Unsupervised Legal Evidence Retrieval via Contrastive Learning with Approximate Aggregated Positive

Feng Yao[1], **Jingyuan Zhang**[2*], **Yating Zhang**[2], **Xiaozhong Liu**[3], **Changlong Sun**[2], **Yun Liu**[1*], **Weixing Shen**[1*]

[1]School of Law, Institute for AI and Law, Tsinghua University, Beijing, China
[2]DAMO Academy, Alibaba Group, Hangzhou, Zhejiang, China
[3]Worcester Polytechnic Institute, MA, USA
yaof20@mails.tsinghua.edu.cn, weishi.zjy@alibaba-inc.cn
{liuyun89, wxshen}@tsinghua.edu.cn

Abstract

Verifying the facts alleged by prosecutors before the trial requires the judges to retrieve evidence within the massive materials accompanied. Existing Legal AI applications often assume the facts are already determined and fail to notice the difficulty of reconstructing them. To build practical Legal AI applications and free judges from the manual searching work, we introduce the task of Legal Evidence Retrieval, which aims to automatically retrieve precise fact-related verbal evidence within a single case. We formulate the task in a dense retrieval paradigm and jointly learn the contrastive representations and alignments between facts and evidence. To avoid tedious annotations, we construct an approximated positive vector for a given fact by aggregating a set of evidence from the same case. An entropy-based denoising technique is further applied to mitigate the impact of false positive samples. We train our models on tens of thousands of unlabeled cases and evaluate them on a labeled dataset containing 919 cases and $4,336$ queries. Experimental results indicate that our approach is effective and outperforms other state-of-the-art representation and retrieval models. The dataset and code are available at https://github.com/yaof20/LER.

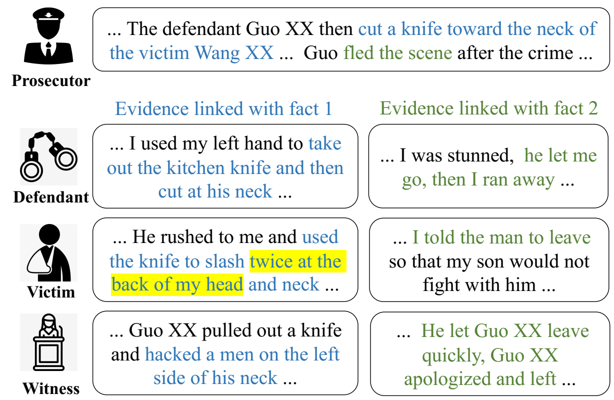

Figure 1: The way judges linked the facts written by the prosecutor with the verbal evidence from multiple resources. Facts and evidence in the same color are relevant to each other. The highlighted evidence contradicts the facts above.

1 Introduction

Linking each fact with the relevant evidence is an essential step for the judge to make findings of fact, and it is the precondition of application of law and the foundation of legal judgment. In judicial practice, the facts and evidence for the same case tend to be submitted in separate files and are not linked with each other, which may cost the judges a lot of time to retrieve relevant evidence to validate the authenticity of each fact. Though tremendous advances have been made in Legal AI, such as Legal Information Extraction (Chen et al. 2020; Yao et al. 2022), Legal Case Retrieval (Ma et al. 2021, 2022) and Legal Judgment Prediction (Zhong et al. 2018), little attention has been paid to evidence-related research and most existing works assume the facts determined by the judges, ignoring the expensive cost behind it.

In this work, we introduce the task of **L**egal **E**vidence **R**etrieval (**LER**), which aims to automatically retrieve the relevant evidence given a fact description within a case. Specifically, we focus on the retrieval of sentence-level verbal evidence in criminal cases, where there tend to be multiple participants of different roles involved and their narratives are in various styles, making LER a challenging yet practical Legal AI task. Figure 1 shows an example of how the judge linked the prosecuted facts with the verbal evidence from different parties. Some pieces of the facts can be simultaneously mentioned by the defendant, victim, and witness. Therefore, the task of LER can be formulated to retrieve the relevant evidence by querying with any piece of the prosecuted facts or verbal evidence. The former type of query can help the judges find the fact-relevant evidence and verify the authenticity of the prosecuted facts, and the latter can be useful to identify the underlying conflicts between relevant evidence. By convention, the prosecutors have to summarize the testimony from different litigants and restate the facts in a formal document before the court. Therefore, there exists a certain semantic gap between the prosecuted facts and the verbal evidence, which distinguishes LER from the traditional retrieval problem that can be well-handled by the conventional models utilizing word co-occurrence between the queries and candidates.

*Corresponding authors.

To facilitate the research of LER, we propose a large-scale dataset named LERD, consisting of more than $300k$ fact queries and over 11 million "query (fact) and candidate (evidence)" pairs, within which $4,436$ queries and their corresponding $234,693$ candidates are annotated with the relevance ranking scores.

Considering the versatility of the model and the huge cost of data annotations, the default setting of LER task is unsupervised with unlabeled data for training and annotated ones for evaluation. We also provide a split of annotated dataset in case of the need for supervised training.

Due to the vocabulary mismatch problem in LER task, we formulate our task in the dense retrieval paradigm (Karpukhin et al. 2020; Sciavolino et al. 2021; Zhang et al. 2022), where the facts and verbal evidence are encoded into dense embeddings by pretrained models, and the retrieval is conducted in the dense representation space.

The most challenging part of dense retrieval without supervision is to construct a positive sample for a given query. Previous works handle this problem by sub-sequence sampling that: (1) generating two non-overlapping spans from the same document as positive (Lee, Chang, and Toutanova 2019), (2) randomly sampling two arbitrary continuous spans that may overlap with each other as positive (Izacard et al. 2021b), (3) recurring spans across passages in a document to create pseudo positives (Ram et al. 2022). All of these strategies are designed for document-level retrieval tasks and the assumption is that any two sub-sequences sampled from the same document are positive to each other. However, LER is a fine-grained sentence-level retrieval task where only the relevant evidence and fact are positive to each other and the rest sub-sequences of the case are negatives.

To tackle the challenges mentioned above, we propose **S**tructure-a**W**are contrastive learning with **A**pproximate aggregated **P**ositive (**SWAP**) which leverages the legal case structure information to construct approximate positives and sample negatives. Based on the premise that the true positive evidence for a given fact query must be within the same case, we construct an approximate positive for each fact by aggregating the representations of all the evidence from the same case. Then, we sample negatives from both inner-case facts and inter-case evidence, and adopt contrastive learning to pull together the positives and push apart the negatives in the representation vector space. Finally, considering that the approximated positives are generated by aggregating the potential samples and can be noisy, we explore an entropy-based denoising technique to reduce the influence of false positives and negatives during training.

Extensive experiments are conducted on LERD, and the results indicate that LER is a challenging task and our proposed method SWAP significantly outperforms state-of-the-art methods. We summarize our contributions as follows:

- We introduce a novel task of Legal Evidence Retrieval (LER), which is a challenging yet practical task with promising value for real-world Legal AI applications. A large-scale dataset is proposed with fine-grained relevance ranking annotations as well as a coarse parallel fact-evidence aligned corpus.
- We propose a novel framework for unsupervised dense retrieval that constructs positive and negative samples with case structure knowledge injected. A denoising approach based on entropy theory is further introduced to mitigate the influence brought by the false positives among the approximated samples.
- Extensive experiments show the effectiveness of our approach and we substantially outperform other strong sparse and dense retrieval baselines. To motivate other scholars, the dataset and code are publicly available.

2 Task and Dataset

2.1 Task Definition

Given a prosecuted facts collection $F_k = \{f_i^k\}_{i=1}^{m_k}$ and a verbal evidence collection $E_k = \{e_j^k\}_{j=1}^{n_k}$ from the same case k, the task of Legal Evidence Retrieval (LER) is to find and rank the relevant evidence e within E_k for each fact f from F_k. The fact f is the concise description of what happened in the case, formally written by the prosecutor in third person. While the evidence e is the verbose record of oral statements by case participants (victim, defendant, witness) in first person. Both f and e are sentences, and there can be zero, one, or multiple evidence relevant to a given fact query. The unique point is that different queries from the same case can be highly similar since they reveal the same crime in general, increasing the difficulty of query understanding.

LER task is mainly faced with the following challenges: (1) **Expression Mismatch.** To keep the reliability and better restore the truth, the evidence is directly quoted from the oral statements of the case participants, which are verbose and less informative than the concise facts, resulting in the semantic gap between them. (2) **Fine Granularity.** LER is targeted at retrieving the relevant sentence-level verbal evidence from multiple resources and requires a fine-grained relevance annotation between facts and evidence. Whereas the common IR task focuses more on document-level retrieval and fails to highlight fine-grained informative statements. (3) **Dynamic Retrieval Pools.** In most IR tasks, the candidate pool is the same for each query. Therefore, the representation of the documents in the candidate pool can be computed offline in advance. However, the evidence pool in LER task varies from case to case and the facts and evidence from different cases are irrelevant by nature.

2.2 Data Construction

In this section, we described in detail the construction of the Dataset for LER task (named LERD). In order to collect data sets aligned with facts and evidence, we found in judgment documents that judges usually cite the facts described by prosecutors and the testimony of each party (an example is shown in Figure 1). Therein, we collect the judgment documents of the criminal cases from the public legal judgment document website[1] as the document pool. To enable model training and testing, we further create a large unsupervised training data set and a relatively small supervised data set for evaluation or weakly supervised training.

[1]https://wenshu.court.gov.cn/

Task	#Query	#Can./que.	#Que-can. pair	#Char./que.	#Pos./que.	#Case	#Crime	Granularity
LeCaRD	107	100	10,700	444.58	10.33	10,700	20	Document-level
LERD-usp	308,749	35	11,079,998	63.22	–	35,423	255	Sentence-level
LERD-sup	4,336	54	234,693	67.63	3.15	919	91	Sentence-level

Table 1: The statistics of LERD and LeCaRD datasets. The suffix '-usp' and '-sup' indicate the unsupervised and labeled parts. 'Can.', 'Char.', and 'Pos.' are short for candidate, character and positive, the relevant candidate to a query. '/que.' denotes value per query, and – means the value is not applicable since there is no annotation for the unsupervised part.

Unsupervised Dataset. A paragraph-level labeling model is trained to parse the crawled judgment documents into several semantic segments (e.g. litigant description, plaintiff's claim, fact identification, evidence description, rationale, and judgment). Then, we utilize regular expressions to refine the prosecuted facts and verbal evidence, and split them into non-overlapping sentences. We filter out the cases with few facts and evidence, thus obtaining a collection of $36,423$ pairs of facts and evidence.

Supervised Dataset. For quantitative evaluation of LER task, we also construct a corpus with fine-grained annotations of the relevance between evidence and each query, which can be also employed for weakly supervised training. We randomly extract $1,000$ cases from the raw judgment documents excluding the ones in unsupervised dataset. We then invite 10 lawyers to annotate the evidence ranking by the relevance to each prosecuted fact. Each case is firstly annotated by two lawyers independently, and a third lawyer is required to handle the disagreement. The criteria to rate the relevance scores between a fact-evidence pair are as follows:

- **Score 2**, Highly Relevant: The occurring time described in evidence (if any), the participants and the types of the events[2] mentioned in the evidence are very similar to the ones in the fact.
- **Score 1**, Partially Relevant: The occurring time described in evidence (if any), the participants and the types of the events mentioned in the evidence are partially matched with the ones in the fact.
- **Score 0**, Irrelevant: The occurring time, the participants or the types of the events mentioned in the evidence are completely different from the ones in the fact.

In this paper, we mainly utilize unsupervised data for training and supervised data for evaluation. We also extend the experiments to the supervised setting (see Sec.5.1) to explore the different usage of our dataset.

2.3 Data Analysis

To better understand the proposed dataset LERD, we make a comparison with another legal domain dataset LeCaRD (Ma et al. 2021), which is commonly adopted in the scenario of legal case retrieval. The detailed statistics are shown in Table 1. It can be observed that our data focuses on fine-grained retrieval and covers a wide range of types of crimes. As for the supervised part, LERD contains more queries and query-candidate pair annotations than LeCaRD, which contributes to a more reliable evaluation. Moreover, the unsupervised data contains richer types of crimes and a large number of cases with facts and evidence parallels, serving as a valuable resource for the unsupervised solutions to LER.

[2]Typically the key actions involved, like steal, bodily-harm, etc.

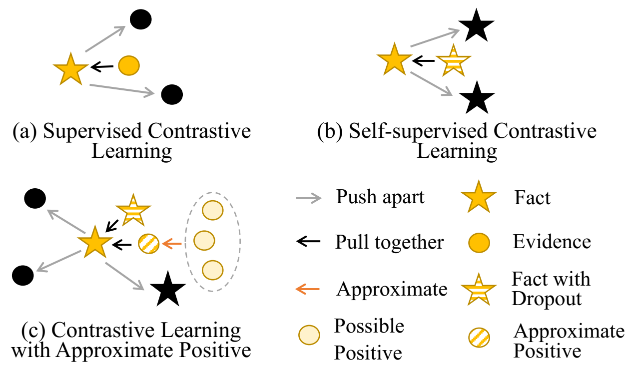

Figure 2: Illustration of the contrastive learning framework with different positive and negative settings. Stars and circles in black are ground-truth negatives.

3 Preliminaries

The followings are preliminaries about our model architecture and training strategies:

Bi-Encoder Architecture The bi-encoder architecture consists of a query encoder ENC_Q and a document encoder ENC_D to map sparse queries and documents into separate dense vectors, and leverages similarity function to measure their relevance (Karpukhin et al. 2020; Izacard et al. 2021b; Ram et al. 2022). For the LER task, we denote the fact encoder and evidence encoder as ENC_f and ENC_e respectively, which are both Transformer encoders. For an input fact f_i^k of case k, the encoder produces a sequence of hidden states and leverages a pooling layer (e.g. averaging) to obtain a vector $\hat{f}_i^k \in \mathcal{R}^d$ as the dense representation. The vector $\hat{e}_i^k$ for each evidence e_i^k is produced in the same way using ENC_e. The cosine similarity function is typically utilized to measure the similarity between the fact f and evidence e as follows:

$$sim(f,e) = \frac{ENC_f(f) \cdot ENC_e(e)}{\|ENC_f(f)\| \|ENC_e(e)\|} \tag{1}$$

Contrastive Learning for Retrieval Contrastive Learning (CL) is a type of technique that pulls together embeddings of related data pairs and pushes away irrelevant

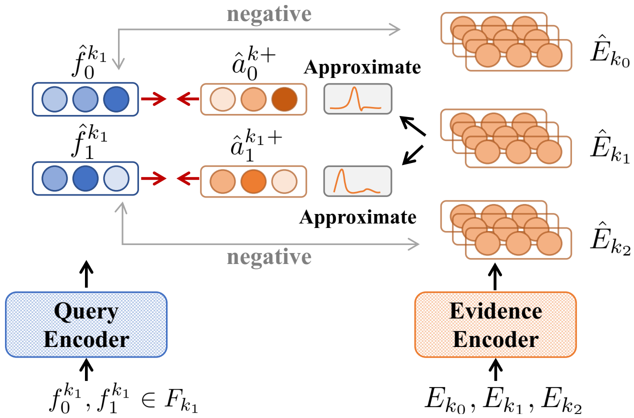

Figure 3: Illustration of the contrastive learning framework with approximate aggregated positive samples. Corroborating facts and approximate positive samples inside a case are pulled together as shown with red arrows.

ones. Under this paradigm, given a fact f_i, its relevant evidence e_i^+ and a set of r irrelevant evidence $E_i^- = \{e_{i,j}^-\}^r$, the contrastive optimization object is to minimize:

$$\mathcal{L}^{\mathrm{C}} = -log\frac{\exp^{sim(f_i,e_i^+)/\tau}}{\exp^{sim(f_i,e_i^+)/\tau} + \sum_{j=1}^{r}\exp^{sim(f_i,e_{i,j}^-)/\tau}} \tag{2}$$

where τ is the temperature parameter ranging from 0 to 1. For the sake of simplicity, we abstract Equation 2 as $Contra(f, f^+, N^-)$ where f, f^+ and N^- denotes anchor fact, positive sample, and a set of negative samples, respectively. As shown in Figure 2, supervised CL (a) utilizes ground-truth relevant evidence as positive sample and irrelevant evidence as negatives, while self-supervised model (b) may leverage augmented data points as positive and other facts as negatives. We further discuss the positive and negative sampling strategies of SWAP (c) in the next section.

4 Method

We formulate the LER task in the Dense Retrieval (DR) paradigm, and propose a structure-aware contrastive learning framework. We first introduce a procedure to construct approximate positive samples in unsupervised settings and then present the method to integrate both positive and negative samples in the contrastive learning framework. A denoising technique to alleviate the negative impacts of generated samples will be discussed in subsection 4.3.

4.1 Construct Positive Instances

For unsupervised dense retrieval, the e_i^+ in Equation 2 is not readily available. Previous unsupervised methods solve this problem by sub-sequence sampling (Lee, Chang, and Toutanova 2019; Izacard et al. 2021b; Ram et al. 2022), which treats the sub-sequence sampled from the same document as a positive instance. However, these sorts of text-based positive building strategies are not applicable to our task where most of the facts and evidence from the same document (case) are not necessarily positive to each other.

Therefore, we propose to construct representation-level positives to jointly learn the contrastive representations and alignment between the facts and evidence. To simplify the notations, we denote $\hat{F}_k = \{\hat{f}_i^k\}_{i=1}^m$ and $\hat{E}_k = \{\hat{e}_j^k\}_{j=1}^n$ as collections of d-dimensional dense vector representations of facts and evidence.

Dropout Positive Inspired by the great success achieved by (Gao, Yao, and Chen 2021), we feed the same input to the encoder twice to obtain two representations with different dropout (Srivastava et al. 2014) masks, and treat one of them as the positive instance to the other. Using the dropout positive in contrastive learning leads to a strong and robust representation of the input text. In our implementations, we build dropout positive $f_{i,dp}^{k+}$ for fact f_i^k and $e_{j,dp}^{k+}$ for evidence e_j^k simultaneously. We refer to the instance constructed by this strategy as Dropout Positive (DP) for simplification.

Approximate Aggregated Positive Though the dropout positives can provide powerful representations of the facts and evidence, the problem of not having a labeled positive e_i^+ for the fact f_i remains unsolved. Fortunately, we notice that the true positive e_i^{k+} for the fact f_i^k is doomed to be within the evidence collection $E_k = \{e_j^k\}_{j=1}^n$ that from the same case k by nature. Therefore, we propose to construct an approximate a_i^{k+} through aggregating the representations of all e_j^k in $E_k = \{e_j^k\}_{j=1}^n$. We denote the approximate positive as AP for short, and the vector $\hat{a}_i^{k+}$ of a_i^{k+} for f_i^k is as calculated by the following equation:

$$\hat{a}_i^{k+} = \sum_{j=1}^{n}\frac{\mathrm{e}^{\hat{f}_i^k\cdot\hat{e}_j^k}}{\sum_{l=1}^{m}\mathrm{e}^{\hat{f}_i^k\cdot\hat{e}_l^k}}\cdot\hat{e}_j^k \tag{3}$$

4.2 Structure-Aware Contrastive Learning

Since the facts and evidence from the same case are relevant in general, we propose a structure-aware contrastive learning framework that considers both the inner-case and inter-case structure when sampling positives and negatives. To keep the case structure information, we use case-level examples during training. Assume the mini-batch size is B, the input fact and evidence examples in the mini-batch are $\{F_1, \cdots, F_B\}$ and $\{E_1, \cdots, E_B\}$, where $F_k = \{f_i^k\}_{i=1}^{m_k}$ and $E_k = \{e_j^k\}_{j=1}^{n_k}$. The training loss consists of two terms regarding the dropout positive and approximate aggregated positive respectively.

We first construct the dropout positives $f_{i,dp}^{k+}$ for fact f_i^k and $e_{j,dp}^{k+}$ for evidence e_j^k for each fact and evidence in the mini-batch. For negative sampling, we consider both in-case and out-case negatives that come from other cases respectively. Take the fact f_i^k for example, the in-case and out-case negatives are denoted in Equation 4 and 5 respectively,

$$\mathcal{N}_{f_i^k} = \{f_x^k\}_{x=1,x\neq i}^{m^k} \tag{4}$$

$$\mathcal{U}_{f_i^k} = \{\{f_x^y\}_{x=1}^{m^y}\}_{y=1,y\neq k}^{B} \tag{5}$$

where m^k and m^y denote the number of evidence in the k-th and the y-th case, respectively.

Training loss regarding the Dropout Positive (DP) for fact f_i^k is calculated by:

$$\mathcal{L}^{\text{DP}}_{f_i^k} = Contra(f_i^k, f_{i,dp}^{k+}, [\mathcal{N}_{f_i^k}; \mathcal{U}_{f_i^k}]) \tag{6}$$

where $[;]$ denotes merging two collections of vectors. The calculation of the loss $\mathcal{L}^{\text{DP}}_{e_j^k}$ for evidence e_j^k is the same as $\mathcal{L}^{\text{DP}}_{f_i^k}$.The overall loss regarding the dropout positive is defined in Equation 7. Note that different from the original implementation (Gao, Yao, and Chen 2021) of contrastive learning with dropout positive where all of the sentences are mixed up for training, we keep the facts and evidence apart and calculate the loss from them separately.

$$\mathcal{L}^{\text{DP}} = \sum_{k=1}^{B} \sum_{i=1}^{m^k} \mathcal{L}^{\text{DP}}_{f_i^k} + \sum_{k=1}^{B} \sum_{j=1}^{n^k} \mathcal{L}^{\text{DP}}_{e_j^k} \tag{7}$$

Secondly, we build the approximate aggregated positive a_i^{k+} for each fact f_i^k by Equation 3. The negatives sampled in this part also include in-case negatives $\mathcal{N}_{a_i^k}$ and out-case negatives $\mathcal{U}_{a_i^k}$ which share the forms in Equation 4 and 5.

The loss $\mathcal{L}^{\text{AP}}_{a_i^k}$ concerning the Approximated Positive (AP) is calculated by the following equation:

$$\mathcal{L}^{\text{AP}}_{f_i^k} = Contra(f_i^k, a_i^{k+}, [\mathcal{N}_{a_i^k}; \mathcal{U}_{a_i^k}]) \tag{8}$$

The loss with respect to the Approximated Positive (AP) is calculated by:

$$\mathcal{L}^{\text{AP}} = \sum_{k=1}^{B} \sum_{i=1}^{m_k} \mathcal{L}^{\text{AP}}_{a_i^k} \tag{9}$$

The final optimization object of the structure-aware contrastive learning framework is:

$$\mathcal{L} = \mathcal{L}^{\text{DP}} + \mathcal{L}^{\text{AP}} \tag{10}$$

4.3 Instance Denoising

There are two underlying problems with the proposed structure-aware contrastive learning framework, which are (1) **False Positive**: the approximate aggregated positive is built for each fact in the case, but there can be no relevant evidence involved for some of the facts in the training data; (2) **False Negative:** the second part of the objective function $\mathcal{L}^{\text{AP}}$ involves the in-case negatives $N_{I_{a,i}}^{k}$ for each fact f_i^k. As mentioned in Section 2.1, a fraction of the facts from the same case can be highly similar. Therefore, the approximate aggregated positives generated by them might be nearly identical, which are not necessarily negative to f_i^k.

To handle these problems, we introduce an entropy-based denoising method that lowers the weights of the false positives and false negatives when computing the loss. The entropy we adopt here is the uniformity of the weights used for aggregating evidence to approximate a positive instance which is used in Equation 3. The updated weight of the approximate positive a_i^{k+} for loss calculation is defined as:

$$w_i^k = \sqrt{\sum_{j=1}^{n} \left(\frac{\mathrm{e}^{f_i^k \cdot e_j^k}}{\sum_{l=1}^{n} \mathrm{e}^{f_i^k \cdot e_l^k}} \right)^2} \tag{11}$$

The intuition behind is that a close approximation of the true positive should be dominated by the relevant evidence rather than the averaging of all evidence. Hence, when computing the loss, we decrease the importance of those false approximate positives contributed by all evidence evenly.

Since the false in-case negatives are nearly indistinguishable, we set their weights to zero for loss calculation. The loss $\mathcal{L}_{a_i^k}$ with instance denoising is defined as:

$$L^{\text{DE}}_{a_i^k} = -\log \frac{w_i^k \cdot \mathrm{e}^{sim(f_i^k, a_i^{k+})/\tau}}{w_i^k \cdot \mathrm{e}^{sim(f_i^k, a_i^{k+})/\tau} + \sum\limits_{a_j \in \mathcal{U}_{a_i^k}} w_j^k \cdot \mathrm{e}^{sim(f_i^k, a_j)/\tau}} \tag{12}$$

5 Experiments

5.1 Experiment Settings

Dataset We conducted experiments on LERD in both unsupervised and supervised settings. Specifically, in the unsupervised setting, we use LERD-usp for training and split LERD-sup into valid and test sets for evaluation. And we split LERD-sup into train, valid, and test sets for the supervised experiments. The statistics of the data splits in both settings are shown in Table 2.

Setting	Split	#Query	#Que-can.	#Case	#Crime
Usp	train	308,749	11,079,998	35,423	255
	valid	943	57,017	200	44
	test	3,393	177,676	719	84
Sup	train	2,940	154,257	619	78
	valid	453	23,419	100	34
	test	943	57,017	200	44

Table 2: The data splits for experiments. 'Que-can.' is short for query-candidate pair. 'Usp' and 'Sup' indicate supervised and unsupervised settings, respectively.

Model We employ the bi-encoder architecture that consists of a fact encoder Enc_F and an evidence encoder Enc_E, both of which are Transformers-based models. For the main experiments, we initialize the encoders with RoBERTa-base-Chinese checkpoint (Cui et al. 2020) and the parameters are shared between them. The dense representations of the facts and evidence are obtained by the average pooling strategy. We use cosine similarity as the function to measure the similarity between the fact and evidence representations. We also conduct experiments with different backbones and pooling strategies to verify the effectiveness and robustness of our proposed methods. The experimental results and detailed analysis are discussed in Section 5.2 and 5.3.

Training During the training stage for SWAP, we use case-level examples to retain the structure information of each case in the mini-batch. We randomly sample cases from

Category	Method	MAP	MRR	R@1	R@3	R@5	NDCG@1	NDCG@3	NDCG@5
Sparse Retrieval	BM25	39.03	45.83	29.03	38.20	48.29	31.10	35.29	39.75
	Legal-Event-IR	37.25	45.29	30.12	36.71	44.52	32.45	35.15	38.49
Text Representation	BERT	47.97	58.15	43.77	49.46	57.25	46.16	47.64	50.58
	RoBERTa	51.63	61.62	47.51	53.11	61.44	50.10	51.28	54.58
	LawFormer	52.25	62.52	48.78	54.05	61.73	51.40	52.42	55.23
	SBERT$^{\circ}$	40.51	50.13	34.39	40.52	49.19	36.63	38.61	42.11
	SimCSE*	56.09	66.39	52.99	58.73	66.06	55.60	56.54	59.29
Dense Retrieval	Contriever$^{\circ}$	45.44	56.11	41.37	46.56	55.18	43.64	44.65	48.08
	Contriever(MS)$^{\circ}$	53.67	64.68	50.89	55.52	64.02	53.50	53.88	57.09
	Condensor*	54.65	64.82	51.01	57.14	64.40	53.57	55.08	57.80
	SWAP-BERT(ours)	59.65	69.58	56.68	62.09	69.83	59.44	60.43	63.33
	SWAP-RoBERTa(ours)	**61.45**	**71.25**	**58.92**	**64.11**	**71.97**	**61.62**	**62.34**	**65.27**
Supervised	DPR-RoBERTa	62.07	72.94	60.02	64.91	70.04	63.06	63.50	63.99
	DPR-SWAP-RoBERTa(ours)	**64.07**	**75.67**	**64.05**	**67.92**	**73.44**	**66.82**	**66.22**	**67.64**

Table 3: The performances of different methods on LERD. Baseline marked with * is initialized with RoBERTa and trained on our unsupervised corpus, and those marked with $^{\circ}$ are the multilingual version and 'MS' means pretrained on MS MARCO.

the training data and set the maximum input length of facts and evidence to 128 tokens. We treat fact as query and evidence as candidate. We train SWAP on 1 $\times$ Tesla-A100 80G GPU with a batch size of 8 and optimize the model with AdamW with a learning rate of 1e-5, 10% steps for warm-up and 5 epochs. The temperature hyper-parameter τ is 0.1.

Evaluation Sentence-level evidence is retrieved and ranked for each fact by the cosine similarity score between their dense representation. We employ Mean Average Precision (MAP), Mean Reciprocal Rank (MRR), top-k Recall (R@k), and Normalized Discounted Cumulative Gain (NDCG@k) as the evaluation metrics and report the overall test results averaged over the fact queries.

Supervised Setting We utilize dense retrieval model DPR (Karpukhin et al. 2020) as the supervised baseline and train the model on our dataset with the released code and initialize the encoders with RoBERTa. The batch size is set to 32, we regard irrelevant evidence of a given fact in the same case as hard negative and use default settings for other options.

5.2 Overall Performance

In the unsupervised setting, we compare SWAP with three types of baselines. For sparse retrieval, which is based on word co-occurrence, we choose BM25 (Robertson and Zaragoza 2009) and Legal-Event-IR (Yao et al. 2022) for comparison. Regarding dense retrieval, Contriever (Izacard et al. 2021a) and Condenser (Gao and Callan 2021) for both unsupervised and transfer settings are adopted as baselines. We also consider text representation methods including pretrained language models, such as BERT (Devlin et al. 2019), Roberta (Cui et al. 2020) and LawFormer (Xiao et al. 2021) using average pooling, along with sentence embedding methods including SBERT (Reimers and Gurevych 2019) and SimCSE (Gao, Yao, and Chen 2021).

The overall performances are shown in Table 3. In the unsupervised setting, SWAP substantially outperforms both sparse and dense retrieval baselines. All dense models yield better results than BM25 and Legal-Event, which can not deal with the vocabulary mismatch problem in LERD. The unsupervised SimCSE fine-tuned with LERD outperforms other baselines and even beats methods designed for information retrieval. Since the adopted dense retrieval baselines are mainly focused on modeling coarse-grained relevance at the document level, it is reasonable that they do not perform well on LER task, which requires a meticulous comparison between facts and evidence. SWAP models outperform other baselines by a large margin, indicating the proposed structure-aware contrastive learning framework is effective.

In the supervised setting, we train DPR on LERD with RoBERTa initialization. Further, we utilize the trained SWAP-RoBERTa as initialization and achieve a performance gain of 2 points on MAP, which indicates that pretraining with SWAP also works in the supervised scenario.

Method	MAP	MRR	R@5	NDCG@5
SWAP	61.45	71.25	71.97	65.27
$SWAP_{wo\text{-}DE}$	60.72	70.57	70.56	64.09
−AP	51.01	61.98	59.39	53.92
−DP	52.90	62.89	63.16	56.03
−DP-in-case	54.81	64.72	65.37	58.15
$SWAP^{cls}$	57.10	67.39	67.00	60.78
$SWAP^{cls}_{wo\text{-}DE}$	56.77	66.94	66.02	59.97
−AP	45.17	55.69	54.13	47.79
−DP	48.18	58.05	57.89	51.13
−DP-in-case	51.01	60.93	60.84	54.02

Table 4: Comparison of different training strategies. $SWAP_{wo\text{-}DE}$: without denoising, *−AP*: without approximate positive, *−DP*: without dropout positive, *−DP-in-case*: without dropout in-case negatives, SWAP^{cls}: with [cls] pooling strategy.

5.3 Ablation Study

We verify the effectiveness of the different parts in SWAP by removing each of them independently, and the results are shown in Table 4. We find that both dropout positive and approximate positive are indispensable. Since facts in a case

Backbone	Model	MAP	MRR	R@5	NDCG@5
BERT-tiny	*Vanilla*	45.82	55.77	54.99	48.15
	$SWAP_{wo\text{-}DE}$	51.48	61.39	60.99	54.17
	SWAP	52.17	62.38	61.85	55.00
BERT	*Vanilla*	47.97	58.15	57.25	50.58
	$SWAP_{wo\text{-}DE}$	57.42	67.93	67.67	61.14
	SWAP	59.65	69.58	69.83	63.33
RoBERTa	*Vanilla*	51.63	61.62	61.44	54.58
	$SWAP_{wo\text{-}DE}$	60.72	70.57	70.56	64.09
	SWAP	61.45	71.25	71.97	65.27
MENGZI	*Vanilla*	48.91	59.25	57.90	51.47
	$SWAP_{wo\text{-}DE}$	61.17	70.98	71.78	64.94
	SWAP	61.31	71.32	71.16	65.06
ERNIE	*Vanilla*	47.03	56.89	55.98	49.46
	$SWAP_{wo\text{-}DE}$	59.29	69.68	68.61	62.77
	SWAP	60.83	70.34	71.34	64.71

Table 5: Performance of applying our training strategy on different backbone models. '*Vanilla*' denotes directly using the backbone model with avg-pooling.

can be highly similar, adding the in-case negatives is another key factor to enable the model to differentiate between similar facts. The denoising strategy also leads to a gain on all metrics, indicating that approximate positives are noisy and our entropy-guided denoising strategy is effective. We also conduct ablation on SWAP with cls pooling strategy and the results indicate SWAP is pooling-independent and robust.

5.4 Effect of Backbones

We conduct experiments on different backbones to verify the generalization of SWAP, results are shown in Table 5.

Among those backbones, the parameter size of Bert-tiny (Turc et al. 2019) is 7% of the others, Mengzi (Zhang et al. 2021) utilizes a lightweight training strategy and Ernie (Sun et al. 2019) is a knowledge-enhanced language model. From those results, we could conclude that the proposed method is constantly effective on different backbones with various sizes and training objectives.

5.5 Effect of Training Samples

Scale of Training Data To validate the influence of the training data size, we train SWAP with 1K, 3K randomly sampled cases, and test the performance on the whole test set. The results in Table 6 illustrate that training with only 1K data achieves a comparable result and scaling up the training data can steadily promote the performance and adding more data brings a higher performance gain, which exhibits that SWAP is an effective method of leveraging the unsupervised data in the legal domain.

Train	MAP	MRR	R@5	NDCG@5
1K	59.17	69.18	69.21	62.48
3K	60.31	70.12	70.54	63.90
All	61.45	71.25	71.97	65.27

Table 6: Test results of training with data of different scales.

Train	MAP	MRR	R@5	NDCG@5
Drug	55.06	65.81	64.87	58.22
Steal	59.21	69.45	69.36	62.61
Bodily-harm	58.42	68.85	68.92	62.15

Table 7: Test results of training with data on different crimes.

Type of Training Data There are over 400 crimes in the criminal law of China and the facts involved vary a lot. To test the generalization ability of SWAP, we train SWAP on each different crime with 2000 training cases and test them on the whole test set. As shown in Table 7, training with Drug data achieves worse performance, because the facts in the drug cases are relatively fixed while those in the other two crimes involve more kinds of actions. The overall performances of training with a single crime are within a satisfying range, indicating that SWAP generalizes to all crimes.

6 Related Works

Despite the success of NLP techniques for legal applications in recent years, only a few works focus on the crucial step of fact retrieval. Tomlinson et al. (2007) proposed to retrieve business records in legal databases, but they only consider tobacco-related documents. Teng and Chao (2021) introduce the task of evidence association to clustering evidence, however, their method only operates on document titles and could not align facts with evidence. Different from legal case retrieval (Ma et al. 2021; Shao et al. 2020) that aims to acquire similar cases with fact, the proposed legal evidence retrieval task requires finer-grained text representation and the ability to handle expression mismatch. Building positive samples is the vital step toward unsupervised dense retrieval. Previous works(Lee, Chang, and Toutanova 2019; Izacard et al. 2021b; Ram et al. 2022) typically leverage a sub-sequence sampling strategy that randomly selects a span from the initial document as the query and treats the rest part (all of them or another random span) as the positive sample. While these strategies work well for open-domain information retrieval and question-answering tasks, they are designed to learn coarse-grained text correlation, which is inherently different from the fine-grained matching problem of our task. As far as we know, we are the first to propose the legal evidence retrieval task and tackle the positive sample generation problem through approximate aggregation.

7 Conclusion

In this paper, we propose the task of Legal Evidence Retrieval (LER) to build real-world Legal AI applications that can help judges efficiently find relevant oral evidence for a given fact. A large-scale dataset is constructed for the design and evaluation of LER algorithms, including well-annotated cases and a partially aligned corpus. We introduce **S**tructure-a**W**are contrastive learning with **A**pproximate aggregated **P**ositive (**SWAP**), which involves a novel strategy of approximating positives along with an effective technique for denoising the false positive samples. We use the SWAP framework to train dense retrieval models in an unsupervised manner, achieving state-of-the-art performance on LERD.

Ethical Statement

The task of LER is aimed at helping the judges quickly find the relevant evidence to review and check the prosecuted facts before the trial instead of helping the judges make decisions. And the facts will be further checked with the defendant, victim, and witness during the trial. All source files of our dataset are from the official legal document website which is publicly available. All techniques we introduced in this paper are only designed to serve as an auxiliary tool in the finding of fact process and do not play any decisive role. We do not analyze the content of the case or the litigants in any way other than evidence retrieval.

Acknowledgments

We give our sincere thanks to the data annotation team led by Qingpeng Yang from Alibaba DAMO Group for their effort in offering high-quality data. We are also grateful to Danyang Guo from School of Law, Tsinghua University for the professional legal help. This work is supported by the National Key Research and Development Program of China (No. 2022YFC3301504, 2020YFC0832505).

References

Chen, Y.; Sun, Y.; Yang, Z.; and Lin, H. 2020. Joint Entity and Relation Extraction for Legal Documents with Legal Feature Enhancement. In *Proceedings of the 28th International Conference on Computational Linguistics*, 1561–1571. Barcelona, Spain (Online): International Committee on Computational Linguistics.

Cui, Y.; Che, W.; Liu, T.; Qin, B.; Wang, S.; and Hu, G. 2020. Revisiting Pre-Trained Models for Chinese Natural Language Processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, 657–668. Online: Association for Computational Linguistics.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics.

Gao, L.; and Callan, J. 2021. Condenser: a Pre-training Architecture for Dense Retrieval. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 981–993. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Gao, T.; Yao, X.; and Chen, D. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6894–6910. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2021a. Towards Unsupervised Dense Information Retrieval with Contrastive Learning. *CoRR*, abs/2112.09118.

Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2021b. Unsupervised Dense Information Retrieval with Contrastive Learning. *arXiv preprint arXiv:2112.09118*.

Karpukhin, V.; Oguz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W.-t. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6769–6781. Online: Association for Computational Linguistics.

Lee, K.; Chang, M.-W.; and Toutanova, K. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6086–6096.

Ma, Y.; Ai, Q.; Wu, Y.; Shao, Y.; Liu, Y.; Zhang, M.; and Ma, S. 2022. Incorporating Retrieval Information into the Truncation of Ranking Lists for Better Legal Search. In Amigó, E.; Castells, P.; Gonzalo, J.; Carterette, B.; Culpepper, J. S.; and Kazai, G., eds., *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, 438–448. ACM.

Ma, Y.; Shao, Y.; Wu, Y.; Liu, Y.; Zhang, R.; Zhang, M.; and Ma, S. 2021. LeCaRD: A Legal Case Retrieval Dataset for Chinese Law System. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 2342–2348. ACM.

Ram, O.; Shachaf, G.; Levy, O.; Berant, J.; and Globerson, A. 2022. Learning to Retrieve Passages without Supervision. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2687–2700. Seattle, United States: Association for Computational Linguistics.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 3982–3992. Hong Kong, China: Association for Computational Linguistics.

Robertson, S. E.; and Zaragoza, H. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4): 333–389.

Sciavolino, C.; Zhong, Z.; Lee, J.; and Chen, D. 2021. Simple Entity-Centric Questions Challenge Dense Retrievers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 6138–6148. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Shao, Y.; Mao, J.; Liu, Y.; Ma, W.; Satoh, K.; Zhang, M.; and Ma, S. 2020. BERT-PLI: Modeling Paragraph-Level Interactions for Legal Case Retrieval. In Bessiere, C., ed., *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, 3501–3507. International Joint Conferences on Artificial Intelligence Organization. Main track.

Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1): 1929–1958.

Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Chen, X.; Zhang, H.; Tian, X.; Zhu, D.; Tian, H.; and Wu, H. 2019. ERNIE: Enhanced Representation through Knowledge Integration. *CoRR*, abs/1904.09223.

Teng, Y.; and Chao, W. 2021. Argumentation-Driven Evidence Association in Criminal Cases. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, 2997–3001. Punta Cana, Dominican Republic: Association for Computational Linguistics.

Tomlinson, S.; Oard, D. W.; Baron, J. R.; and Thompson, P. 2007. Overview of the TREC 2007 Legal Track. In *TREC*.

Turc, I.; Chang, M.; Lee, K.; and Toutanova, K. 2019. Well-Read Students Learn Better: The Impact of Student Initialization on Knowledge Distillation. *CoRR*, abs/1908.08962.

Xiao, C.; Hu, X.; Liu, Z.; Tu, C.; and Sun, M. 2021. Lawformer: A pre-trained language model for Chinese legal long documents. *AI Open*, 2: 79–84.

Yao, F.; Xiao, C.; Wang, X.; Liu, Z.; Hou, L.; Tu, C.; Li, J.; Liu, Y.; Shen, W.; and Sun, M. 2022. LEVEN: A Large-Scale Chinese Legal Event Detection Dataset. In *Findings of the Association for Computational Linguistics: ACL 2022*, 183–201. Dublin, Ireland: Association for Computational Linguistics.

Zhang, S.; Liang, Y.; Gong, M.; Jiang, D.; and Duan, N. 2022. Multi-View Document Representation Learning for Open-Domain Dense Retrieval. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 5990–6000. Dublin, Ireland: Association for Computational Linguistics.

Zhang, Z.; Zhang, H.; Chen, K.; Guo, Y.; Hua, J.; Wang, Y.; and Zhou, M. 2021. Mengzi: Towards Lightweight yet Ingenious Pre-trained Models for Chinese. *CoRR*, abs/2110.06696.

Zhong, H.; Guo, Z.; Tu, C.; Xiao, C.; Liu, Z.; and Sun, M. 2018. Legal Judgment Prediction via Topological Learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 3540–3549. Brussels, Belgium: Association for Computational Linguistics.