Tao Liu, Wei Lu, Xi Chen, Haotang Deng, Qi Ju, and Xiaoyong Du. 2019. Uer: An open-source toolkit for pre-training models. *EMNLP-IJCNLP 2019*, page 241.

Zibin Zheng, Shaoan Xie, Hong-Ning Dai, Weili Chen, Xiangping Chen, Jian Weng, and Muhammad Imran. 2020. An overview on smart contracts: Challenges, advances and platforms. *Future Generation Computer Systems*, 105:475–491.

A Example Appendix

A.1 Data Processing Workflow

A.1.1 Legal Provision Library Construction

To enhance future scalability in contract review automation, the annotators also categorize the legal provisions into 15 tags based on contract elements as shown in Figure 7, including *subject, quality, price, delivery and performance, obligations, confidentiality, breach, interest, termination, validity, amendments and transfer, parties, property, litigation, and guarantees.* The *obligations* label constitutes the largest proportion at 28.1%, followed by *parties* at 16.9%. The *breach* and *termination* labels are similar, at 10.4% and 9.4%, respectively. The remaining labels have lower proportions, with *guarantees* and *confidentiality* being the least, each below 1%.

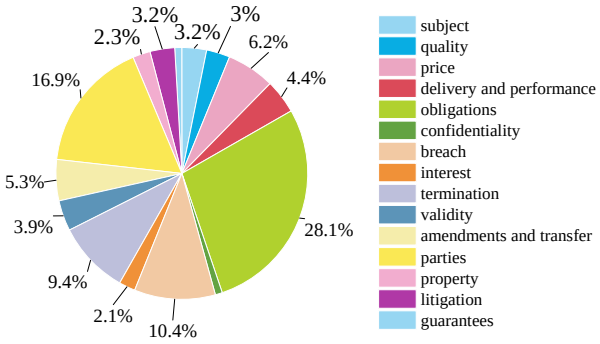

Figure 7: Distribution of legal semantic categories.

A.1.2 Legal Provisions Recommendation

During the legal provision recommendation phase, annotators first create a CSV file containing three columns: "contract clause," "legal provision ID," and "conflict label." Since each contract clause corresponds to three legal provisions, each clause is repeated three times in the resulting file 'temp.csv'. For each contract clause, the annotators manually select the top 3 most semantically relevant legal provisions from the legal provision library. Then, they evaluate the logical relationship between the clause and each provision. Consistent relationships are labeled as "0" (entailment), while contradictory ones are labeled as "1" (contradiction). The process can be formulated as:

The process to recommend legal provisions

1. Initialize
Create an empty list 'csv_rows' to store the entries.
2. Select legal provisions.
For each 'contract clause' in the list of contract clauses:
a. Select the top 3 most relevant legal provisions from the legal provision library.
b. For each 'provision' in the selected relevant provisions:
i. Evaluate the logical relationship between the 'contract clause' and the 'provision'.
ii. If the relationship is consistent (entailment), label it as '0'; otherwise, label it as '1' (contradiction).
c. Append a new row to 'csv_rows' with:
- 'contract clause': The current clause.
- 'legal provision ID': The ID of the provision.
- 'conflict label': The evaluated conflict label.
3. Generate a CSV file: 'temp.csv'.

A.1.3 Generation of ProvData

This section illustrates the detailed process of constructing the dataset, which involves organizing contract clauses and their associated legal provisions into a structured JSON format. As shown below, the process begins by extracting relevant information, such as legal provision IDs and con-

Steps to generate the final version of the PROVDATA

1. Initialize Variables.
$id \leftarrow 0$: Unique identifier for each data.
$contractRows \leftarrow []$: Temporary storage for the legal provision id associated with each contract.
PROVDATA$\leftarrow []$: Final structured data.
2. Iterate Through Data Rows.
Extract $contractClause$, $legalProvisionID$, and $conflictLabel$ from each row in $contractClauseData$.
Retrieve detailed legal information (e.g., $legalProvision$, $legalCategory$, $legalProvisionNumber$) using $legalProvisionID$.
Append extracted data to law_j and store in $contractRows$.
3. Group Data into Contracts
Check if 3 rows in $contractRows$ have consistent contract clause:
If consistent, proceed.
If inconsistent, print a warning about contract content mismatch.
Generate a complete contract JSON structure:
- id: Contract ID (starting from 0).
- $contractClause$: Contract clause stored at the top level.
- law_1, law_2, law_3: 3 associated legal data entries.

Add generated JSON to PROVDATA.
Reset $contractRows$, increment id.
4. Handle Incomplete Contracts
If the remaining data is less than 3 rows, repeat the above steps to generate the corresponding JSON.
Check contract content consistency in $contractRows$ and consolidate the remaining data into a complete contract JSON.
5. Output Result
PROVDATAcontains all processed contract JSON data, with each contract clause associated with 3 legal data entries.

TextEncoder	R@1	R@3	R@5	Top-3 EM
T5♮	84.38	93.75	96.88	52.90
T5	86.38	95.09	97.10	60.94
BERT♮	90.40	97.32	98.21	72.77
BERT	90.63	96.86	98.21	76.79
RoBERTa♮	88.62	94.92	95.98	69.20
RoBERTa	89.96	95.31	97.54	70.31
LegalBERT♮	88.84	95.94	97.54	67.86
LegalBERT	92.19	97.77	98.88	73.88
Mamba♮	70.09	88.17	94.42	28.35
Mamba	91.29	97.99	99.33	57.37

Table 7: Ablation study results, where ♮ indicates the model without the attention mechanism.

flict labels, and retrieving detailed legal data. The data is grouped into sets of three provisions per contract clause to ensure consistency across entries. Each group is then consolidated into a complete JSON structure, accommodating any remaining incomplete data. This systematic approach ensures that each contract clause is paired with three relevant legal provisions, forming a well-structured and comprehensive dataset.

A.2 Example of PROVDATA

A detailed example from the PROVDATA is shown in Table 8. It illustrates a sales contract clause, where is paired with three relevant legal provisions. Each provision is annotated with its corresponding legal code, serial number within the legal code, and category. Additionally, a conflict label is assigned to each provision, indicating the logical relationship between the contract clause and the legal provision, such as whether the relationship is entailment or contradictory.

B Ablation Study

As shown in Table 7, we conduct an ablation study to evaluate the effectiveness of incorporating a variant of Dot Product Attention (Luong, 2015), following the design proposed by Lee et al. (2018). The results demonstrate that the attention mechanism generally leads to improved performance across most text encoders for recommending legal provisions. However, the improvement is relatively limited for T5 and BERT-based models. We speculate that these models have already acquired strong language understanding capabilities through large-scale pretraining, rendering the additional attention module less impactful.

C Prompt Example for ChatGPT-o1

In our research, we replace the proposed conflict detector with ChatGPT-o1 to conduct an extended test on using large models for aligning contract clauses with legal provisions. This allows us to explore the feasibility of using a pre-trained large

► Examples in Chinese	► Examples after translating into English
“id”: 2	id: 2
“contract clause:” “乙方应根据甲方要求提供与现有监测系统兼容的环境监测设备。如设备引起系统故障或误差,乙方须负责更换设备并赔偿由此造成的损失。”	Party B shall provide environmental monitoring equipment that is compatible with the existing monitoring system according to the requirements of Party A. In case the equipment causes system failures or errors, Party B shall be responsible for replacing the equipment and compensating for the losses thus caused.
“law1”: “买受人在检验期间,质量保证期间,合理期间内提出质量异议, 出卖人未按要求予以修理或者因情况紧急,买受人自行或者通过第三人修理标的物后,主张出卖人负担因此发生的合理费用的,人民法院应予支持.”	If the buyer raises an objection regarding the quality of the subject matter during the inspection period, the quality guarantee period or a reasonable period, and the seller fails to make repairs as required, or in case of emergency where the buyer has the subject matter repaired by itself or through a third party, if the buyer claims that the seller should bear the reasonable expenses incurred thereby, the people’s court shall support such a claim.
“num1”: “22”	num1: 22
“cate1”: 《最高人民法院关于审理买卖合同纠纷案件适用法律问题的解释》	cate1: The Interpretation of the Supreme People’s Court on the Applicable Law in the Trial of Disputes over Sale Contracts.
“label1”: “2”	label1: 2
“conflict_label”: “0”	conflict_label: 0
“law2”: “出卖人交付的标的物不符合质量要求的,买受人可以依据本法第五百八十二条至第五百八十四条的规定请求承担违约责任.”	If the subject matter delivered by the seller fails to meet the quality requirements, the buyer may request the seller to bear the liability for breach of contract in accordance with the provisions of Articles 582 to 584 of this Law.
“num2”: “617”	num2: 617
“cate2”: 《中华人民共和国民法典》	cate2: The Civil Code of People’s Republic of China.
“label2”: “2”	label: 2
“conflict_label”: “0”	conflict_label: 0
“law3”: “因标的物不符合质量要求,致使不能实现合同目的的,买受人可以拒绝接受标的物或者解除合同.买受人拒绝接受标的物或者解除合同的,标的物毁损,灭失的风险由出卖人承担.”	If the subject matter fails to meet the quality requirements, resulting in the failure to achieve the purpose of the contract, the buyer may refuse to accept the subject matter or terminate the contract. Where the buyer refuses to accept the subject matter or terminates the contract, the risk of damage to or loss of the subject matter shall be borne by the seller.
“num3”: “610”	num3: 610
“cate3”: 《中华人民共和国民法典》	cate3: The Civil Code of People’s Republic of China.
“label3”: “8”	label3: 8
“conflict_label”: “0”	conflict_label: 0

Table 8: Example of sale contract clause

language model for legal conflict detection. The design of prompt focuses on identifying logical contradictions between contract clauses and relevant legal provisions. It uses structured inputs consisting of a contract clause and legal provisions, and instructs the model to determine whether they are logically entailment or contradictory. The prompt is provided as follows:

Prompt Design for Conflict Detection using ChatGPT-o1

►**Definition**: Determine whether there is a potential semantic contradiction between the contract in each data and the corresponding law1, law2, and law3, and return a "CSV" file in the format of: ids, law1_result, law2_result, law3_result, where the corresponding values of law1_result, law2_result, and law3_result are 0 or 1, 0 means there is no semantic contradiction, and 1 means there is a potential conflict. Don't explain, just output the result directly.
►**Input**: 合同条款: 借款人如未按合同规定的期限偿还债务，贷款人有权暂停提供任何进一步的贷款，并向借款人发出书面通知要求偿还借贷款项。借款人若在收到通知后的15日内未偿还逾期款项，贷款人可解除合同并追究赔偿。
法律条款:
法律 1: 债权人分立,合并或者变更住所没有通知债务人,致使履行债务发生困难的,债务人可以中止履行或者将标的物提存
法律 2: 应当先履行债务的当事人,有确切证据证明对方有下列情形之一的,可以中止履行:(一)经营状况严重恶化;(二)转移财产,抽逃资金,以逃避债务;(三)丧失商业信誉;(四)有丧失或者可能丧失履行债务能力的其他情形.当事人没有确切证据中止履行的,应当承担违约责任.
法律 3:当事人依据前条规定中止履行的,应当及时通知对方.对方提供适当担保的,应当恢复履行.中止履行后,对方在合理期限内未恢复履行能力且未提供适当担保的,视为以自己的行为表明不履行主要债务,中止履行的一方可以解除合同并可以请求对方承担违约责任.
►**Input Translate to English**:
Contract Clause: If the borrower fails to repay the debt within the time specified in the contract, the lender has the right to suspend any further loans and send a written notice to the borrower requesting repayment of the loan amount. If the borrower fails to repay the overdue amount within 15 days after receiving the notice, the lender may terminate the contract and pursue compensation.
Legal Clauses:
Legal Provision 1: If the creditor divides, merges, or changes its address without notifying the debtor, causing difficulties in performing the debt, the debtor may suspend performance or deposit the subject matter.
Legal Provision 2: The party that should perform the debt first may suspend performance if there is conclusive evidence proving that the other party has one of the following situations: (1) severe deterioration in business conditions; (2) transfer of property or withdrawal of funds to evade debts; (3) loss of commercial credit; (4) other situations where the party may lose the ability to perform its debt. If the party does not have conclusive evidence to suspend performance, it shall bear the responsibility for breach of contract.
Legal Provision 3: If a party suspends performance based on the preceding provisions, it must promptly notify the other party. If the other party provides appropriate guarantees, performance must be resumed. If the other party fails to restore the ability to perform within a reasonable period and does not provide appropriate guarantees, it shall be deemed to have indicated by its own actions that it will not perform the primary debt, and the party that suspended performance may terminate the contract and request the other party to bear the responsibility for breach of contract.
►**Output**:[0,0,1]