Graph-Enhanced Prompt Learning for Cross-Domain Contract Element Extraction

ZIHAN WANG, Shandong University, Qingdao, China and University of Amsterdam, Amsterdam, Netherlands
HANBING WANG and PENGJIE REN, Shandong University, Qingdao, China
ZHUMIN CHEN, Shandong University, Jinan, China
MAARTEN DE RIJKE, University of Amsterdam, Amsterdam, Netherlands
ZHAOCHUN REN, Leiden University, Leiden, Netherlands

Cross-domain contract element extraction (CEE) aims to transfer knowledge from a source domain to facilitate the extraction of legally relevant elements (e.g., contract dates or payments) from contracts in a target domain. To achieve this goal, recent studies encode the domain-invariant relations between elements and legal clause types and enhance performance through bidirectional supervision between the CEE task and the clause classification task. However, two challenges remain unresolved—(i) data sparsity due to expensive annotation costs and a large number of element types, and (ii) label discrepancies among element types across domains, both of which severely impede effective knowledge transfer from the source to the target domain.

Recent developments in prompt learning have shown promising performance in low-resource settings. Drawing inspiration from these advances, we propose a novel framework, *graph-enhanced prompt learning* (GEPL), for the cross-domain CEE task to address these challenges. GEPL includes two kinds of prompt: (i) instance-oriented prompts and (ii) label-oriented prompts. Given the input instances, instance-oriented prompts are automatically generated by retrieving relevant examples in the training data, providing auxiliary supervision to enhance the transfer process in low-resource scenarios. To mitigate label discrepancies across different domains, we identify relations among element types using mutual-information criteria and transform these into label-oriented prompt templates. On this basis, a multi-task training strategy is designed

This work was supported by the National Key R&D Program of China with Grant No. 2022YFC3303004, the Natural Science Foundation of China (Grant Nos. 62372275, 62272274, 62202271, T2293773, 62102234, 62072279), the Natural Science Foundation of Shandong Province (Grant Nos. ZR2022QF004, ZR2021QF129), the Key Scientific and Technological Innovation Program of Shandong Province (Grant No. 2019JZZY010129), the Fundamental Research Funds of Shandong University, the China Scholarship Council under Grant No. 202206220085, the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organization for Scientific Research, https://hybrid-intelligence-centre.nl, project LESSEN with project number NWA.1389.20.183 of the research program NWA ORC 2020/21, which is (partly) financed by the Dutch Research Council (NWO), project ROBUST with project number KICH3.LTP.20.006, which is (partly) financed by the Dutch Research Council (NWO) and the Dutch Ministry of Economic Affairs and Climate Policy (EZK) under the program LTP KIC 2020-2023, and the FINDHR (Fairness and Intersectional Non-Discrimination in Human Recommendation) project that received funding from the European Union's Horizon Europe research and innovation program under grant agreement No. 101070212. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

Authors' Contact Information: Zihan Wang, Shandong University, Qingdao, China and University of Amsterdam, Amsterdam, Netherlands; e-mail: zihanwang.sdu@gmail.com; Hanbing Wang, Shandong University, Qingdao, China; e-mail: hanbingwang01@gmail.com; Pengjie Ren, Shandong University, Qingdao, China; e-mail: jay.ren@outlook.com; Zhumin Chen, Shandong University, Jinan, China; e-mail: chenzhumin@sdu.edu.cn; Maarten de Rijke, University of Amsterdam, Amsterdam, Netherlands; e-mail: m.derijke@uva.nl; Zhaochun Ren (corresponding author), Leiden University, Leiden, Netherlands; e-mail: z.ren@liacs.leidenuniv.nl.

This work is licensed under a Creative Commons Attribution International 4.0 License.

© 2025 Copyright held by the owner/author(s).

ACM 1558-2868/2025/3-ART66
https://doi.org/10.1145/3715100

to simultaneously optimize the representations of the original input sentence and prompts, enabling GEPL to better understand the tasks and capture label relations in both source and target domains. Empirical results on cross-domain CEE datasets indicate that GEPL significantly outperforms state-of-the-art baselines. Moreover, extensive experiments reveal that GEPL achieves the state-of-the-art performance on cross-domain named entity recognition datasets and demonstrates a high level of generalizability. Our code is released at https://github.com/WZH-NLP/GEPL.

CCS Concepts: • **Information systems** → **Information extraction**; **Specialized information retrieval**;

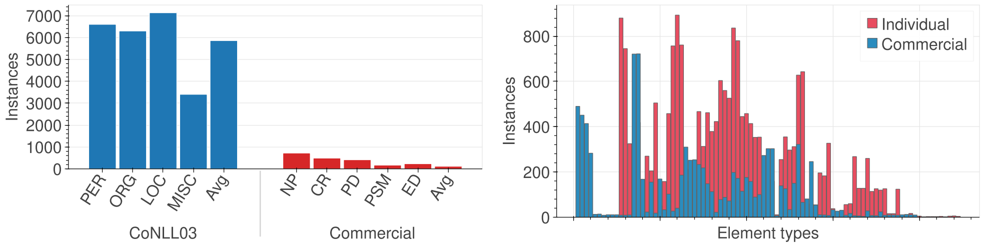
Additional Key Words and Phrases: Legal information extraction and retrieval, Cross-domain information extraction, Sequence labeling, Cross-domain learning

ACM Reference format:
Zihan Wang, Hanbing Wang, Pengjie Ren, Zhumin Chen, Maarten De Rijke, and Zhaochun Ren. 2025. Graph-Enhanced Prompt Learning for Cross-Domain Contract Element Extraction. *ACM Trans. Inf. Syst.* 43, 3, Article 66 (March 2025), 24 pages.
https://doi.org/10.1145/3715100

1 Introduction

Every day, a multitude of contracts are drafted for various transactions, such as services, leases, or sales. These contracts contain many crucial elements, such as termination dates and information of parties involved [7, 9, 49]. The manual monitoring of legally relevant elements within a large volume of contracts imposes a heavy burden for law firms, companies, and government agencies [67]. Consequently, automatic ***Contract Element Extraction*** **(CEE)** has become an essential task for businesses globally, aimed at identifying legally relevant elements within contract clauses. For example, as shown in Figure 2, given the input clause "After Party A accepts the delivered goods from Party B as satisfactory, ... within 15 working days." from a contract, our goal is to identify the contract elements, such as Payment Condition "After Party A accepts the delivered goods from Party B as satisfactory" and **Payment Period (PP)** "within 15 working days." The automatic CEE task enables a wide range of downstream applications like clause relation extraction and risk assessment [4, 48, 74]. Nonetheless, due to the high costs associated with manual labeling, there has been increasing research interest in cross-domain CEE, focusing on transferring knowledge from a source domain to a low-resource target domain.

As mentioned in [7, 8], from multiple perspectives, cross-domain CEE is similar to cross-domain **Named Entity Recognition (NER)** [27, 28, 77]. Despite these similarities, the cross-domain CEE task presents two additional difficulties compared to cross-domain NER: (i) Cross-domain CEE focuses on transferring a more extensive set of fine-grained contract element types. While a generic named entity recognizer typically extracts a limited set of entity types, such as persons, organizations, or locations, the CEE task aims to identify a much broader range of specific contract elements [72]. In the cross-domain setting, this larger number of fine-grained contract elements makes transferring the element extractor from one domain to another challenging. (ii) The extraction zones for contract elements (i.e., contract clauses) are substantially larger. Whereas the NER task primarily focuses on identifying entities within a single sentence, the CEE task necessitates extracting contract elements that often span multiple sentences within a clause. This larger extraction zone with varied organizational structures across domains significantly impedes the effectiveness of existing cross-domain methods [72]. To address these difficulties, a bidirectional feedback scheme between the CEE task and the **Clause Classification (CC)** task has recently been designed by Wang et al. [72]. The key idea is to identify domain-agnostic relations between elements and legal clauses. However, current cross-domain CEE methods [72] still face two challenging problems:



(a) Data sparsity.

(b) Label mismatches.

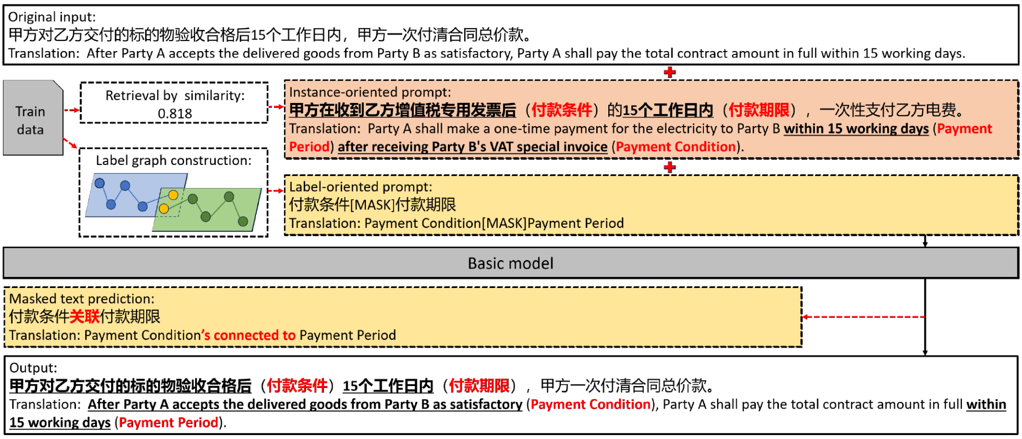
Fig. 1. (a) Data sparsity. Blue bars represent the number of instances for entity types (i.e., Person (PER), Organization (ORG), Location (LOC), and Miscellaneous (MISC)) in the CoNLL03 dataset, and red bars represent the number of instances for element types (i.e., Name of Party A (NPA), Penalty Payment Ratio (PPR), Payment Date (PD), Price of Subject Matter (PSM), and Effective Date (ED)) in the Commercial dataset [72]. (b) Label discrepancies. Blue bars represent the number of instances in the Commercial dataset, and red bars represent the number of instances in the Individual dataset [72].

Challenge 1: Data Sparsity. As mentioned earlier, cross-domain CEE involves transferring a more extensive range of fine-grained contract element types compared to tasks like cross-domain NER. For example, the Commercial dataset [72] (a CEE dataset) contains over 70 types of elements, while the widely used CoNLL03 dataset [63] for NER includes only four types of named entity. Additionally, manual labeling of contract elements is expensive, labor-intensive, and prone to errors [67]. Consequently, existing cross-domain CEE methods suffer from severe data sparsity problems. Specifically, Figure 1(a) demonstrates the notably lower number of instances for element types in the Commercial dataset compared to entity types in CoNLL03.

*Challenge 2: Label Discrepancies.*Another key challenge that has not yet been fully addressed is the inconsistency of labels between the source and target domains. Existing cross-domain CEE methods incorporate domain-invariant knowledge about legal clauses and elements, employing a multi-task framework with bidirectional feedback between CC and CEE [72]. Nevertheless, the discrepancies in element labels across different domains have often been overlooked. For example, as Figure 1(b) shows, 17.2% of element types in the Individual dataset do not appear in the Commercial dataset, significantly impeding the transfer of contract element extractors from one domain to another. Additionally, there is a notable variance in the distribution of instances for element types across different domains, and the dependencies between element labels are not always consistent between the source and target domains.

To alleviate data sparsity (Challenge 1), we automatically generate auxiliary supervisions for a given input instance by retrieving closely relevant examples in the training data. In order to mitigate label discrepancies across domains (Challenge 2), we take steps to identify relationships among different types of elements across domains. These relationships are then transformed into textual templates, which serve as model input to capture dependencies among element labels.

In this article, we use prompt-based learning, which has been pioneered by the GPT series of **Large Language Models (LLMs)** [5, 58, 59] and achieves promising performance in low-resource scenarios [19, 34, 37, 44]. Prompt-based learning methods formulate the downstream task as a (masked) **Language Modeling (LM)** problem by finding appropriate natural language prompts, reducing or eliminating the need for large supervised datasets [37]. Inspired by this, we propose a framework, named ***Graph-Enhanced Prompt Learning*** **(GEPL)**, that incorporates auxiliary supervision and relationships among element labels across domains. As Figure 2 shows, GEPL includes two types of prompts: (i) instance-oriented prompts and (ii) label-oriented prompts.



Original input:
甲方对乙方交付的标的物验收合格后15个工作日内，甲方一次付清合同总价款。
Translation: After Party A accepts the delivered goods from Party B as satisfactory, Party A shall pay the total contract amount in full within 15 working days.

Instance-oriented prompt:
甲方在收到乙方增值税专用发票后（**付款条件**）的**15个工作日内**（**付款期限**），一次性支付乙方电费。
Translation: Party A shall make a one-time payment for the electricity to Party B **within 15 working days** (**Payment Period**) **after receiving Party B's VAT special invoice** (**Payment Condition**).

Label-oriented prompt:
付款条件[MASK]付款期限
Translation: Payment Condition[MASK]Payment Period

Masked text prediction:
付款条件**关联**付款期限
Translation: Payment Condition**'s connected to** Payment Period

Output:
甲方对乙方交付的标的物验收合格后（**付款条件**）**15个工作日内**（**付款期限**），甲方一次付清合同总价款。
Translation: **After Party A accepts the delivered goods from Party B as satisfactory** (**Payment Condition**), Party A shall pay the total contract amount in full **within 15 working days** (**Payment Period**).

Fig. 2. An overview of the proposed GEPL. GEPL incorporates basic cross-domain CEE models with two types of prompt—instance-oriented prompts and label-oriented prompts. We also develop a multi-task training strategy that enhances the model's understanding of the task and captures relationships between different labels across domains.

Given the input instances, instance-oriented prompts are automatically generated by selecting semantically relevant examples from the training data, providing auxiliary supervision to mitigate the data sparsity problem. To further reduce the impact of label discrepancies across different domains, we begin by constructing a label graph based on mutual information criteria. The identified relationships between element labels are then transformed into natural language templates for label-oriented prompts. Moreover, we develop a joint training strategy that refines the representations of both the original input sentence and the prompts, empowering GEPL to better understand the tasks and effectively encode label relations in both source and target domains.

Note that GEPL functions as a fully automated approach, free from the reliance on supplementary hand-labeled data or human interventions. GEPL is capable of seamlessly integrating with various cross-domain CEE and NER frameworks. Experimental results using both the cross-domain NER and CEE datasets demonstrate the effectiveness of GEPL.

The contributions of this article can be summarized as follows:

- To the best of our knowledge, our work is the first to investigate prompt learning for the cross-domain CEE task.
- Given input sentences, we design instance-oriented prompts by retrieving semantically similar examples in train data to alleviate the data sparsity problem.
- To address label discrepancies, we initially employ mutual information criteria to construct a label graph. Subsequently, we transform the extracted relationships between element types into label-oriented prompts, capturing the dependencies among labels across various domains.
- Experimental results reveal that the proposed GEPL model significantly outperforms the baselines in both cross-domain NER and CEE tasks. Furthermore, GEPL exhibits a robust ability to align prompts with input sentences and to effectively model the relationships between various element labels.

The rest of this article is organized as follows: Related work is reviewed in Section 2. The preliminaries and the proposed GEPL framework are detailed in Sections 3 and 4. Evaluations of both cross-domain CEE and NER tasks, along with detailed analyses, are presented in Sections 5 and 6. Finally, our conclusions and future work are formulated in Section 8.

2 Related Work

We survey related work along four dimensions: (i) legal **Information Retrieval (IR)** and **Information Extraction (IE)**, (ii) cross-domain CEE, (iii) cross-domain NER, and (iv) prompt learning.

2.1 Legal IR and IE

The identification of pertinent materials and elements is fundamental in legal practice. However, due to the heavy burden on the worldwide legal system and high dependence on professional knowledge, automatic legal IR and IE systems are urgently needed. In recent years, with the digitization of legal documents, numerous datasets focusing on legal IR and IE tasks, such as COLIEE [57], CAILIE [6], AILA [3], and CUAD [24], have been released. These benchmarks have spurred extensive research in this field [2, 17, 31, 48], and various challenges have been presented [12, 30]. Traditional legal search systems, primarily keyword-based, depend heavily on the user's expertise. Recent advancements aim to reduce this dependency and enhance retrieval effectiveness by automatically classifying legal documents and queries [16, 31, 53, 71]. Additionally, semantic matching methods and user-system interactions have been incorporated to enhance legal case retrieval [46, 61, 65, 66]. Furthermore, current studies indicate that extracting key legal concepts can streamline the retrieval process [4, 70]. In light of this, Chen et al. [11] introduce a triplet extraction system to jointly recognize entities and relations from unstructured crime judgment documents. Martín-Chozas and Revenko [47] use legal thesauri to automatically perform entity annotation and minimize manual efforts by expanding the initial training set of relations. Kwak et al. [32] present an IE dataset focusing on complex legal wills and evaluate an in-context learning-based framework in both in-domain and out-of-domain settings.

In this article, we conduct a detailed investigation into the challenges of data sparsity and label discrepancies in cross-domain CEE.

2.2 Cross-Domain CEE

The CEE task focuses on extracting key legal elements from documents, such as execution dates, jurisdictions, and amounts, as highlighted in prior studies [13, 26]. Initial methods in CEE are predominantly rule-based or employ traditional statistical techniques. Chalkidis et al. [8], for instance, define 11 types of contract elements and implement their extraction using Logistic Regression and SVM, augmented with hand-crafted features. Similarly, García-Constantino et al. [20] develop the CLIEL system to extract core information from commercial law documents, utilizing rule-based techniques for recognizing five distinct types of contract elements. Additionally, Azzopardi et al. [1] propose a hybrid approach that combines regular expressions with a dedicated contract editing tool for legal practitioners. The evolution of CEE methods has recently shifted toward deep learning, approaching the task as a sequence labeling problem. In this context, Chalkidis and Androutsopoulos [7] delve into deep learning, specifically using a BiLSTM model that omits the need for manually crafted rules. Sun et al. [67] classify clauses into seven distinct semantic categories and introduce a TOI pooling layer to manage nested elements. Moreover, Chalkidis et al. [9] reassess the CEE task, focusing on how sequence encoders, CRF layers, and input representations influence extraction outcomes. Nevertheless, a major challenge faced by these CEE methods is how to transfer knowledge from one domain to another [72, 76]. To address this, Wang et al. [72] employ a multi-task framework that conducts both CC and element extraction, integrating invariant knowledge about clauses and element types.

However, as discussed in Section 1, the current cross-domain CEE methods still suffer from challenges of data sparsity and label discrepancies, which seriously impede knowledge transfer from the source domain to the target domain.

2.3 Cross-Domain Named Entity Recognition

Similar to cross-domain CEE, cross-domain NER aims to recognize crucial information (i.e., named entities) in a target domain by using knowledge transferred from a source domain [72]. Recent cross-domain NER methods are designed using either parameter transfer or label representation techniques. Parameter transfer methods focus on modeling domain-invariant features with shared components or auxiliary tasks [10, 27, 28, 33, 35, 36, 39, 56, 62, 76]. For example, Qu et al. [56] model the correlation between source and target entity types with a two-layer neural network. Jia et al. [27] propose a parameter generation network and incorporate the LM task to deal with zero-shot learning settings. Jia and Zhang [28] transfer entity type level knowledge using a multi-cell compositional LSTM structure and model each entity type using a separate cell state. Chen et al. [10] study the augmentation for the cross-domain NER task, modeling patterns (e.g., style, noise, abbreviation) and transforming the data representation from a high-resource to a low-resource domain. Liu et al. [40] collect a new dataset, named CrossNER, containing five diverse domains and propose BERT-based competitive baselines for domain adaptation. Moreover, Hu et al. [25] use subsequence-level features that help the model distinguish different meanings of the same word in different domains. In a line of work on label representation methods, Pan et al. [52] project both labels and features into the same low-dimensional space, fully exploiting relations between labels and reducing distance in data distribution across domains. Liu et al. [39] propose a template regularization framework to enhance the adaptation robustness by regularizing the representation of utterances. To reduce the complexity of the labeling scheme, Zhang et al. [77] and Xu and Cai [73] decompose the monolithic NER task into two sub-tasks: entity span detection and type classification. Tang et al. [69] aim to mitigate entity type conflicts and design a machine reading comprehension-based framework to identify domain-specific semantic differences. Notably, several recent studies have conducted analyses of the performance of current LLMs, such as the GPT series [5, 51], in the context of the NER tasks [21, 22, 68]. Nevertheless, these investigations have revealed a substantial performance gap between recent LLMs and state-of-the-art methods.

Despite their success, cross-domain NER methods cannot be directly applied to the cross-domain CEE task [72] due to two main challenges, namely the need for more granular element type identification and the requirement to handle broader extraction scopes within cross-domain CEE.

2.4 Prompt-Based Learning

Prompting is the practice of adding natural language texts or continuous vectors to the original inputs or outputs to empower LLMs to perform specific tasks. By reformulating downstream **Natural Language Processing (NLP)** tasks, prompting enables a better alignment of the new task formulation with the pretraining objectives (e.g., masked text prediction). In this way, prompt-based methods are able to better use the knowledge captured in the pretraining phases and have shown remarkable performance in low-resource scenarios [5, 19, 34, 37, 44, 64]. For example, given appropriate prompts and only a few input-output pairs, GPT-3 produces the desired outputs for unseen inputs [5]. Scao and Rush [64] also show that a good prompt can be worth hundreds of labeled data points, effectively reducing the number of task-specific training examples required to achieve similar performance to previous approaches or even eliminating the reliance on supervised datasets. Prompt-based learning is currently implemented for a growing list of NLP tasks. For instance, for the NER task in low-resource settings, Lee et al. [34] propose a demonstration-based learning approach that uses automatically constructed auxiliary supervision. Furthermore, Das et al. [14] incorporate contrastive learning techniques with prompts to enhance the capture of label dependencies. For the event extraction task in cross-lingual settings, Fincke et al. [18] present a language-agnostic approach, augmenting the transformer stack's language model differently depending on the question(s) being asked of the model at runtime. Besides, prompts also enable knowledge probing to quantify knowledge presented in the LLMs for the specific tasks of interest [23, 29, 38, 54, 55, 80]. For instance, Jiang et al. [29] investigate model knowledge with discrete prompt templates, while Qin and Eisner [55] apply continuous prompts to factual knowledge probing. Please refer to [37, 50] for a thorough review of prompt-based learning and its applications.

In this article, we mainly focus on the cross-domain CEE task. The work most closely related to ours is [72]. However, the previous cross-domain CEE methods still face two challenging problems: (i) data sparsity and (ii) label discrepancies across domains. To alleviate data sparsity, our proposed GEPL generates instance-oriented prompts by retrieving the most relevant example to the input sentence. To mitigate label mismatches in different domains, we design label-oriented prompts to capture relations among element types across domains. To the best of our knowledge, ours is the first study to explore prompt learning for the cross-domain CEE task.

3 Preliminaries

Prior to presenting the details of the proposed GEPL method, we introduce the problem formulation of cross-domain CEE and three types of basic model used in the experiments.

3.1 Problem Formulation

Following Wang et al. [72], we write $\mathbf{C} = (\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_n)$ for a clause from a contract, where $\mathbf{s}_i$ is the ith sentence in the clause $\mathbf{C}$; $\mathbf{s}_i = (x_{i,1}, x_{i,2}, \dots, x_{i,m})$, where $x_{i,j}$ is the jth word in sentence $\mathbf{s}_i$. A *contract element* $\mathbf{e}$ in the clause $\mathbf{C}$ is a sequence of words in one sentence: $\mathbf{e} = \{(x_{i,start}, x_{i,start+1}, ..., x_{i,end}), \mathbf{l}^e\}$, where $\mathbf{l}^e \in \mathcal{E}$ is the type label of the element $\mathbf{e}$ (such as PP or Payment Condition). The aim of the CEE task is to find the element $\mathbf{e}$ in the clause $\mathbf{C}$. For the cross-domain CEE task, our goal is to transfer the contract element extractor to the target domain from the source domain. Specifically, the extractor is trained on the labeled clauses in the source domain $\mathcal{D}_s$ and the target domain $\mathcal{D}_t$ to detect all the elements for the test set from the target domain.

A similar task is cross-domain NER, which focuses on identifying several types of named entities in a single sentence. In contrast, cross-domain CEE aims at extracting much more fine-grained elements in multiple sentences. As mentioned before, the transfer of larger extraction zones and more element categories brings new challenges to cross-domain CEE over and above cross-domain NER.

3.2 Basic Models

Given that the proposed GEPL method is designed to be model-agnostic, we select three recent cross-domain CEE (or NER) methods as our basic models—BERT [40], Bi-FLEET [72], and MTD [77]. In the following sections, these three types of basic model are outlined briefly. Note that, as mentioned in Section 2, due to their high costs and inferior performance on the NER task, we do not include recent LLMs, such as the GPT series [5, 51], as our basic models

3.2.1 BERT. Following previous work [40, 72], we employ pre-trained language models BERT [15] to conduct the CEE (or NER) task. Specifically, BERT first generates contextualized word embeddings. These embeddings are then input into a linear classifier with a softmax function to predict the probability distribution of element (or entity) types. The process involves feeding each token $x \in \mathbf{s}$ into the feature encoder BERT to obtain the corresponding contextualized word embeddings $\mathbf{h}$:

$$\mathbf{h} = \mathrm{BERT}(\mathbf{s}), \tag{1}$$

where $\mathbf{h}$ represents the sequence of contextualized embeddings derived from the pre-trained language models. To recognize contract elements (or entities), we optimize the following cross-entropy loss $\mathcal{L}_{basic}$ as:

$$\mathcal{L}_{basic} = -\sum_{l^e \in \mathcal{E}} y_{x,e} \log \left(p_{x,e}\right), \tag{2}$$

where N denotes the number of classes, y is a binary indicator (0 or 1) indicating whether the gold label l^e is the correct prediction for observation x, and p is the predicted probability of l^e for x. Following Liu et al. [40], we consider two training settings, namely BERT-PF and BERT-JF, for the cross-domain CEE (or NER) task in our experiments:

- *BERT-PF*. We initially pre-train BERT using data from the source domain and then fine-tune it with samples from the target domain.
- *BERT-JF*. We perform joint fine-tuning of BERT using data samples from both the source and target domains. Given that the data sample size in the target domains is smaller than in the source domain, we apply upsampling to the target domain data. This approach helps to balance the data samples between the source and target domains.

3.2.2 *Bi-FLEET.*

Bi-FLEET [72], as the first work to formulate the cross-domain CEE task, captures domain-invariant knowledge about element types and introduces a bidirectional feedback scheme between the CEE and CC tasks. BI-FLEET contains three main components—a context encoder, a clause-element relation encoder, and an inference layer. The context encoder embeds every word x in a sentence $\mathbf{s}$ from a given clause. The clause-element relation encoder is shared by the source and target domain and calculates representations of clause and element types using a hierarchical graph neural network. The word embeddings and type representations are input to the inference layer for CC and CEE across domains. To simultaneously conduct the CEE and CC tasks, the overall loss function $\mathcal{L}_{basic}$ of Bi-FLEET is defined as follows:

$$\begin{aligned} \mathcal{L}_{CEE} &= -\frac{1}{N}\sum_{n=1}^{N} \log(p(\mathbf{y}_n^{CEE}|\mathbf{s}_n)), \\ \mathcal{L}_{CC} &= -\frac{1}{N}\sum_{n=1}^{N} \mathbf{y}_n^{CC} \log(p(\hat{\mathbf{y}}_n^{CC})), \\ \mathcal{L}_{basic} &= \sum_{d \in \{\mathcal{D}_s, \mathcal{D}_t\}} \lambda^d (\mathcal{L}_{CEE}^d + \lambda^t \mathcal{L}_{CC}^d), \end{aligned} \tag{3}$$

where $\mathcal{L}^{CEE}$ and $\mathcal{L}^{CC}$ are loss functions for the CEE and CC tasks, respectively. N is the size of the training dataset. For the input sentence $\mathbf{s}_n$, $\mathbf{y}^{CC}n$ and $\mathbf{y}^{CEE}n$ represent the ground truth labels for the CC and CEE tasks, respectively. Furthermore, $\hat{\mathbf{y}}^{CC}n$ represents the predicted label for $\mathbf{s}_n$ in the CC task. The output probabilities $p(\mathbf{y}^{CEE}n \mid \mathbf{s}_n)$ and $p(\hat{\mathbf{y}}^{CC}n)$ are calculated by the standard CRF [45] and softmax layers, respectively. λ^d is the domain weight, and λ^t is the task weight. Given that the BERT-based Bi-FLEET demonstrates superior performance in the cross-domain CEE task compared to other variants [72], we choose BERT as the context encoder for Bi-FLEET in our experiments.

3.2.3 *MTD.*

In MTD [77], the NER task is segmented into two subtasks, entity span detection and entity type classification, aimed at reducing the label space and enhancing the transfer process. Specifically, MTD initially employs two distinct BERT-based encoders to extract unique representations for each subtask and then combines these representations to derive the final outcomes. Additionally, to facilitate mutual enhancement between the subtasks, MTD incorporates a modular interaction mechanism for dual-loss reweighting and linguistic consistency learning, along with target-domain adversarial regularization for robust training. To effectively fulfill the cross-domain CEE (or NER) task, given an input sentence $\mathbf{X} = (w_1, \ldots, w_n)$, MTD undergoes joint training in a supervised manner by minimizing the following total loss, $\mathcal{L}_{basic}$:

$$\begin{aligned}
\mathcal{L}_{span} &= -\sum_{i=1}^{n} \gamma_i^{tp} \sum_{k=1}^{c_1} y_{i,t} \log(p(span_t \mid w_i)) \\
\mathcal{L}_{type} &= -\sum_{i=1}^{n} \gamma_i^{sp} \sum_{t=1}^{c_2} y_{i,t} \log(p(type_t \mid w_i)) \\
\mathcal{L}_{basic} &= \mathcal{L}_{span}^{\mathcal{D}_s} + \mathcal{L}_{span}^{\mathcal{D}_t} + \mathcal{L}_{type}^{\mathcal{D}_s} + \mathcal{L}_{type}^{\mathcal{D}_t} \\
&\quad + \lambda \left(\mathcal{L}^{I} + \mathcal{L}_{Sha} + \mathcal{L}^{AT1} + \mathcal{L}^{AT2} \right),
\end{aligned} \tag{4}$$

where $\mathcal{L}_{span}$ and $\mathcal{L}_{type}$ are the loss functions for entity span detection and entity type classification, respectively. γ_i^{tp} and γ_i^{sp} denote the weights for these loss functions. The terms c_1 and c_2 represent the number of span types and entity categories, respectively. $p(span_t \mid w_i)$ and $p(type_t \mid w_i)$ indicate the predicted distributions for entity span and entity type, respectively, while λ is the weight coefficient. Moreover, $\mathcal{L}^I$ and $\mathcal{L}_{Sha}$ are the loss functions for soft labeling and linguistic consistency, respectively. $\mathcal{L}^{AT1}$ and $\mathcal{L}^{AT2}$ pertain to target domain regularizations. For more details, please refer to [77].

4 Method

In this section, we provide a detailed description of the GEPL framework. As Figure 2 illustrates, GEPL incorporates basic models with two types of prompts—(i) instance-oriented prompts and (ii) label-oriented prompts. Specifically, to construct instance-oriented prompts, we retrieve instances in training data that are semantically similar to the given input sentences. Next, a label graph is established based on mutual information criteria, and identified relationships between element labels are then modified into natural language prompt templates. On this basis, both instance-oriented and label-oriented prompts, along with the original sentences, serve as the input to the basic model. Finally, a multi-task joint training algorithm is designed to further fine-tune the representations of both the original input sentence and the prompts.

Below, we first present the process for retrieving and incorporating instance-oriented prompts (Section 4.1). Then, the construction of the label graph and the generation of prompts for the identified relationships between element labels are explained (Section 4.2). Finally, we demonstrate the overall loss function and joint training algorithm (Section 4.3).

4.1 Instance-Oriented Prompts

Given an input sentence $\mathbf{s}$, our aim is to retrieve an instance example $\mathbf{s}_{io}$ that is relevant to the input from both the source and target domains. To this end, we employ the widely used sentence embedding method, SBERT [60], using a BERT-based Siamese network to retrieve semantically similar sentences. By deriving the embeddings of the "[CLS]" token independently for the input $\mathbf{s}$ and $\mathbf{s}_{io}$, we compute the cosine similarity between these two sentence embeddings to rank $\mathbf{s}_{io}$ within $\mathcal{D}_s \cup \mathcal{D}_t$. Following Reimers and Gurevych [60], we use mean squared loss as the objective function for SBERT pretraining. To reduce computational complexity in our experiments, we consider only the relevant instances $\mathbf{s}_{io}$ that contain at least one type of element found in the original input sentence $\mathbf{s}$ and select the training instance $\mathbf{s}_{io}$ with the highest similarity score.

Then, to provide auxiliary supervision to the basic model, we modify the original input $\mathbf{s}$, along with the retrieved relevant instance $\mathbf{s}_{io}$ and its containing elements and labels, denoted as $\{(\mathbf{e}_1, \mathbf{l}_1), \ldots, (\mathbf{e}_E, \mathbf{l}_E)\}$, using the following prompt template function $f_{io}(\cdot)$:

$$f_{io}(\mathbf{s}) = \text{“}\mathtt{[CLS]}\mathbf{s}\mathtt{[SEP]}\mathbf{s}_{io}\mathtt{[SEP]}\mathbf{e}_1 \text{ is } \mathbf{l}_1\mathtt{[SEP]}, \ldots, \mathtt{[SEP]}\mathbf{e}_E \text{ is } \mathbf{l}_E\text{”}. \tag{5}$$

For example, as Figure 2 shows, given the input sentence $\mathbf{s}_1 =$ “After Party A accepts the delivered goods from Party B as satisfactory, ... within 15 working days,” the instance-oriented prompt $f_{io}(\mathbf{s}_1)$ is represented as follows:

$$\begin{aligned} f_{io}(\mathbf{s}_1) = \text{“}&\mathtt{[CLS]}\text{After Party A ... within 15 working days.}\\ &\mathtt{[SEP]}\text{Party A shall ... VAT special invoice.}\mathtt{[SEP]}\\ &\text{After receiving Party B's VAT special invoice}\\ &\text{is Payment Condition}\mathtt{[SEP]}\text{within 15 working}\\ &\text{days is Payment Period”.} \end{aligned} \tag{6}$$

4.2 Label-Oriented Prompts

In this section, we detail the construction of the label graph and the generation of label-oriented prompts.

4.2.1 Label Graph Construction. Given the training dataset $\mathcal{D}_s \cup \mathcal{D}_t$ and the set of element types $\mathcal{E}$, our goal is to construct a label graph that spans across domains and captures the relationships between various element types. To accomplish this, we develop the mutual information criteria to identify dependencies between different element types. Specifically, for a given domain $\mathcal{D}$, we define $\mathcal{D}_i$ to be the set that contains all sentences from the domain $\mathcal{D}$ where elements of type $\mathbf{l}_i \in \mathcal{E}$, and $\mathcal{D}\backslash\mathcal{D}_i$ to be the set that contains sentences without entities of type $\mathbf{l}_i$.

To investigate the relationships between element types $\mathbf{l}_i$ and $\mathbf{l}_j$ within $\mathcal{E}$, we calculate the mutual information between $\mathcal{D}_i$ and all elements of type $\mathbf{l}_j$, as well as between $\mathcal{D}_j$ and all elements of type $\mathbf{l}_i$. Then, we introduce a filtering condition as follows:

$$\frac{C_{\mathcal{D}\backslash\mathcal{D}_i}(\mathbf{l}_j)}{C_{\mathcal{D}_i}(\mathbf{l}_j)} \leq \rho, \quad \frac{C_{\mathcal{D}\backslash\mathcal{D}_j}(\mathbf{l}_i)}{C_{\mathcal{D}_j}(\mathbf{l}_i)} \leq \rho, \quad C_{\mathcal{D}_i}(\mathbf{l}_j), C_{\mathcal{D}_j}(\mathbf{l}_i) > 0, \tag{7}$$

where $C_{\mathcal{D}_i}(l_j)$ denotes the count of elements of type $\mathbf{l}_j$ within $\mathcal{D}_i$. The term $C_{\mathcal{D}\backslash\mathcal{D}_i}(\mathbf{l}_j)$ represents the count of type $\mathbf{l}_j$ elements in all sentences excluding those in $\mathcal{D}_i$. The hyperparameter ρ is defined as the element frequency ratio. Element types $\mathbf{l}_i$ and $\mathbf{l}_j$ are connected in the label graph when they satisfy the mutual information criteria in Equation (7). By applying this condition, we ensure that element types $\mathbf{l}_i$ and $\mathbf{l}_j$ are strongly associated with each other while also being related to other element types in $\mathcal{D}\backslash\mathcal{D}_i$. Given that the number of sentences in $\mathcal{D}_i$ may be much smaller than the number of examples in $\mathcal{D}\backslash\mathcal{D}_i$, we set $\rho \geq 1$ but avoid setting it to a large value. In our experiments, we opt for $\rho = 3$ according to the performance in the validation set. As depicted in Figure 3, we initially identify edges between element types from the source domain (blue and yellow nodes) by setting $\mathcal{D} = \mathcal{D}_s$ in Equation (7). Subsequently, we retain these established edges and incorporate new ones between element types from the target domain (yellow and green nodes) by setting $\mathcal{D} = \mathcal{D}_t$ in Equation (7).

4.2.2 Prompt Generation. To integrate dependencies between element types into the basic model, we generate the label-oriented prompt for each input sentence. Specifically, given an input sentence $\mathbf{s}$ with its containing element types $\{\mathbf{l}_1, \mathbf{l}_2, \ldots, \mathbf{l}_E\}$, our aim is to train the basic model to distinguish the presence or absence of a relationship between any pair of element types found within the sentence $\mathbf{s}$. To this end, we formulate the incorporation of relations between element

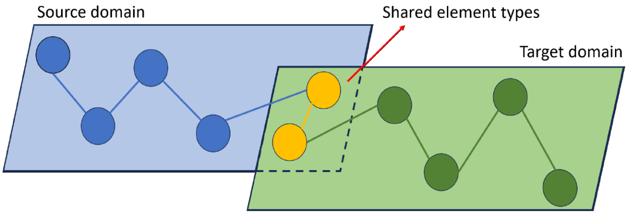


Fig. 3. Label graph construction across domains. Nodes are color-coded to indicate their domain associations: blue nodes represent element types specific to the source domain, yellow nodes denote shared element types across domains, and green nodes correspond to element types specific to the target domain. Edges denote the identified relationships between element types.

types as a cloze-style task for the basic model (see Section 3.2), and the prompt template function $f_{lo}(\cdot)$ with K [MASK] tokens is defined as follows:

$$f_{lo}(\mathbf{s}) = f_{io}(\mathbf{s}) + \text{“[SEP]}\mathbf{l}_1\text{[MASK]}\mathbf{l}_2\text{[SEP]}\ldots\text{[SEP]}\mathbf{l}_{E-1}\text{[MASK]}\mathbf{l}_E\text{”}, \tag{8}$$

where $f_{io}(\mathbf{s})$ is the instance-oriented prompt template function for sentence $\mathbf{s}$ (see Section 4.2). Note that we remove all pairs of element types $\mathbf{l}_j$ and $\mathbf{l}_i$ with $j > i$ to avoid redundancy. During inference, since element type labels of the input sentence are not available, we construct label-oriented prompts with element types appearing in the retrieved relevant instance (see Section 4.1).

For example, as depicted in Figure 2, the label-oriented prompt $f_{lo}(\mathbf{s}_1)$ corresponding to $\mathbf{s}_1$ = “After Party A accepts the delivered goods from Party B as satisfactory, ... within 15 working days.” with the Payment Condition and PP element types can be represented as follows:

$$f_{lo}(\mathbf{s}_1) = f_{io}(\mathbf{s}_1) + \text{“[SEP] Payment Condition[MASK]Payment Period”}. \tag{9}$$

By inputting $f_{lo}(\mathbf{s})$ into the basic model, we obtain the hidden vector $\mathbf{h}_{\text{[MASK]}}$ of “[MASK]”. Following this, we compute the probability that token r can fill the masked position:

$$p(\text{[MASK]} = r | f_{lo}(\mathbf{x}))) = \frac{\exp(\mathbf{r} \cdot \mathbf{h}_{\text{[MASK]}})}{\sum_{\tilde{r}} \exp(\tilde{\mathbf{r}} \cdot \mathbf{h}_{\text{[MASK]}})}, \tag{10}$$

where r and $\tilde{r}$ are two distinct tokens within the label $\mathbf{l}^{\text{[MASK]}}$ for the “[MASK]” tokens, where $\mathbf{l}^{\text{[MASK]}} \in$ {“’s connected to”, “’s not connected to”}. The embeddings $\mathbf{r}$ and $\tilde{\mathbf{r}}$, corresponding to tokens r and $\tilde{r}$, respectively, are generated by the basic model. The token with the highest probability is then selected as the prediction for each input “[MASK]” token.

To train the basic model for selection, we define the loss function $\mathcal{L}_{gen}$ as follows:

$$\mathcal{L}_{gen} = -\frac{1}{|\mathcal{D}_s \cup \mathcal{D}_t|} \sum_{\mathbf{s} \in \mathcal{D}_s \cup \mathcal{D}_t} \sum_{i=1}^{K} \log p(\text{[MASK]}_i = r_i \mid f_{io}(\mathbf{s})), \tag{11}$$

where $r_i \in \mathbf{l}_i^{\text{[MASK]}}$ represents the ground truth token for the ith “[MASK]” token in the prompt $f_{io}(\mathbf{s})$. Here, $\mathbf{l}_i^{\text{[MASK]}}$ specifies the ground truth label for this ith “[MASK]” token.

4.3 Joint Training

To simultaneously refine the representations of the original input and its corresponding prompts, we train our model using a multi-task framework. The overall loss function is defined as follows:

$$\mathcal{L} = (1 - \alpha) \cdot \mathcal{L}'_{basic} + \alpha \cdot \mathcal{L}_{gen}, \tag{12}$$

where $\mathcal{L}'_{basic}$ denotes the normalized loss function for the basic model loss $\mathcal{L}_{basic}$ (see Section 3.2). $1 - \alpha$ is the weight assigned to $\mathcal{L}'_{basic}$ with prompts as inputs. The weight α is assigned to the loss function $\mathcal{L}_{gen}$ for label-oriented prompt generation (see Equation (11)). In our experiments, we optimize the overall loss function using AdamW [41].

During each epoch, GEPL is trained on all sentences from both source and target domains. Note that instance-oriented and label-oriented prompts are generated in the pre-training stage. For instance-oriented prompt generation, cosine similarities between sentences in the training dataset are computed, resulting in a computational complexity of $O(|\mathcal{D}_{\text{train}}|^2)$, where $|\mathcal{D}_{\text{train}}|$ denotes the number of sentences in the training dataset. The computational complexity of label-oriented prompt generation is $O(|\mathcal{D}_{\text{train}}| \cdot l_{\text{avg}} \cdot |\mathcal{L}|)$, where l_{avg} and $\mathcal{L}$ represent the average sentence length and the set of element types, respectively. The mutual information criteria in Equation (7) can be computed for all pairs of element types by traversing the tokens in each training sentence just once. In future work, we plan to retrieve semantically relevant instances by computing cosine similarities between clauses instead of sentences. This strategy would considerably improve the computational efficiency of instance-oriented prompt generation from $O(|\mathcal{D}_{\text{train}}|^2)$ to $O(|\mathcal{D}_{\text{clause}}|^2)$, where $|\mathcal{D}_{\text{clause}}|$ is the number of clauses in the training dataset.

5 Experiments

5.1 Research Questions

We aim to answer the following research questions:

(RQ1) Does GEPL outperform the state-of-the-art methods on the cross-domain CEE taks? (Section 6.1)
(RQ2) Can GEPL be generalized to the cross-domain NER task? (Section 6.2)
(RQ3) How do the instance-oriented and label-oriented prompts contribute to the improvements? (Section 7.1)
(RQ4) How do the amount of target-domain data and loss weight α influence the performance of cross-domain CEE? (Sections 7.2 and 7.3)
(RQ5) Is GEPL able to outperform baselines at element type level and generate appropriate prompts? (Sections 7.4 and 7.5)

5.2 Datasets

In our experiments, our proposed GEPL is evaluated on both the cross-domain CEE and NER datasets. Detailed statistics of the datasets that we use are provided in Table 1.

5.2.1 *Cross-Domain CEE Datasets.*

To conduct cross-domain CEE, we used datasets curated by Wang et al. [72]. Specifically, 340 individual Chinese contracts are gathered from online sources to form the Individual dataset, and 1,422 business Chinese contracts are acquired from various partners to compose the Commercial dataset. Each contract undergoes annotation by at least two legal experts. Initial preprocessing involves segmenting contracts into sentences, excluding elements occurring fewer than 20 times, and removing sentences exceeding 100 characters. Subsequently, the sentences were partitioned into training, validation, and test sets in an 8:1:1 ratio. Following Wang et al. [72], we construct two cross-domain CEE datasets, namely I2C and C2I. For the I2C (or C2I) dataset, the Individual (or Commercial) dataset serves as the source domain, while the other dataset functions as the target domain.

Table 1. Statistics of the Datasets Used

Task	Domain	#Elements or entities	#Train	#Valid	#Test
CEE	Individual	70	13.6K	1.7K	1.7K
	Commercial	79	4.8K	0.6K	0.6K
NER	CoNLL03	4	14,041	-	-
	Politics	9	200	541	651
	Science	17	200	450	543
	Music	13	100	380	465
	Literature	12	100	400	416
	AI	14	100	350	431

5.2.2 Cross-Domain NER Datasets. For cross-domain NER, we leverage the CrossNER datasets curated by Liu et al. [40]. We conduct experiments on five domain pairs, aiming to transfer NER models from the source domain, CoNLL03 (English) [63], to five distinct target domains—Politics, Natural Science, Music, Literature, and Artificial Intelligence [40]. The source domain dataset, CoNLL03, is derived from the Reuters News domain and encompasses four general entity categories—person, **Location (LOC)**, organization, and miscellaneous. In contrast, each target domain introduces domain-specific entity types, such as "Politician" and "Scientist," posing a challenge for model adaptation from a high-resource source domain to a low-resource target domain. In addition, the vocabulary overlaps between domains are generally small, indicating the diversity of the constructed cross-domain NER datasets [40].

5.3 Baselines

In our experiments, we compare our proposed GEPL with the following competitive cross-domain baselines:

- *BiLSTM-CRF* [33] models output dependencies via a simple conditional random field and a transition-based algorithm to explicitly construct and label chunks of the input.
- *Coach* [39] employs a coarse-to-fine detection framework to address the unseen type issue.
- *LM-NER* [27] bridges NER domains by leveraging cross-domain LM and designs a novel parameter generation network for cross-task knowledge transfer.
- *MultiCell-LM* [28] investigates a multi-cell compositional LSTM structure on top of BERT, modeling each entity type using a separate cell state.
- Liu et al. [40] explore two domain adaptation settings for the BERT model. *BERT-JF* jointly fine-tunes BERT on both source and target domain data with upsampling in the target domain, while *BERT-PF* first pretrains BERT on the source domain data and then fine-tunes it on the target domain.
- *Style-NER* [10] projects data representations from a high-resource to a low-resource domain by learning the patterns, such as style, noise, and abbreviation.
- *DoSEA* [69] applies a machine reading comprehension framework, identifying domain-specific information and mitigating entity-type conflicts.
- *BMRU* [25] transfers more fine-grained local information within dense subsequences, distinguishing different meanings of the same word in different domains.

- *Bi-FLEET* [72] pioneers the formulation of the cross-domain CEE task, capturing domain-invariant relations between clauses and elements and implementing a bidirectional feedback scheme between the CEE and CC tasks.
- *MTD* [77] incorporates the modular task decomposition by segregating the monolithic NER into two sub-tasks—entity span detection and entity type classification. This strategy aims to mitigate label space disparity across domains.

In our experiments, we evaluate all the aforementioned baselines for the cross-domain NER task. Following Wang et al. [72], we carefully select representative baselines for cross-domain CEE, including BiLSTM-CRF [33], LM-NER [27], BERT-JF [40], BERT-PF [40], Bi-FLEET [72], and MTD [77], since not all cross-domain NER methods are applicable to the cross-domain CEE task. As mentioned in Section 2, our proposed GEPL is able to seamlessly integrate with various cross-domain NER or CEE frameworks. Consequently, we consider three basic models for our proposed GEPL. Specifically, GEPL (BERT) uses BERT-PF or BERT-JF [40] as the basic model, and we report the best results obtained from our experiments for this model variant. Additionally, GEPL (Bi-FLEET) and GEPL (MTD) employ the state-of-the-art baselines Bi-FLEET [72] and MTD [77] as their basic models, respectively. In addition to the baselines mentioned above, we further compare our proposed GEPL with recent LLMs, specifically GPT-3.5 (`gpt-3.5-turbo-0125`[1]) and Qwen2.5 (`Qwen2.5-7B-Instruct`[2]) (see Section 6.1).

5.4 Evaluation Metrics

Following previous work [28, 40, 67, 72, 77], we assess Precision (P), Recall (R), and F1-score (F1) at the entity (or element) level. Precision signifies the percentage of named entities (or elements) correctly identified by the method. Recall indicates the percentage of entities (or elements) within the datasets that the method successfully predicts. In this context, an entity (or element) is deemed correct only when it exactly matches the corresponding entity (or element) in the dataset. The F1-score represents the harmonic mean of Precision and Recall. In our experiments, we present P, R, and F1 for the cross-domain CEE task and F1 for the cross-domain NER task.

5.5 Implementation Details

Our parameter settings mainly follow prior studies [72, 77]. In line with Wang et al. [72], we use the base-sized BERT [15], pretrained on the Chinese Wikipedia corpus, and employ the NCRF++ toolkit [75] for the cross-domain CEE. The AdamW optimizer [41] is used to optimize the overall loss function $\mathcal{L}$, as defined in Equation (12), with a warmup ratio of 0.1. We experiment with different learning rates, exploring values from the set {1e-5, 3e-5, 5e-5}, to find the optimal rate for various model variants and tasks. The batch size is set to 30 for cross-domain CEE and 4 for cross-domain NER. To determine the optimal loss weight α, we conduct a search over the range 0.1, 0.25, 0.5, 0.75, 0.9, based on validation set performance, and find that the optimal value for α is 0.5. The maximum input and output lengths for all model variants are standardized at 256, and the frequency ratio hyperparameter ρ is consistently set to 3 across all tasks.

6 Experimental Results

To answer RQ1 and RQ2, we assess the performance of GEPL on both cross-domain CEE and NER tasks.

[1] https://platform.openai.com/docs/models/gpt-3-5-turbo
[2] https://huggingface.co/Qwen/Qwen2.5-7B-Instruct

Table 2. Precision, Recall, and F1-Scores on the Two Cross-Domain CEE Datasets [72]

Model	I2C			C2I		
	P	R	F1	P	R	F1
BiLSTM-CRF [33]	63.21	67.50	65.28	65.30	70.36	67.73
LM-NER [27]	62.57	66.96	64.69	65.45	70.84	68.04
BERT-JF [40]	67.57	71.25	69.36	68.72	75.70	72.04
BERT-PF [40]	67.42	71.33	69.32	68.72	75.70	72.41
Bi-FLEET [72]	70.17	73.75	71.92	70.71	78.69	74.49
MTD [77]	72.78	74.96	73.85	71.13	78.66	74.71
GPT-3.5	40.35	47.19	43.50	38.62	35.49	36.99
Qwen2.5	36.15	40.53	38.21	30.56	33.95	32.17
GEPL (BERT)	71.70*	74.15*	72.90*	72.20*	78.84*	75.37*
GEPL (Bi-FLEET)	73.50*	75.73*	74.60*	73.35*	80.84*	76.91*
GEPL (MTD)	**74.19***	**77.21***	**75.67***	**74.48***	**80.12***	**77.20***

Significant improvements against the corresponding basic models are marked with $*$ (t-test, $p < 0.05$). The bold numbers represent the highest performance for each metric.

6.1 Cross-Domain CEE (RQ1)

We turn to RQ1. Following Wang et al. [72], we compare GEPL with the state-of-the-art baselines on the cross-domain CEE task. Table 2 shows the experimental results on the I2C and C2I datasets, where Precision (P), Recall (R), and F1-score (F1) are reported. Based on Table 2, we have the following observations:

- The cross-domain CEE task is challenging, and most baseline models struggle to achieve an F1-score above 70%. This observation aligns with the conclusion mentioned by Wang et al. [72]. In sharp contrast, our proposed framework, GEPL, can effectively transfer contract element extractors from one domain to another. By leveraging our proposed GEPL, models consistently surpass an F1-score of 70% on both the I2C and C2I datasets, demonstrating their ability to effectively transfer knowledge across different contract domains.
- GEPL delivers substantial performance improvements over the baselines. Notably, GEPL (BERT), GEPL (Bi-FLEET), and GEPL (MTD) consistently outperform corresponding basic models (i.e., BERT, Bi-FLEET, and MTD) in terms of Precision (P), Recall (R), and F1-score (F1). For instance, GEPL (MTD) exhibits a 5.79% improvement over MTD on average. This remarkable improvement demonstrates the superiority of GEPL in CEE.
- To assess the influence of different base models on performance, we examine three base models, including BERT [40], Bi-FLEET [72], and MTD [77]. The results, presented in Table 2, suggest a correlation between the chosen base model and the overall performance of GEPL. For instance, on both the I2C and C2I datasets, GEPL based on BERT demonstrates the lowest F1-scores, whereas GEPL based on MTD achieves the highest performance. This performance difference may be attributed to MTD surpassing both BERT-JF and BERT-PF by a significant margin. This observation sheds light on the crucial role played by the selection of the basic model.
- Furthermore, we compare GEPL with recent LLMs, specifically GPT-3.5 (`gpt-3.5-turbo-0125`) and Qwen2.5 (`Qwen2.5-7B-Instruct`). To implement LLMs for cross-domain CEE, we

Table 3. F1-Scores on the CrossNER Dataset [40]

	Politics	Science	Music	Literature	AI
BiLSTM-CRF [33]	56.60	49.97	44.79	43.03	43.56
Coach [39]	61.50	52.09	51.66	48.35	45.15
LM-NER [27]	68.44	64.31	63.56	59.59	53.70
BERT-JF [40]	68.85	65.03	67.59	62.57	58.57
BERT-PF [40]	68.71	64.94	68.30	63.63	58.88
MultiCell-LM [28]	70.56	66.42	70.52	66.96	58.28
Style-NER [10]	68.78	63.95	65.43	60.94	58.73
DoSEA [69]	75.52	71.69	73.10	68.59	66.03
BMRU [25]	71.31	68.65	72.42	67.05	60.89
Bi-FLEET [72]	70.57	66.63	71.53	67.22	58.74
MTD [77]	75.53**	71.51**	76.10**	69.22**	68.07**
GEPL (BERT)	73.56*	70.09*	75.33*	69.45*	61.36*
GEPL (Bi-FLEET)	74.67*	71.91*	77.21*	69.86*	63.35*
GEPL (MTD)	**77.56***	**73.48***	**77.40***	**72.05***	**68.72***

Significant improvements against the corresponding basic models are marked with $*$ (t-test, $p < 0.05$). Reproduced results for MTD are highlighted with $**$. The bold numbers represent the highest performance for each metric.

employ few-shot in-context learning [22], prompting the LLMs to extract the given contract element types from input sentences and using semantically relevant instances retrieved in Section 4.1 as task demonstrations. Based on the results in Table 2, we observe that LLMs achieve substantially lower precision, recall, and F1-scores compared to smaller fine-tuned models, including our proposed GEPL and even BERT-based baselines. Two possible reasons for this outcome are as follows: (i) Large gaps exist between cross-domain CEE and the pretraining tasks of LLMs. (ii) Without costly fine-tuning, LLMs lack specialized knowledge for cross-domain CEE.

In summary, GEPL shows its effectiveness in recognizing contract elements across various domains. The incorporation of instance-oriented and label-oriented prompts is beneficial for cross-domain CEE.

6.2 Cross-Domain NER (RQ2)

To investigate the generalizability of GEPL, we move on to RQ2 and conduct experiments on the cross-domain NER task. Table 3 presents the experimental results on the CrossNER dataset [40]. In line with previous studies [28, 77], F1-scores are employed to assess the overall performance. For the MTD baseline, we employ our reproduced results, derived from the source code provided by Zhang et al. [77], which are highlighted with "$**$". Based on the results in Tables 2 and 3, we arrive at the following conclusions:

- GEPL demonstrates strong generalizability, enabling effective transfer of named entity recognizers across diverse domains. Notably, GEPL-based models achieve the state-of-the-art performance on five distinctive target NER domains, considerably exceeding an F1-score of 70%. For example, GEPL (MTD) attains F1-scores of 77.56% and 77.40% on the Politics and Music domains, respectively.
- GEPL significantly outperforms the state-of-the-art cross-domain NER methods. Specifically, GEPL (MTD) achieves the highest F1-score across five target domains. Additionally, GEPL

Table 4. Precision, Recall, and F1-Scores on the Two Cross-Domain CEE Datasets

Model	I2C			C2I			Science	Music
	P	R	F1	P	R	F1	F1	F1
GEPL (BERT)	**71.70**	**74.15**	**72.90**	**72.20**	**78.84**	**75.37**	**70.09**	**75.33**
- lo prompts	69.29	72.75	70.98	71.23	77.68	74.31	69.27	74.72
- io prompts	67.57	71.25	69.36	68.72	75.70	72.04	65.03	67.59
GEPL (Bi-FLEET)	**73.50**	**75.73**	**74.60**	**73.35**	**80.84**	**76.91**	**71.91**	**77.21**
- lo prompts	71.56	74.71	73.10	71.67	79.77	75.50	68.86	72.68
- io prompts	70.17	73.75	71.92	70.71	78.69	74.49	66.63	71.53
GEPL (MTD)	**74.19**	**77.21**	**75.67**	**74.48**	**80.12**	**77.20**	**73.48**	**77.40**
- lo prompts	73.67	75.35	74.50	73.90	80.08	76.87	72.89	76.86
- io prompts	72.78	74.96	73.85	71.13	78.66	74.71	71.51	76.10

The bold numbers represent the highest performance for each metric.

gains substantial improvements compared to the corresponding basic NER methods. For instance, GEPL(BERT), GEPL (Bi-FLEET), and GEPL (MTD) achieve substantial F1-score boosts over BERT, Bi-FLEET, and MTD of 7.58%, 6.66%, and 2.40% on average, respectively.

— GEPL's overall performance is highly influenced by the capabilities of its backbone models. For both cross-domain CEE and NER tasks, GEPL achieves higher F1-scores with a stronger backbone model. For example, on the cross-domain CEE datasets, MTD outperforms BERT by an average of 3.86% in F1-scores, while GEPL (MTD) achieves an average 3.12% increase in F1-scores over GEPL (BERT). Therefore, selecting an appropriate backbone model is crucial for enhancing GEPL's effectiveness across domains.

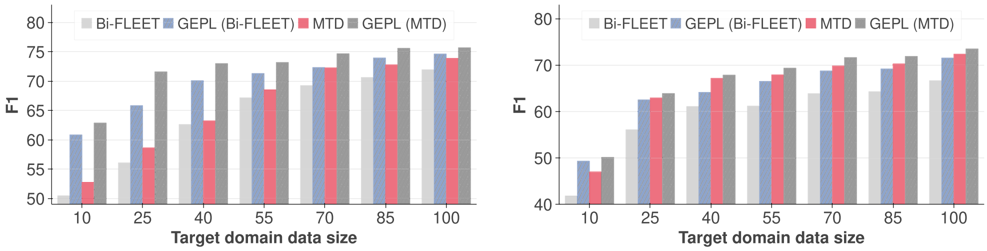
In conclusion, our proposed GEPL exhibits strong generalizability. The GEPL framework not only effectively addresses the cross-domain CEE problem but also consistently delivers the state-of-the-art experimental results on the cross-domain NER task.

7 Analysis

Now that we have answered our research questions, we take a closer the look at GEPL to analyze its performance. We examine how instance-oriented and label-oriented prompts contribute to its performance, how the amount of target-domain data influences the performance, and how performance varies across element types.

7.1 Ablation Studies (RQ3)

To delve into the individual contributions of each component to GEPL's performance, we conduct ablation studies on a range of cross-domain CEE and NER datasets. Similar to the experimental setting employed in Sections 6.1 and 6.2, our evaluation includes three basic models—BERT, Bi-FLEET, and MTD. The outcomes for the I2C, C2I, Science, and Music datasets are detailed in Table 4. In tests where label-oriented prompts are purposefully excluded (indicated as "- lo prompts"), these prompts are omitted from the input of the basic models, and the loss associated with masked text prediction in Equation (12) is also discarded. This exclusion leads to a significant decrease in GEPL's performance across all evaluated metrics. For instance, on the I2C dataset, GEPL (MTD) surpasses its model variant without label-oriented prompts by 2.47% in Recall and 1.57% in F1-score. In scenarios where we ablate both the label-oriented and instance-oriented prompts (indicated as "- io prompts"), our models revert to their basic forms (i.e., BERT, Bi-FLEET, and MTD). Based on the



(a) I2C.

(b) Science.

Fig. 4. Influence of target domain data size on the I2C and Science datasets.

results, we find that the integration of instance-oriented prompts into GEPL significantly enhances performance compared to the basic models in all settings. For example, GEPL (MTD) with only instance-oriented prompts achieves F1-score improvements of 2.89%, 1.93%, and 1.00% over the basic MTD model on the C2I, Science, and Music datasets.

In summary, the inclusion of both label-oriented and instance-oriented prompts plays a crucial role in elevating GEPL's effectiveness in cross-domain CEE and NER tasks.

7.2 Influence of Target-Domain Data (RQ4)

Next, we investigate the impact of target-domain data size on the cross-domain CEE and NER tasks. We compare the F1-scores of the baselines and GEPL, using different amounts of target-domain data ranging from 10% to 100% of the original training set. Based on the results in Figure 4, we find that GEPL outperforms baselines with different target domain data sizes. Initially, GEPL demonstrates a substantial improvement of over 10% compared to the baselines. For example, GEPL (MTD) achieves 19.13% F1-score improvements over the baseline MTD with only 10% target domain data on the I2C dataset. As the amount of target-domain data increases, the performance gap between GEPL and the baselines becomes smaller. For example, GEPL (MTD) improves the F1-score on the I2C dataset by 3.89% over MTD, training on 85% of the target domain training data. Notably, both GEPL (Bi-FLEET) and GEPL (MTD) consistently outperform the baselines across different amounts of target-domain data, showcasing the effectiveness of our proposed methods.

7.3 Influence of Loss Weight (RQ4)

To analyze the influence of the loss weight α in Equation (12), we vary α from 0.1 to 0.9 to observe performance changes in GEPL with different backbones on the I2C dataset. The results are shown in Figure 5. We observe that as α increases, GEPL's F1-scores initially improve, as a larger α enables the model to better encode dependencies between element types across domains. For example, GEPL (MTD) with $\alpha = 0.5$ achieves 2.84% higher F1-scores compared to GEPL (MTD) with $\alpha = 0.1$. However, performance declines when α becomes too large. Consequently, in our experiments, we set GEPL's optimal loss weight α to 0.5.

7.4 Fined-Grained Comparisons (RQ5)

To thoroughly assess the performance of GEPL at the element or entity type level, we conduct fine-grained comparisons on the I2C, Science, and Music datasets. The results for the cross-domain CEE and NER tasks are presented in Table 5, where we examined five contract element types—**Invoice Type (IT)**, **Arbitration Commission (AC)**, **Currency of Payment (CP)**, **Name of Subject Matter (NSM)**, and PP, and four entity types—LOC, **Country (COU)**, **Band (BAN)**, and **Album (ALB)**. Among these, IT, CP, PP, and LOC are shared across both the source and target domains,

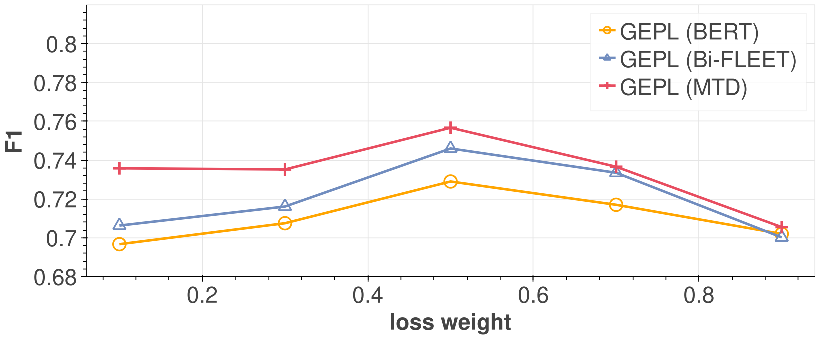


Fig. 5. Influence of loss weight α on the I2C dataset.

Table 5. Fine-Grained Comparisons of F1-Scores on I2C, Science, and Music

Model	I2C					Science		Music	
	IT	AC	CP	NSM	PP	LOC	COU	BAN	ALB
Bi-FLEET [72]	68.41	70.63	71.53	72.73	72.80	68.21	70.17	78.62	69.10
MTD [77]	69.18	70.13	72.24	73.74	74.25	75.50	76.90	80.45	69.48
GEPL (Bi-FLEET)	69.46	73.52	73.80	75.07	76.32	72.67	74.56	80.37	74.53
GEPL (MTD)	**71.35**	**74.18**	**74.93**	**75.90**	**77.10**	**80.72**	**80.00**	**82.05**	**75.41**

The bold numbers represent the highest performance for each metric.

while AC, COU, BAN, and ALB are specific types exclusive to the target domains. Based on the findings in Table 5, GEPL demonstrates strong effectiveness in identifying both shared and domain-specific types. For instance, GEPL (MTD) achieves F1-scores of 77.10% on the PP type and 74.18% on the AC type, respectively. Furthermore, models based on GEPL consistently outperformed baselines. Specifically, GEPL (MTD) achieved significant F1-score improvements of 3.85% and 8.54% on the PP and ALB types, respectively, compared to the state-of-the-art baseline MTD. These observations underscore the efficacy of leveraging instance-oriented and label-oriented prompts, effectively mitigating data sparsity and minimizing label discrepancies across domains.

7.5 Case Studies (RQ5)

To study whether GEPL can generate appropriate prompts from original input sentences, we conducted case studies on the I2C and Science datasets, as shown in Table 6. The first two examples in the table are from the I2C dataset, while the last is from the Science dataset. The results demonstrate that GEPL is able to identify instance-oriented prompts with similar semantics and relevant elements, effectively providing auxiliary supervision and mitigating the issue of data sparsity. For instance, the original input sentence in the first example involves a penalty clause with legal elements of Penalty Payment Ratio and Penalty Payment Reference. Given the input sentence, GEPL successfully retrieves an instance-oriented prompt that also includes the Penalty Payment Ratio (i.e., “5%”) and Penalty Payment Reference (i.e., “total contract value”). This illustrates GEPL’s capability to closely align the prompts with the original input contexts. Moreover, GEPL shows proficiency in identifying and modeling the relationships between different element labels, thereby considerably reducing label discrepancies across various domains. A notable example is its construction of a label-oriented prompt, namely “Penalty Payment Ratio`[MASK]`Payment Period,” in the

Table 6. Examples from the I2C and Science Datasets

Original input sentence	Instance-oriented prompt	Label-oriented prompt
"(iii) If Party B's appointed personnel fail to participate in the contract's technical services or are unilaterally changed by Party B, Party B must pay a penalty of **5%** (Penalty Payment Ratio) of the **total technical service fee** (Penalty Payment Reference) to Party A."	"(ii) If Party B's delivered goods fail to meet the contract's terms, hindering Party A's intended purpose, Party B must pay Party A a penalty of **5%** (Penalty Payment Ratio) of the **total contract value** (Penalty Payment Reference)."	"Penalty Payment Ratio `[MASK]` Penalty Payment Reference"
"**After Party A accepts the delivered goods from Party B as satisfactory** (Payment Condition), Party A shall pay the total contract amount in full **within 15 working days** (Payment Period)."	"Party A shall make a one-time payment for the electricity to Party B **within 15 working days** (Payment Period) **after receiving Party B's VAT special invoice** (Payment Condition)."	"Payment Condition `[MASK]`Payment Period"
"In 1917, he was appointed as the first **Palit Professor of Physics** (Award) by **Ashutosh Mukherjee** (Scientist) at the **Rajabazar Science College** (University)."	"From 1916 to 1921, he was a lecturer in the physics department of the **Rajabazar Science College** (University)."	"Award`[MASK]`Scientist`[SEP]` Award`[MASK]`University`[SEP]` Scientist`[MASK]`University"

Table 7. Error Analysis on Sentence Lengths in Test Sets

		Sentence length					Sentence length			
Method	Dataset	< 25	25–50	> 50	Avg.	Dataset	< 25	25–50	> 50	Avg.
GEPL (BERT)		68.88	70.04	74.86	72.90		72.22	73.38	77.49	75.37
GEPL (Bi-FLEET)	I2C	70.96	72.51	77.14	74.60	C2I	73.10	74.20	79.03	76.91
GEPL (MTD)		72.78	73.47	77.87	75.67		75.82	76.28	79.56	77.20

The scores are F1-scores.

second case. By filling in the "`[MASK]`" token, the basic model is capable of better comprehending the interdependencies between the Penalty Condition and PP legal elements.

7.6 Error Analysis

Although GEPL outperforms the state-of-the-art baselines, it is important to understand where it fails. Specifically, we compare the performance of GEPL on sentences of different lengths in the test sets of the I2C and C2I datasets. The results based on BERT, Bi-FLEET, and MTD are presented in Table 7. We observe that GEPL's F1-scores on sentences with more than 50 characters (> 50) are substantially higher than the average F1-scores. For instance, on the I2C dataset, GEPL (MTD) achieves an F1-score of 77.87%, which is 4.25% higher than the overall F1-score (77.10%). In contrast, the F1-scores on sentences with 25 to 50 characters (25–50) or less than 25 characters (< 25) consistently fall below the average F1-scores. For example, on the I2C dataset, GEPL (MTD) achieves F1-scores of 72.78% on sentences with 25 to 50 characters and 73.47% on sentences with less than 25 characters, which are considerably lower than the overall F1-score (77.10%). This suggests that GEPL faces greater challenges in generating appropriate instance-oriented and label-oriented prompts for input instances when presented with less context.

8 Conclusions

In this article, we have investigated the problem of cross-domain CEE, aiming to leverage knowledge from a source domain to enhance the extraction of legally relevant elements in a target domain. Prior work on cross-domain CEE still faces two major challenges: data scarcity due to the high cost of annotations and label discrepancies across various contract domains. To overcome these problems, we have proposed a novel framework named GEPL. GEPL effectively mitigates data sparsity by generating auxiliary supervisions for each input instance and bridges the gap between contract domains by automatically identifying label relations across distinct domains.

In our experiments, we conduct a comprehensive evaluation of GEPL on both cross-domain CEE and NER tasks. Experimental results underscore that GEPL significantly outperforms the state-of-the-art baselines and demonstrates a high degree of generalizability across the two cross-domain CEE datasets and the five cross-domain CEE datasets. Additionally, by incorporating instance-oriented and label-oriented prompts, GEPL effectively recognizes both shared and domain-specific element or entity types, highlighting its robustness to data sparsity and ability to capture relations for various labels across domains.

Building upon the current study, we envisage four lines of future work—(i) It is worth exploring the prompt generation method developed for sentences with limited context. Our error analysis revealed that GEPL struggles to retrieve semantically relevant instances or to generate effective label-oriented prompts for input instances with limited context (see Section 7.6 for details). To address this limitation, one solution could be to use the clauses in which the input instances appear to obtain appropriate prompts. Another solution would be to treat preceding and following sentences as additional context for the input sentences. These strategies could provide more context-rich prompts for short input instances, potentially improving performance in these cases. (ii) In addition to prompting relationships between contract element labels, we intend to incorporate prompts that capture clause-element relations [72]. This would allow GEPL to leverage both domain-specific and domain-invariant features, further enhancing its cross-domain transfer capabilities. (iii) Our experiments focused on datasets from contract domains. To broaden the applicability of GEPL, we plan to study its connection with other legally relevant tasks (e.g., legal judgment prediction [42, 43, 78, 79]) and investigate how to leverage other legal contexts to further mitigate data sparsity and improve generalizability. (iv) Last, apart from the basic models mentioned in the experiments, we also plan to incorporate GEPL with different kinds of LLMs to further enhance its performance.

Acknowledgments

We thank our reviewers for their constructive feedback. All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

References

[1] Shaun Azzopardi, Albert Gatt, and Gordon J. Pace. 2016. Integrating natural language and formal analysis for legal documents. In *LTDH*, 1–4.

[2] Trevor J. M. Bench-Capon, Michal Araszkiewicz, Kevin D. Ashley, Katie Atkinson, Floris Bex, Filipe Borges, Danièle Bourcier, Paul Bourgine, Jack G. Conrad, Enrico Francesconi, et al. 2012. A history of AI and law in 50 papers: 25 Years of the international conference on AI and law. *Artif. Intell. Law* 20, 3 (2012), 215–319.

[3] Paheli Bhattacharya, Kripabandhu Ghosh, Saptarshi Ghosh, Arindam Pal, Parth Mehta, Arnab Bhattacharya, and Prasenjit Majumder. 2019. FIRE 2019 AILA track: Artificial intelligence for legal assistance. In *FIRE*, 4–6.

[4] Lukasz Borchmann, Dawid Wisniewski, Andrzej Gretkowski, Izabela Kosmala, Dawid Jurkiewicz, Lukasz Szalkiewicz, Gabriela Palka, Karol Kaczmarek, Agnieszka Kaliska, and Filip Gralinski. 2020. Contract discovery: Dataset and a few-shot semantic retrieval challenge with competitive baselines. In *EMNLP*, 4254–4268.

[5] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are Few-Shot learners. In *NeurIPS*.

[6] Yu Cao, Yuanyuan Sun, Ce Xu, Chunnan Li, Jinming Du, and Hongfei Lin. 2022. CAILIE 1.0: A dataset for challenge of AI in law—Information extraction V1.0. *AI Open* 3 (2022), 208–212. DOI: https://doi.org/10.1016/j.aiopen.2022.12.002

[7] Ilias Chalkidis and Ion Androutsopoulos. 2017. A deep learning approach to contract element extraction. In *JURIX*, 155–164.

[8] Ilias Chalkidis, Ion Androutsopoulos, and Achilleas Michos. 2017. Extracting contract elements. In *ICAIL*, 19–28.

[9] Ilias Chalkidis, Manos Fergadiotis, Prodromos Malakasiotis, and Ion Androutsopoulos. 2019. Neural contract element extraction revisited. In *NeurIPS*.

[10] Shuguang Chen, Gustavo Aguilar, Leonardo Neves, and Thamar Solorio. 2021. Data augmentation for cross-domain named entity recognition. In *EMNLP*, 5346–5356.

[11] Yanguang Chen, Yuanyuan Sun, Zhihao Yang, and Hongfei Lin. 2020. Joint entity and relation extraction for legal documents with legal feature enhancement. In *COLING*, 1561–1571.

[12] Heting Chu. 2011. Factors affecting relevance judgment: A report from TREC legal track. *J. Documentation* 67, 2 (2011), 264–278. DOI: https://doi.org/10.1108/00220411111109467

[13] Michael Curtotti and Eric Mccreath. 2010. Corpus based classification of text in Australian contracts. In *ALTA*.

[14] Sarkar Snigdha Sarathi Das, Arzoo Katiyar, Rebecca J. Passonneau, and Rui Zhang. 2022. CONTaiNER: Few-shot named entity recognition via contrastive learning. In *ACL*, 6338–6353.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*, 4171–4186.

[16] Phong-Khac Do, Huy-Tien Nguyen, Chien-Xuan Tran, Minh-Tien Nguyen, and Minh-Le Nguyen. 2017. Legal question answering using ranking SVM and deep convolutional neural network. arXiv:1703.05320. Retrieved from https://arxiv.org/abs/1703.05320

[17] John Doyle. 1992. WESTLAW and the American digest classification scheme. *Law. Libr. J.* 84 (1992), 229.

[18] Steven Fincke, Shantanu Agarwal, Scott Miller, and Elizabeth Boschee. 2022. Language model priming for cross-lingual event extraction. In *AAAI*, 10627–10635. DOI: https://doi.org/10.1609/aaai.v36i10.21307

[19] Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *ACL-IJCNLP*, 3816–3830.

[20] Matías García-Constantino, Katie Atkinson, Danushka Bollegala, Karl Chapman, Frans Coenen, Claire Roberts, and Katy Robson. 2017. CLIEL: Context-based information extraction from commercial law documents. In *ICAIL*, 79–87.

[21] Bernal Jimenez Gutierrez, Nikolas McNeal, Clayton Washington, You Chen, Lang Li, Huan Sun, and Yu Su. 2022. Thinking about GPT-3 in-context learning for biomedical IE? Think again. In *EMNLP*, 4497–4512.

[22] Ridong Han, Tao Peng, Chaohao Yang, Benyou Wang, Lu Liu, and Xiang Wan. 2023. Is information extraction solved by ChatGPT? An analysis of performance, evaluation criteria, robustness and errors. arXiv:2305.14450. Retrieved from https://arxiv.org/abs/2305.14450

[23] Adi Haviv, Jonathan Berant, and Amir Globerson. 2021. BERTese: Learning to speak to BERT. In *EACL*, 3618–3623.

[24] Dan Hendrycks, Collin Burns, Anya Chen, and Spencer Ball. 2021. CUAD: An expert-annotated NLP dataset for legal contract review. In *NeurIPS Datasets and Benchmarks*.

[25] Jinpeng Hu, Dandan Guo, Yang Liu, Zhuo Li, Zhihong Chen, Xiang Wan, and Tsung-Hui Chang. 2023. A simple yet effective subsequence-enhanced approach for cross-domain NER. In *AAAI*, 12890–12898. DOI: https://doi.org/10.1609/aaai.v37i11.26515

[26] Kishore Varma Indukuri and P. Radha Krishna. 2010. Mining E-contract documents to classify clauses. In *COMPUTE*, 1–5.

[27] Chen Jia, Liang Xiao, and Yue Zhang. 2019. Cross-domain NER using cross-domain language modeling. In *ACL*, 2464–2474.

[28] Chen Jia and Yue Zhang. 2020. Multi-Cell compositional LSTM for NER domain adaptation. In *ACL*, 5906–5917.

[29] Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. 2021. How can we know *when* language models know? On the calibration of language models for question answering. *Trans. Assoc. Comput. Linguist.* 9 (2021), 962–977. DOI: https://doi.org/10.1162/tacl_a_00407

[30] Yoshinobu Kano, Mi-Young Kim, Randy Goebel, and Ken Satoh. 2017. Overview of COLIEE 2017. In *COLIEE*, 1–8.

[31] Phi Manh Kien, Ha-Thanh Nguyen, Ngo Xuan Bach, Vu Tran, Minh Le Nguyen, and Tu Minh Phuong. 2020. Answering legal questions by learning neural attentive text representation. In *COLING*, 988–998.

[32] Alice Saebom Kwak, Cheonkam Jeong, Gaetano Forte, Derek E. Bambauer, Clayton T. Morrison, and Mihai Surdeanu. 2023. Information extraction from legal wills: How well does GPT-4 do? In *EMNLP*, 4336–4353.

[33] Guillaume Lample, Miguel Ballesteros, Sandeep Subramanian, Kazuya Kawakami, and Chris Dyer. 2016. Neural architectures for named entity recognition. In *NAACL*, 260–270.

[34] Dong-Ho Lee, Akshen Kadakia, Kangmin Tan, Mahak Agarwal, Xinyu Feng, Takashi Shibuya, Ryosuke Mitani, Toshiyuki Sekiya, Jay Pujara, and Xiang Ren. 2022. Good examples make a faster learner: Simple demonstration-based learning for low-resource NER. In *ACL*, 2687–2700.

[35] Ji Young Lee, Franck Dernoncourt, and Peter Szolovits. 2018. Transfer learning for named-entity recognition with neural networks. In *LREC*.

[36] Bill Yuchen Lin and Wei Lu. 2018. Neural adaptation layers for cross-domain named entity recognition. In *EMNLP*, 2012–2022.

[37] Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv*. 55, 9 (2023), Article 195, 1–35. DOI: https://doi.org/10.1145/3560815

[38] Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021. GPT understands, too. arXiv:2103.10385. Retrieved from https://arxiv.org/abs/2103.10385

[39] Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020. Coach: A coarse-to-fine approach for cross-domain slot filling. In *ACL*, 19–25.

[40] Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2021. CrossNER: Evaluating cross-domain named entity recognition. In *AAAI*, 13452–13460. DOI: https://doi.org/10.1609/aaai.v35i15.17587

[41] Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR*.

[42] Yougang Lyu, Jitai Hao, Zihan Wang, Kai Zhao, Shen Gao, Pengjie Ren, Zhumin Chen, Fang Wang, and Zhaochun Ren. 2023. Multi-defendant legal judgment prediction via hierarchical reasoning. In *EMNLP*, 2198–2209.

[43] Yougang Lyu, Zihan Wang, Zhaochun Ren, Pengjie Ren, Zhumin Chen, Xiaozhong Liu, Yujun Li, Hongsong Li, and Hongye Song. 2022. Improving legal judgment prediction through reinforced criminal element extraction. *Inf. Process. Manag*. 59, 1 (2022), 102780.

[44] Ruotian Ma, Xin Zhou, Tao Gui, Yiding Tan, Linyang Li, Qi Zhang, and Xuanjing Huang. 2022. Template-free prompt tuning for few-shot NER. In *NAACL*, 5721–5732.

[45] Xuezhe Ma and Eduard H. Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *ACL*.

[46] Arpan Mandal, Kripabandhu Ghosh, Saptarshi Ghosh, and Sekhar Mandal. 2021. Unsupervised approaches for measuring textual similarity between legal court case reports. *Artif. Intell. Law* 29, 3 (2021), 417–451. DOI: https://doi.org/10.1007/s10506-020-09280-2

[47] Patricia Martín-Chozas and Artem Revenko. 2021. Thesaurus enhanced extraction of Hohfeld's relations from Spanish labour law. In *DeepOntoNLP 2021*, 30–38.

[48] K. Tamsin Maxwell and Burkhard Schafer. 2008. Concept and context in legal information retrieval. In *JURIX*, 63–72.

[49] Zoran Milosevic, Simon Gibson, Peter F. Linington, James Cole, and Sachin Kulkarni. 2004. On design and implementation of a contract monitoring facility. In *WEC*, 62–70.

[50] Bonan Min, Hayley Ross, Elior Sulem, Amir Pouran Ben Veyseh, Thien Huu Nguyen, Oscar Sainz, Eneko Agirre, Ilana Heintz, and Dan Roth. 2024. Recent advances in natural language processing via large pre-trained language models: A survey. *ACM Comput. Surv*. 56, 2 (2024), Article 30, 1–40. DOI: https://doi.org/10.1145/3605943

[51] OpenAI. 2023. GPT-4 technical report. arXiv:2303.08774. Retrieved from https://arxiv.org/abs/2303.08774

[52] Sinno Jialin Pan, Zhiqiang Toh, and Jian Su. 2013. Transfer joint embedding for cross-domain named entity recognition. *ACM Trans. Inf. Syst*. 31, 2 (2013), 1–27. DOI: https://doi.org/10.1145/2457465.2457467

[53] Amal Shehan Perera, Vindula Jayawardana, Dimuthu Lakmal, and Madhavi Perera. 2018. Legal document retrieval using document vector embeddings and deep learning. In *CCIC*, 160–175.

[54] Fabio Petroni, Patrick S. H. Lewis, Aleksandra Piktus, Tim Rocktäschel, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. 2020. How context affects language models' factual predictions. In *AKBC*.

[55] Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying LMs with mixtures of soft prompts. In *NAACL-HLT*, 5203–5212.

[56] Lizhen Qu, Gabriela Ferraro, Liyuan Zhou, Weiwei Hou, and Timothy Baldwin. 2016. Named entity recognition for novel types by transfer learning. In *EMNLP*, 899–905.

[57] Juliano Rabelo, Mi-Young Kim, Randy Goebel, Masaharu Yoshioka, Yoshinobu Kano, and Ken Satoh. 2019. A summary of the COLIEE 2019 competition. In *JSAI-isAI*, 34–49.

[58] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever. 2018. Improving language understanding by generative pre-training. In *OpenAI*.

[59] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever. 2019. Language models are unsupervised multitask learners. In *OpenAI*.

[60] Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-Networks. In *EMNLP-IJCNLP*, 3982–3992.

[61] Julien Rossi and Evangelos Kanoulas. 2019. Legal information retrieval with generalized language models. In *COLIEE*.

[62] Devendra Singh Sachan, Pengtao Xie, Mrinmaya Sachan, and Eric P. Xing. 2018. Effective use of bidirectional language modeling for transfer learning in biomedical named entity recognition. In *MLHC*, 383–402.

[63] Erik F. Tjong, Kim Sang, and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-Independent named entity recognition. In *CoNLL*, 142–147.

[64] Teven Le Scao and Alexander M. Rush. 2021. How many data points is a prompt worth? In *NAACL-HLT*, 2627–2636.

[65] Yunqiu Shao, Jiaxin Mao, Yiqun Liu, Weizhi Ma, Ken Satoh, Min Zhang, and Shaoping Ma. 2020. BERT-PLI: Modeling paragraph-level interactions for legal case retrieval. In *IJCAI*, 3501–3507.

[66] Yunqiu Shao, Yueyue Wu, Yiqun Liu, Jiaxin Mao, and Shaoping Ma. 2023. Understanding relevance judgments in legal case retrieval. *ACM Trans. Inf. Syst.* 41, 3 (2023), 1–32. DOI: https://doi.org/10.1145/3569929

[67] Lin Sun, Kai Zhang, Fule Ji, and Zhenhua Yang. 2019. TOI-CNN: A solution of information extraction on Chinese insurance policy. In *NAACL-HLT*, 174–181.

[68] Xiaofei Sun, Linfeng Dong, Xiaoya Li, Zhen Wan, Shuhe Wang, Tianwei Zhang, Jiwei Li, Fei Cheng, Lingjuan Lyu, Fei Wu, et al. 2023. Pushing the limits of ChatGPT on NLP tasks. arXiv:2306.09719. Retrieved from https://arxiv.org/abs/2306.09719

[69] Minghao Tang, Peng Zhang, Yongquan He, Yongxiu Xu, Chengpeng Chao, and Hongbo Xu. 2022. DoSEA: A domain-specific entity-aware framework for Cross-Domain named entity recognition. In *COLING*, 2147–2156.

[70] Vu D. Tran, Minh Le Nguyen, and Ken Satoh. 2019. Building legal case retrieval systems with lexical matching and summarization using a pre-trained phrase scoring model. In *ICAIL*, 275–282. DOI: https://doi.org/10.1145/3322640.3326740

[71] Vu D. Tran, Minh Le Nguyen, Satoshi Tojo, and Ken Satoh. 2020. Encoded summarization: Summarizing documents into continuous vector space for legal case retrieval. *Artif. Intell. Law* 28, 4 (2020), 441–467. DOI: https://doi.org/10.1007/s10506-020-09262-4

[72] Zihan Wang, Hongye Song, Zhaochun Ren, Pengjie Ren, Zhumin Chen, Xiaozhong Liu, Hongsong Li, and Maarten de Rijke. 2021. Cross-domain contract element extraction with a bi-directional feedback clause-element relation network. In *SIGIR*, 1003–1012.

[73] Jingyun Xu and Yi Cai. 2023. Decoupled hyperbolic graph attention network for cross-domain named entity recognition. In *SIGIR*, 591–600.

[74] Weiwen Xu, Yang Deng, Wenqiang Lei, Wenlong Zhao, Tat-Seng Chua, and Wai Lam. 2022. ConReader: Exploring implicit relations in contracts for contract clause extraction. In *EMNLP*, 2581–2594.

[75] Jie Yang and Yue Zhang. 2018. NCRF++: An open-source neural sequence labeling toolkit. In *ACL*, 74–79.

[76] Zhilin Yang, Ruslan Salakhutdinov, and William W. Cohen. 2017. Transfer learning for sequence tagging with hierarchical recurrent networks. In *ICLR*.

[77] Xinghua Zhang, Bowen Yu, Yubin Wang, Tingwen Liu, Taoyu Su, and Hongbo Xu. 2022. Exploring modular task decomposition in cross-domain named entity recognition. In *SIGIR*, 301–311.

[78] Haoxi Zhong, Zhipeng Guo, Cunchao Tu, Chaojun Xiao, Zhiyuan Liu, and Maosong Sun. 2018. Legal judgment prediction via topological learning. In *EMNLP*, 3540–3549.

[79] Haoxi Zhong, Chaojun Xiao, Cunchao Tu, Tianyang Zhang, Zhiyuan Liu, and Maosong Sun. 2020. How does NLP benefit legal system: A summary of legal artificial intelligence. In *ACL*, 5218–5230.

[80] Zexuan Zhong, Dan Friedman, and Danqi Chen. 2021. Factual probing is [MASK]: learning vs. learning to recall. In *NAACL-HLT*, 5017–5033.

Received 12 February 2024; revised 27 October 2024; accepted 16 December 2024