Label Drop for Multi-Aspect Relation Modeling in Universal Information Extraction

Lu Yang[1], **Jiajia Li**[2,3], **En Ci**[1], **Lefei Zhang**[1], **Zuchao Li**[1,*], **Ping Wang**[2,3]
[1]School of Computer Science, Wuhan University, Wuhan, China
[2]School of Information Management, Wuhan University, Wuhan, China
[3]Key Laboratory of Archival Intelligent Development and Service, NAAC
{yang_lu, cantata, cien.cs}@whu.edu.cn
{zhanglefei, zcli-charlie, wangping}@whu.edu.cn

Abstract

Universal Information Extraction (UIE) has garnered significant attention due to its ability to address model explosion problems effectively. Extractive UIE can achieve strong performance using a relatively small model, making it widely adopted. Extractive UIEs generally rely on task instructions for different tasks, including single-target instructions and multiple-target instructions. Single-target instruction UIE enables the extraction of only one type of relation at a time, limiting its ability to model correlations between relations and thus restricting its capability to extract complex relations. While multiple-target instruction UIE allows for the extraction of multiple relations simultaneously, the inclusion of irrelevant relations introduces decision complexity and impacts extraction accuracy. Therefore, for multi-relation extraction, we propose LDNet, which incorporates multi-aspect relation modeling and a label drop mechanism. By assigning different relations to different levels for understanding and decision-making, we reduce decision confusion. Additionally, the label drop mechanism effectively mitigates the impact of irrelevant relations. Experiments show that LDNet outperforms or achieves competitive performance with state-of-the-art systems on 9 tasks, 33 datasets, in both single-modal and multimodal, few-shot and zero-shot settings.[1]

1 Introduction

Information Extraction (IE) (Andersen et al., 1992; Grishman, 2019) tasks, both single-modal and multi-modal, encompass a wide variety of domains and relations between entities, leading to a highly diversified landscape. However, this diversification poses a significant challenge known as model explosion, which refers to the proliferation of models required to handle the diverse structures and relations present in different IE tasks. Traditionally, task-specific models (Zhang et al., 2018a; Wang and Lu, 2020; Zhong and Chen, 2021; Zhang et al., 2022; Peng et al., 2023a; Tian et al., 2023; Li et al., 2024a) have been developed to address the unique requirements of individual tasks. However, this approach is not scalable and becomes increasingly impractical as the number of tasks and their complexity grow.

To tackle the issue of model explosion, Universal Information Extraction (UIE) (Lu et al., 2022; Li et al., 2024b) has emerged as a promising paradigm. UIE aims to develop models that can extract information across different domains and relations, in both single-modal (Fei et al., 2022) and multimodal (Zheng et al., 2023) setting, without relying on task-specific models for each individual task. By leveraging shared knowledge, UIE models can generalize well to various IE tasks, reducing the need for a multitude of specialized models.

Generative UIE (Wang et al., 2022a; Sainz et al., 2024) approaches have been explored, but their reliance on large generative models as the foundation limits their efficiency. These models suffer from computational complexity and resource requirements, hindering their practical applicability. In contrast, extractive UIE (Ping et al., 2023) approaches have gained popularity due to their ability to achieve strong performance using relatively small models.

Single-target instruction UIE (Wadden et al., 2019) allows for the extraction of one type of relation at a time. While it excels in accuracy for simple relations, its limited efficiency and inability to model correlations between relations restrict its applicability to more complex IE tasks. To address these limitations, multiple-target instruction UIE (Zhu et al., 2023) has been proposed, enabling

[*] Corresponding author. This work was supported by the National Natural Science Foundation of China (No. 62306216, No. 72074171, No. 72374161), the Technology Innovation Program of Hubei Province (Grant No. 2024BAB043), the Natural Science Foundation of Hubei Province of China (No. 2023AFB816).

[1]https://github.com/Lu-Yang666/LDNet

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 5021–5040
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

the extraction of multiple relations simultaneously. This approach aims to model correlations between relations and improve extraction efficiency. However, the incorporation of irrelevant relations introduces decision complexity and can have ramifications on extraction accuracy.

Hence, to address the challenges in multi-relation extraction, we propose LDNet, a novel approach that leverages multi-aspect relation modeling and a label drop mechanism. In LDNet, we assign different relations to different levels for understanding and decision-making. This approach allows the model to capture the unique characteristics and nuances of each relation separately. By organizing relations into distinct levels, LDNet reduces decision confusion, enabling more accurate and reliable extraction results. Additionally, LDNet incorporates a label drop mechanism to address the impact of irrelevant relations. During the extraction process, LDNet selectively drops irrelevant labels, focusing on the most relevant relations for extraction. This mechanism helps mitigate the interference caused by irrelevant relations, ensuring that the model can concentrate its attention and resources on extracting the necessary and meaningful information. By filtering out noise and irrelevant signals, LDNet enhances the overall extraction performance and reduces the potential for false positives.

To assess the effectiveness of LDNet, we conduct extensive experiments on a diverse range of IE tasks and benchmark datasets, both single-modal and multi-modal. The evaluation covers few-shot and zero-shot settings to examine the generalization capability of LDNet. The results demonstrate that LDNet outperforms or achieves competitive performance compared to previous state-of-the-art systems across 9 tasks and 33 datasets.

Our Contribution 1) We propose LDNet, a novel approach that leverages multi-aspect relation modeling and a label drop mechanism. 2) We employ model transfer learning, a valuable strategy for further enhancing model performance across various datasets. 3) We conduct experiments on 33 datasets across 9 tasks, in both single-modal and multi-modal, few-shot and zero-shot settings, and the results demonstrate the superiority of LDNet.

2 Related Work

Generative UIE TANL (Paolini et al., 2021) sees IE tasks as a sequence-to-sequence problem and utilizes T5 as the generative model. UIE (Lu et al., 2022) also uses T5 as the backbone. In addition, UIE designs Structured Extraction Language (SEL) that can represent diversified IE tasks, thereby enabling it to perform on a wider range of IE tasks. InstructUIE (Wang et al., 2023) further incorporates the idea of instruction-tuning and utilizes FlanT5-11B (Chung et al., 2022) for IE tasks. DeepStruct (Wang et al., 2022a) and GenIE (Josifoski et al., 2022) both formulate the generated sequence as subject-relation-object triplets, with DeepStruct having a larger model size (10B). LasUIE (Fei et al., 2022) proposes a novel structure-aware generative language model to unleash the power of syntactic knowledge. FSUIE (Peng et al., 2023b) introduces fuzzy span loss and fuzzy span attention to reduce over-reliance on span boundaries. GOLLIE (Sainz et al., 2024) improves zero-shot results on unseen IE tasks by virtue of being fine-tuned to comply with annotation guidelines. TMR (Zheng et al., 2023) addresses text-image misalignment by introducing a back-translation method using diffusion-based generative models. KnowCoder (Li et al., 2024b) introduces a code-style schema representation method. While the above generative UIE approaches offer a powerful solution for diversified IE tasks, they do not possess any notable advantages when it comes to efficiency.

Extractive UIE DyGIE++ (Wadden et al., 2019) utilizes a dynamic span graph to model long-range relations, and with graph propagation, the model can disambiguate challenging entity mentions. UniEX (Ping et al., 2023) converts IE tasks into a token-pair problem, develops a traffine attention mechanism to integrate heterogeneous factors, and obtains the extraction target via a scoring matrix. These single-target extractive UIE approaches can achieve strong performance using a relatively small model; however, they lack the ability to model correlations between relations, thus limiting their capability to extract complex relations.

OneIE (Lin et al., 2020) also uses a span graph, but unlike DyGIE++, it incorporates global features and adopts a CRF-based tagger to remove the constraint on the length of extracted mentions. UMGF (Zhang et al., 2021) adopts a unified intra-modal and inter-modal graph fusion method to represent visual and textual features within the same embedding space. HVPNeT (Chen et al., 2022) designs pyramidal features for images, employing

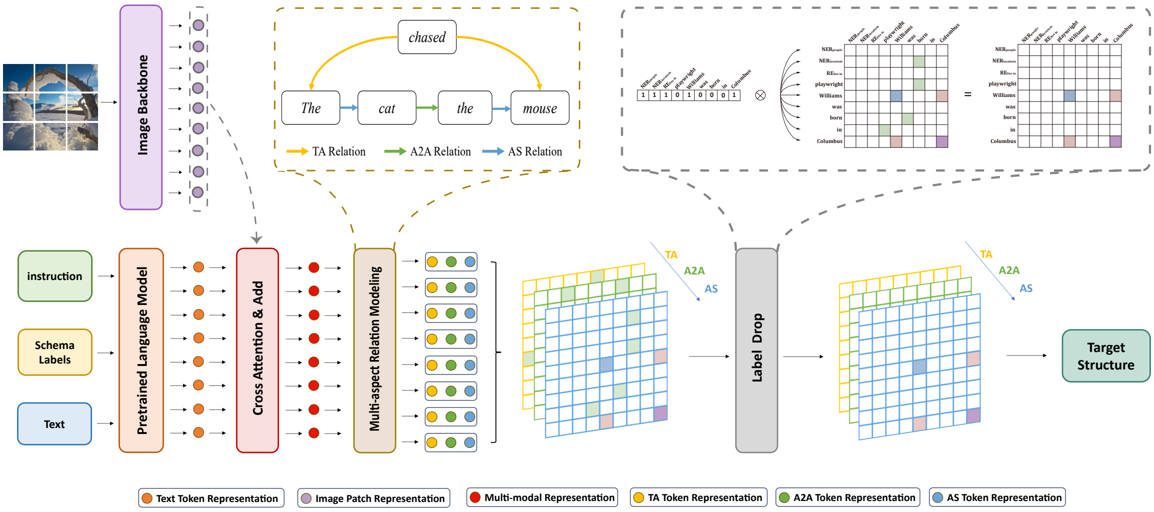

Figure 1: The overview framework of LDNet. LDNet constructs a unified input format, which combines instruction, schema labels, and text. The representation obtained from the PLM is fused with image representation obtained with the image backbone. The multi-modal representation is fed into the multi-aspect relation modeling component to produce probability matrices for TA, A2A, and AS relations, respectively. These matrices are then subjected to label drop to mask out non-existent relations. Finally, the probability matrices are fed into the decoding process to generate target structures.

visual representations as insertable visual prefixes to guide error-insensitive predictive decisions of textual representations. MetaRetriever (Yu et al., 2023) retrieves task-specific knowledge from pretrained language models to enhance performance. Mirror (Zhu et al., 2023) transforms multiple tasks into a multi-span cyclic graph and predicts relations by verifying whether a cycle exists between slots in a tuple. While these multiple-target extractive UIE approaches take interactions between relations into account, they also include irrelevant relations, which leads to decision complexity and inaccuracy.

It is worth noting that the existing UIE models have basically only conducted experiments on single-modal or multi-modal IE tasks, and have not handled single-modal and multi-modal IE tasks simultaneously like LDNet.

3 Methodology

LDNet's overall framework is built upon a pretrained language model and an image backbone, consisting of a multi-aspect relation modeling component and a label drop mechanism, as shown in Figure 1.

We formulate IE tasks as a multi-aspect span-based relation extraction problem. Specifically, we consider three kinds of relations among IE tasks: TA relation (trigger-to-argument relation), A2A relation (argument-to-argument relation), and AS relation (argument-span relation). The TA relation signifies the association of the trigger word with the identified span. The A2A relation describes the connection between two related spans, representing the semantic or contextual relation between the identified spans. The AS relation describes the connection within a span, enabling LDNet to analyze the internal structure and coherence within the span itself. As shown in Figure 1, the AS relation is formed between "the" and "cat", the trigger word "chased" connects to "the" in the span "the cat" and "mouse" in the span "the mouse" through the TA relation, and "cat" is linked to "the" in the span "the mouse" through the A2A relation.

3.1 Multi-aspect Relation Modeling

LDNet regularizes text input format into three components: instruction, schema labels, and text. Given an input sequence $\mathbf{x} = [x_1, x_2, \ldots, x_{|x|}]$, LDNet computes the text representation $\mathbf{H} = [h_1, h_2, \ldots, h_{|x|}] \in \mathbb{R}^{|x| \times d_h}$ as follows:

$$\mathbf{H} = PLM([x_1, x_2, \ldots, x_{|x|}]) \quad (1)$$

$$= [h_1, h_2, \ldots, h_{|x|}] \quad (2)$$

where $PLM(\cdot)$ is a pretrained language model.

To inject image information, given an image I, LDNet initially resizes it to 224 × 224, then divides it into n_p patches according to the patch size specified by the image backbone, and subsequently derives image feature representation $\mathcal{V} \in \mathbb{R}^{n_p \times d_v}$

using the image backbone:

$$\mathcal{V} = VisionTransformer(I) \tag{3}$$

LDNet then employs cross-attention, where the image feature representation functions as the query and the text representation serves as both the key and value. The output of the attention mechanism is then subjected to the hyperbolic tangent activation function, followed by a summation operation. Finally, LDNet changes the sequence length of the resulting image feature representation, yielding the final image representation $\mathbf{V} \in \mathbb{R}^{|x| \times d_h}$:

$$Q = \mathrm{FFNN}_{\mathrm{q}}^{\mathrm{I}}(MLP(\mathcal{V})),\ Q \in \mathbb{R}^{n_p \times d_h} \tag{4}$$

$$K = \mathrm{FFNN}_{\mathrm{k}}^{\mathrm{I}}(\mathbf{H}),\ V = \mathrm{FFNN}_{\mathrm{v}}^{\mathrm{I}}(\mathbf{H}) \tag{5}$$

$$\mathcal{V}' = \sum_{i=1}^{n_p} Tanh\left(Softmax\left(\frac{QK^{\mathrm{T}}}{\sqrt{d_h}}\right)V\right) \tag{6}$$

$$\mathbf{V} = Tanh(\mathrm{FFNN}_{\mathrm{I}}(\mathcal{V}')) \tag{7}$$

where $MLP(\cdot)$ represents a three-layer multilayer perceptron, $\mathrm{FFNN}_{\mathrm{q/k/v}}^{\mathrm{I}} \in \mathbb{R}^{d_h \times d_h}$ represents feed-forward network for generating query/key/value, and $\mathrm{FFNN}_{\mathrm{I}} \in \mathbb{R}^{1 \times |x|}$ represents the feed-forward network changing the sequence length of the resulting image feature representation.

LDNet modulates the fusion of image representation and text representation via a hyperparameter $\alpha \in [0, 1]$, and the fused image-text representation $\mathbf{M}$ is expressed as:

$$\mathbf{M} = \mathbf{H} + \alpha \cdot \mathbf{V} \tag{8}$$

α is set to 0 when only doing single-modal IE tasks.

After obtaining the multi-modal representation $\mathbf{M} = [m_1, m_2, \ldots, m_{|x|}]$, LDNet utilizes Rotary Position Embedding (RoPE) (Su et al., 2022) to achieve relative position encoding via combining the attention computation with absolute position encoding. The queries and keys for different relations are calculated as follows:

$$q_i^r = \mathrm{FFNN}_q^r(m_i),\ k_j^r = \mathrm{FFNN}_k^r(m_j) \tag{9}$$

where $r \in \{TA, A2A, AS\}$, $\mathrm{FFNN}_{q/k}^r \in \mathbb{R}^{d_h \times d_i}$ are feed-forward layers for different relations, and q_i^r and k_j^r are the $i\text{-}th$ query and the $j\text{-}th$ key for different relations.

Afterwards, q_i^r and k_j^r are each left-multiplied by the transformation matrices R_i and R_j used in RoPE respectively. The dot product of R_i and R_j satisfies $R_i^T R_j = R_{j-i}$, thus incorporating relative position information. The probability s_{ij}^r of the relation r existing between the span from i to j is the scaled dot product of the transformed q_i^r and k_j^r:

$$s_{ij}^r = \frac{(R_i q_i^r)^T (R_j k_j^r)}{\sqrt{d_i}} = \frac{q_i^{rT} R_{j-i} k_j^r}{\sqrt{d_i}} \tag{10}$$

By parallelly computing the scaled dot product over all token pairs of different relations separately, we can obtain three probability matrices:

$$S^r = \begin{bmatrix} s_{11}^r & s_{12}^r & \cdots & s_{1|x|}^r \\ s_{21}^r & s_{22}^r & \cdots & s_{2|x|}^r \\ \vdots & \vdots & \ddots & \vdots \\ s_{|x|1}^r & s_{|x|2}^r & \cdots & s_{|x||x|}^r \end{bmatrix} \tag{11}$$

where s_{ij}^r is the probability of the specific relation $r \in \{TA, A2A, AS\}$ existing between the token pair $\langle x_i, x_j \rangle$.

During training, LDNet utilizes multi-label categorical cross-entropy loss as the loss function for multi-aspect relation modeling:

$$l_{MR,neg}^r = log\left(1 + \sum_{\Omega_{neg}} e^{s_{ij}^r}\right) \tag{12}$$

$$l_{MR,pos}^r = log\left(1 + \sum_{\Omega_{pos}} e^{-s_{ij}^r}\right) \tag{13}$$

$$L_{MR} = \sum_{r \in \{TA, A2A, AS\}} (l_{MR,neg}^r + l_{MR,pos}^r) \tag{14}$$

where Ω_{neg} and Ω_{pos} are the sets of negative and positive samples, respectively. Graph labels $G^r \in \mathbb{R}^{|x| \times |x|}$ are used to distinguish between negative and positive samples. Negative samples consist of position pairs where $G_{ij}^r = 0$, while positive samples are pairs where $G_{ij}^r = 1$. G_{ij}^r represents the label of the token pair $\langle x_i, x_j \rangle$ for relation r.

3.2 Label Drop

To prioritize the relational token pairs, we employ label drop to filter out token pairs that are unlikely to have relations.

Specifically, we first design a label vector $\mathbf{l}^r = [l_1^r, l_2^r, \ldots, l_{|x|}^r] \in \mathbb{R}^{1 \times |x|}$ for each relation as the standard. We set the values of the elements in $\mathbf{l}^r$

whose indices fall within the gold spans and their corresponding schema labels to 1, and the rest to 0.

LDNet transforms the representation $\mathbf{M}$ into the predicted matrix $\hat{L}^r \in \mathbb{R}^{1\times|x|\times 1}$ via linear activation and Sigmoid function:

$$\hat{L}^r = Sigmoid\left(\mathrm{FFNN}^r\left(\mathbf{M}\right)\right) \tag{15}$$

where $\mathrm{FFNN}^r \in \mathbb{R}^{d_h \times 1}$ is the feed-forward network for different relations.

Later, LDNet squeezes the matrix $\hat{L}^r$ into $\hat{l}^r \in \mathbb{R}^{1\times|x|}$ and multiplies $\hat{l}^r$ with every row vector in the corresponding probability matrix S^r computed in Section 3.1, respectively, to obtain the final probability matrix P^r, which is later used for decoding:

$$p^r_{i\cdot} = \hat{l}^r \otimes s^r_{i\cdot} \tag{16}$$

where $p^r_{i\cdot}$ represents the i-th row vector of P^r, $s^r_{i\cdot}$ represents the i-th row vector of S^r, and $\otimes$ represents dot product. The detailed logic of label drop mechanism can be seen in Appendix A.4.

LDNet calculates the binary cross-entropy loss between $\hat{l}^r$ and $\mathbf{l}^r$ to make the predicted vector $\hat{l}^r$ approach the label vector $\mathbf{l}^r$ of the same relation during training:

$$l^r_{LD} = -\frac{1}{n}\sum_{i}^{|x|}\left(\mathbf{l}^r_i \log(\hat{l}^r_i) + (1-\mathbf{l}^r_i)\log(1-\hat{l}^r_i)\right) \tag{17}$$

$$L_{LD} = \sum_{r\in\{TA,A2A,AS\}} l^r_{LD} \tag{18}$$

After label drop, LDNet utilizes the three final probability matrices for relation decoding. During the process of relation decoding, LDNet extracts relation between token pair whose final probability is larger than the threshold of 0.5 and identifies potential relation structures. If LDNet detects a closed relation loop as shown in Figure 1, it adds the extracted span to the predicted answer.

3.3 Model Transfer Learning

To further boost LDNet's performance, we propose a model transfer learning approach. We select the best-performance models fine-tuned on each dataset as the teacher models, and the generated probability distributions, namely P^r of all data entries, from these teacher models are used as soft labels when fine-tuning the corresponding pre-trained student models. Mean Squared Error (MSE) loss is employed to guide LDNet in reducing the discrepancy between the distributions of the student model and the teacher model:

$$L_{MT} = \sum_{r\in\{TA,A2A,AS\}} \frac{1}{|x|^2}\sum_{i=1}^{|x|}\sum_{j=1}^{|x|}\left(p^r_{ij} - \hat{p}^r_{ij}\right)^2 \tag{19}$$

where p^r_{ij} represents the ij-th element of the P^r generated by the teacher model, and $\hat{p}^r_{ij}$ represents the ij-th element of the $\hat{P}^r$ generated by the student model. The detailed algorithm is in Algorithm 1. Thus, the complete objective for LDNet model training can be represented as follows:

$$L = L_{MR} + L_{LD} + L_{MT} \tag{20}$$

Algorithm 1 Model Transfer Learning

Input: Teacher model distributions $\mathcal{D}_\mathcal{T}$ and pre-trained student model parameters θ
Output: Fine-tuned student model parameters Θ

```
for $i$ in range (0, epochs) do
    (iterate over fine-tuning epochs)
    for $j$ in range (0, steps) do
        Obtain student model distributions $\mathcal{D}_\mathcal{S}$.
        Set loss $L \leftarrow L_{MR} + L_{LD}$.
        for $d_s \in \mathcal{D}_\mathcal{S}$ do
            if $d_s$ finds the corresponding $d_t \in \mathcal{D}_\mathcal{T}$ then
                $L \leftarrow L + MSE(d_s, d_t)$.
            end if
        end for
        $\Theta = \theta - \gamma\frac{\hat{m}_t}{\sqrt{\hat{v}_t}+\epsilon}$ (use the AdamW optimizer to update parameters)
    end for
end for
return $\Theta$
```

4 Experiments

4.1 Experiment Setup

We use DeBERTa-v3-large (He et al., 2021) as the PLM, ViT (Dosovitskiy et al., 2021) as the image backbone, and AdamW (Loshchilov and Hutter, 2019) as the optimizer. We conduct experiments on ACE04 (Mitchell et al., 2005), ACE05 (Walker et al., 2006), CoNLL03 (Tjong Kim Sang and De Meulder, 2003), CoNLL04 (Roth and Yih, 2004), NYT (Riedel et al., 2010), SciERC (Luan et al., 2018), CASIE (Satyapanich et al., 2020), 14-res and 14-lap (Pontiki et al., 2014), 15-res (Pontiki et al., 2015), 16-res (Pontiki et al., 2016), Twitter2015 (Lu et al., 2018), Twitter2017 (Zhang et al.,

Task	Datasets	TANL	UIE	DeepStruct	InstructUIE	USM	Mirror	FSUIE	UniEX	MetaRetriever	GoLLIE	**LDNet**
NER	ACE04	-	86.89	-	-	87.62	87.66	86.16	87.12	86.10	-	**88.69**
	ACE05	84.90	85.78	86.90	86.66	87.14	86.72	86.91	87.02	84.01	**88.10**	87.79
	CoNLL03	91.70	92.99	93.00	92.94	93.16	92.97	-	92.65	92.38	92.80	**93.44**
RE	ACE05	63.70	66.06	66.80	-	67.88	69.02	**74.16**	66.06	64.37	63.60	69.14
	CoNLL04	71.40	75.00	78.30	78.48	78.84	75.22	-	73.40	73.66	-	**80.79**
	NYT	-	93.54	93.30	90.47	94.07	94.25	-	-	-	-	**94.96**
	SciERC	-	36.53	-	45.15	37.36	40.50	-	38.00	35.77	-	**46.85**
EE	ACE05-Tgg	68.40	73.36	69.80	77.13	72.41	74.44	-	74.08	72.38	72.20	**83.56**
	ACE05-Arg	47.60	54.79	56.20	**72.94**	55.83	57.87	-	53.92	52.62	66.00	61.14
	CASIE-Tgg	-	69.33	-	67.80	71.73	73.09	-	71.46	69.76	59.30	**73.36**
	CASIE-Arg	-	61.30	-	63.53	63.26	61.27	-	62.91	60.37	50.00	**63.88**
ABSA	14-res	-	74.52	-	-	77.26	76.05	74.17	74.77	73.41	-	**79.17**
	14-lap	-	63.88	-	-	65.51	64.08	65.56	65.23	62.83	-	**69.00**
	15-res	-	67.15	-	-	69.86	67.41	70.63	68.58	65.85	-	**71.18**
	16-res	-	75.07	-	-	78.25	77.46	75.80	76.02	73.55	-	**78.85**
MIE	Twitter-2015	-	-	-	-	-	73.08*	-	-	-	-	**76.17**
	Twitter-2017	-	-	-	-	-	83.91*	-	-	-	-	**87.57**
	MNRE	-	-	-	-	-	70.58*	-	-	-	-	**75.79**

Table 1: Results on 16 IE benchmarks. -Tgg and -Arg refer to trigger F1 score and argument F1 score, respectively. Mirror does not test on multi-modal IE datasets. The results marked with * are the performance we obtain using Mirror's model and training checkpoint.

Task	Datasets	Mirror w/ PT w/ Inst.	Mirror w/ PT w/o Inst.	Mirror w/o PT w/ Inst.	Mirror w/o PT w/o Inst.	LDNet$_{MT-}$ w/ PT w/ Inst.	LDNet$_{MT-}$ w/ PT w/o Inst.	LDNet$_{MT-}$ w/o PT w/ Inst.	LDNet$_{MT-}$ w/o PT w/o Inst.	LDNet w/ PT w/ Inst.	LDNet w/ PT w/o Inst.
NER	ACE04	87.16	86.39	87.66	87.26	85.68	85.63	88.21	86.76	86.94	**88.69**
	ACE05	85.34	85.70	86.72	86.45	84.70	85.87	87.11	85.38	87.70	**87.79**
	CoNLL03	92.73	91.93	92.11	92.97	92.23	92.67	93.30	92.70	**93.44**	92.81
RE	ACE05	67.86	67.86	64.88	69.02	66.35	67.73	69.10	68.58	69.02	**69.14**
	CoNLL04	75.22	72.96	71.19	73.58	76.90	78.26	75.81	75.03	**80.79**	80.37
	NYT	93.85	94.25	93.95	93.31	94.00	94.34	94.13	93.73	92.46	**94.96**
	SciERC	36.89	37.12	36.66	40.50	43.79	45.58	43.48	46.06	42.79	**46.85**
EE	ACE05-Tgg	74.44	73.05	72.66	73.38	73.99	75.81	72.92	73.98	72.34	**83.56**
	ACE05-Arg	55.88	54.73	56.51	57.87	56.27	57.88	50.70	58.01	54.80	**61.14**
	CASIE-Tgg	71.81	71.60	73.09	71.40	70.87	71.23	71.97	73.18	71.86	**73.36**
	CASIE-Arg	61.27	61.04	60.44	58.87	61.34	61.92	62.33	63.20	61.51	**63.88**
ABSA	14-res	75.06	74.24	76.05	75.89	73.93	74.76	76.16	77.93	76.40	**79.17**
	14-lap	64.08	62.48	59.56	60.42	66.60	66.03	65.32	63.80	**69.00**	65.42
	15-res	66.40	63.61	60.26	67.41	66.63	66.87	66.30	65.79	69.51	**71.18**
	16-res	74.24	75.40	73.13	77.46	74.41	76.10	77.97	77.19	76.34	**78.85**
Avg.		72.15	71.49	70.99	72.39	72.51	73.38	72.99	73.42	73.66	75.81

Table 2: Results of LDNet compared with Mirror. PT stands for pre-training, and Inst. represents the task instruction. LDNet$_{MT-}$ denotes LDNet without model transfer learning.

	P	R	F1
Discontinuous NER: CADEC			
BART-NER	70.08	71.21	70.64
W2NER	74.09	72.35	73.21
Mirror$_{\text{w/ PT \& Inst.}}$	74.83	65.45	69.83
Mirror$_{\text{w/o PT \& Inst.}}$	68.80	68.38	68.59
LDNet$_{\text{w/ PT \& Inst.}}$	**84.82**	71.16	**77.39**
LDNet$_{\text{w/o PT \& Inst.}}$	71.89	68.18	69.98
N-ary Tuples: HyperRED			
CubeRE	66.39	67.12	66.75
Mirror$_{\text{w/ PT \& Inst.}}$	71.29	62.46	66.58
Mirror$_{\text{w/o PT \& Inst.}}$	75.41	61.14	67.53
LDNet$_{\text{w/ PT \& Inst.}}$	69.39	66.56	**67.95**
LDNet$_{\text{w/o PT \& Inst.}}$	68.40	64.31	66.29

Table 3: Results on multi-span and n-ary information extraction tasks.

2018b), and MNRE (Zheng et al., 2021) datasets. We used the F1 score as the metric unless otherwise specified.

We compare LDNet with TANL (Paolini et al., 2021), DeepStruct (Wang et al., 2022a), UIE (Lu et al., 2022), InstructUIE (Wang et al., 2023), USM (Lou et al., 2023), Mirror (Zhu et al., 2023), FSUIE (Peng et al., 2023b), UniEX (Ping et al., 2023), MetaRetriever (Yu et al., 2023), and GoLLIE (Sainz et al., 2024) in single-modal IE tasks, and with Mirror in Multi-modal Information Extraction (MIE) tasks.

In different pre-training and fine-tuning strategies, we specifically compare LDNet with Mirror. The pre-training of LDNet is before fine-tuning on

Model	Parameter Scale	Movie	Restaurant	AI	Literature	Music	Politics	Science	Avg.
USM	372M	37.73	14.73	28.18	**56.00**	44.93	36.10	44.09	37.39
InstructUIE	11B	**63.00**	20.99	49.00	47.21	53.61	48.15	49.30	47.32
Mirror	304M	39.20	16.32	45.23	46.32	58.61	67.30	54.84	46.83
LDNet	304M	41.92	**22.91**	**49.02**	55.11	**61.10**	**69.03**	**59.83**	**51.27**
Llama-3	8B	7.48*	6.15*	7.40*	5.81*	3.41*	8.55*	4.43*	6.18*

Table 4: Zero-shot results on 7 NER datasets. The results for Llama-3 are obtained from our experiments and are for reference only.

Task	Model	1-shot	5-shot	10-shot	Avg.
NER CoNLL03	UIE	57.53	75.32	79.12	70.66
	USM	71.11	83.25	84.58	79.65
	Mirror	76.49	82.45	84.69	81.21
	LDNet	**78.33**	**84.53**	**85.39**	**82.75**
RE CoNLL04	UIE	34.88	51.64	58.98	48.50
	USM	36.17	53.20	60.99	50.12
	Mirror	26.29	47.42	55.77	43.16
	LDNet	**37.93**	**53.74**	**61.46**	**51.04**
Event Trigger ACE05	UIE	42.37	53.07	54.35	49.93
	USM	40.86	55.61	58.79	51.75
	Mirror	47.77	57.90	59.16	54.94
	LDNet	**54.77**	**62.75**	**64.24**	**60.59**
Event Arg ACE05	UIE	14.56	31.20	35.19	26.98
	USM	19.01	36.69	42.48	32.73
	Mirror	23.18	37.74	39.20	33.38
	LDNet	**25.42**	**39.12**	**43.04**	**35.86**
ABSA 16-res	UIE	23.04	42.67	53.28	39.66
	USM	30.81	52.06	58.29	47.05
	Mirror	36.21	51.65	58.59	48.82
	LDNet	**40.43**	**55.29**	**60.20**	**51.97**

Table 5: Few-shot results on 4 IE tasks. These datasets are not included in pre-training, and LDNet does not apply model transfer learning in this setting, thus avoiding the risk of information leakage.

downstream datasets, the pre-training datasets and hyperparameters are in Appendix C.1. 'With and without pre-training (PT)' refers to whether LDNet underwent pre-training on a pre-training dataset before fine-tuning on the downstream dataset. 'With and without task instructions (Inst.)' indicates whether the instruction part of LDNet's input is an empty string. In the configuration 'w/ PT w/o Inst.', the instruction part is an empty string, while in 'w/ PT w/ Inst.', the instruction part is not empty and resembles a string like 'Please determine the two entities mentioned in the text and specify the nature of their relationship.' The best results we obtained in the main results of Table 1 are from the four configurations: 'w/ PT w/ Inst.', 'w/ PT w/o Inst.', 'w/o PT w/ Inst.', and 'w/o PT w/o Inst.'

BART-NER (Lewis et al., 2020), W2NER (Li et al., 2022), and Mirror are chosen as the baseline models for the multi-span discontinuous NER task. In the hyper RE task, we select CubeRE (Chia et al., 2022) and Mirror as the baseline models. As for the MRC tasks, We compare LDNet to BERT (Devlin et al., 2019), RoBERTa (Liu et al., 2019), DeBERTa-v3 (He et al., 2021), and Mirror.

4.2 Main Results

LDNet main results over 16 IE and MIE datasets are shown in Table 1 and Table 2. We can observe that:

1) By adopting multi-aspect relation modeling and applying label drop on separate probability matrix, LDNet offers an effective methodology for IE and MIE. LDNet achieves state-of-the-art performance across almost all datasets and tasks. Although LDNet slightly underperforms FSUIE on ACE05-RE and underperforms GoLLIE on ACE05-NER, it surpasses them in other tasks and covers a broader range of tasks, including MRC, classification, discontinuous NER, and hyper-RE. MRC and multi-modal results are in Appendix C.5.

2) Despite having a relatively small model scale, LDNet consistently delivers superior results across almost all IE tasks. LDNet outperforms DeepStruct (10B) in all tasks. The comparison of pre-trained language model parameter scales is included in Appendix C.4 and more comparisons with models of different scales can be found in the Appendix C.5. We also conduct ablation studies on LDNet, focusing on pre-training and fine-tuning strategies, as shown in Table 2. LDNet surpasses Mirror in almost all settings and across all datasets.

3) Model transfer learning provides a valuable strategy for enhancing performance across all datasets, enabling LDNet to leverage information sharing between teacher models and student models.

Besides the triplet-based IE and MIE tasks, LDNet also demonstrates its effectiveness in discontinuous NER and n-ary hyper RE tasks. We provide results with pre-training (w/ PT, w/ Inst) and without pre-training (w/o PT, w/ Inst). As presented in Table 3, LDNet achieves improvements over previous methods, increasing by 4.18% and 0.42% on the CADEC (Karimi et al., 2015) and Hyper-

		w/ PT & Inst.		w/ PT & w/o Inst.	
w/ Label Drop		✗	✓	✗	✓
NER	ACE05	87.12	**87.20**	86.45	**86.97**
	CoNLL03	93.75	**94.60**	94.93	**95.11**
RE	NYT	93.09	**94.43**	93.94	**94.41**
	SciERC	43.75	**43.94**	38.71	**46.27**
EE	ACE05-Tgg	63.55	**78.74**	70.40	**80.23**
	ACE05-Arg	49.37	**55.78**	46.67	**62.76**
ABSA	14-res	82.18	**83.58**	83.70	**84.79**
	14-lap	77.23	**78.11**	71.12	**71.72**
	15-res	63.91	**67.38**	65.23	**70.36**
	16-res	76.25	**78.15**	77.24	**78.17**

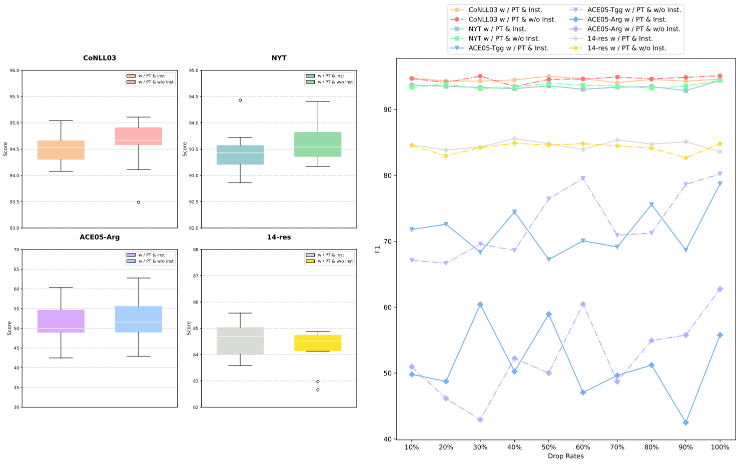

Figure 2: Results of the ablation study on the label drop mechanism. Table on the left shows LDNet's performance with and without the Label Drop mechanism, box plot in the middle and line chart on the right illustrate LDNet's performance under different drop rates.

RED (Chia et al., 2022) datasets, respectively.

4.3 Few-shot Results & Zero-shot Results

Followed by Zhu et al. (2023), we analyze LDNet's quick adaptation ability on NER, RE, EE, and ABSA tasks under 1-shot, 5-shot, and 10-shot settings. We compared LDNet with strong baselines such as UIE, USM, and Mirror. Table 5 shows the superior performance of LDNet under low-resource settings. On average, LDNet improves results by 3.41%, 2.26%, and 1.68% under 1-shot, 5-shot, and 10-shot settings, respectively. Additionally, LDNet achieves average improvements of 1.27%, 0.92%, 3.52%, and 3.02% over the NER, RE, EE, and ABSA tasks, respectively.

We also examine LDNet's extendability in the zero-shot setting using NER datasets from 7 distinct fields (Liu et al., 2013, 2021). LDNet is compared with USM, InstructUIE, Mirror and Llama-3-8B (Dubey et al., 2024). The zero-shot results are presented in Table 4. LDNet consistently outperforms Mirror and Llama-3-8B across all evaluated datasets. Although LDNet scores 0.89 lower than USM on the Literature dataset and is slightly lower than InstructUIE on the Movie dataset, the parameter scale of LDNet's pre-trained language model is smaller than that of USM's, and much smaller than that of InstructUIE's. Additionally, LDNet still achieves the highest average performance overall.

4.4 Analysis on Label Drop

To investigate the effectiveness of our label drop mechanism, we conduct ablation studies under two settings: with and without label drop. We fine-tune for 100 epochs on the IE datasets in both of the settings to fully exploit the potential of the label drop mechanism. And in order to better demonstrate the effectiveness of the label drop mechanism, we do not perform model transfer learning in the ablation study. The results are shown in Figure 2. Compared to the results of LDNet with only the multi-aspect relation modeling component, the model with the label drop mechanism shows improved performance on most datasets. It achieves an improvement of 1.93% on average for the ABSA task and a substantial increase of 11.88% on the ACE05 dataset for the EE task.

To further analyze the capability of the label drop mechanism, we conduct experiments on 4 text datasets involving different tasks. We randomly drop different portions of the probability matrix and test the performance. We test drop rates of 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, and 100%. We also fine-tune 100 epochs for the analysis. The results are shown in Figure 2. More detailed results can be found in the Appendix C.5.

We can see that LDNet exhibits stable and strong performance on the NER, RE, and ABSA tasks, under different drop rates, demonstrating the robustness of the label drop mechanism. For the EE task, the performance is more volatile, with the best performance occurring under the full drop setting, which is as expected. The fluctuations in performance on the ACE05 dataset in Figure 2 are due to the large variety of label texts in the schema labels. The label texts following [LM] include 33 types, such as 'transport', 'elect', 'start position', 'nominate', 'end position', 'attack', 'meet', 'marry',

Task	Datasets	Label Drop Accuracy
NER	CoNLL03	93.04
RE	ACE05	94.69
	CoNLL04	87.98
ABSA	14-lap	92.61
	15-res	89.82
	16-res	90.41
Avg.		91.76

Table 6: Results of the accuracy of label drop mechanism.

Task	Datasets	Mirror w/o PT w/ Inst.	LDNet$_{MR}$ w/o PT w/ Inst.	LDNet$_{LD}$ w/o PT w/ Inst.	LDNet$_{MT-}$ w/o PT w/ Inst.
NER	CoNLL03	92.11	92.84	93.24	**93.30**
RE	ACE05	64.88	65.76	68.09	**69.10**
	CoNLL04	71.19	75.11	71.97	**75.81**
ABSA	14-lap	59.56	62.94	60.75	**65.32**
	15-res	60.26	61.25	60.69	**66.30**
	16-res	73.13	76.81	75.76	**77.97**
Avg.		70.19	72.12	71.75	**74.30**

Table 7: Results of the ablation study on the multi-aspect relation modeling mechanism. LDNet$_{MR}$ represents LDNet using only Multi-aspect Relation Modeling, LDNet$_{LD}$ represents LDNet using only Label Drop, and LDNet$_{MT-}$ indicates LDNet using both Multi-aspect Relation Modeling and Label Drop.

'phone write', and so on. The label texts following [LR] have even more varieties, totaling 104 types. When only part of the probability matrix is dropped, some irrelevant token pairs' TA, A2A, or AS relation probabilities may not be dropped and could still exceed the threshold. As a result, non-existent TA, A2A, and AS relations may be included in the decoding process, and if they form a cycle, non-existent relations are predicted. Additionally, since we are **randomly** dropping portions of the probability matrix, even with a low drop rate, there is still a chance to accurately filter out token pairs that are unlikely to have relations. The content of the schema labels can be found in Appendix A.1, and the specific process of relation cycling can be referenced in Appendix A.2 or Appendix A.3. The stable high performance of other datasets is attributed to the significantly fewer types of label texts; for example, the CoNLL03 dataset has only four types: 'miscellaneous', 'person', 'location', and 'organization'. Even with a very low drop rate like 10%, the difficulty of filtering out non-existent relations is much lower in such a small range.

We also evaluate the accuracy of the label drop mechanism separately. The label drop probabilities $\hat{l}^r$ can be used to assess the accuracy of the label drop model. The accuracy can be computed as follows: we repeat $\hat{l}^r \in \mathbb{R}^{1\times|x|}$ to match the shape of the label matrices to create probability matrices $A^r \in \mathbb{R}^{|x|\times|x|}$, $r \in \{TA, A2A, AS\}$. Values in A^r below 0.5 are considered 0, while values above 0.5 are considered 1. We then compare this to the label matrices, and the number of correct values divided by the total number of values gives the accuracy. We test the accuracy, and the results are shown in the Table 6. It can be seen that the label drop accuracy is generally high, with an average accuracy exceeding 90%.

4.5 Ablation Results of Multi-aspect Relation Modeling

We further conduct experiments on the Multi-aspect Relation Modeling mechanism. In these experiments, we utilize only the Multi-aspect Relation Modeling mechanism of the LDNet model, without incorporating Label Drop or Model Transfer Learning. We report the performance of LDNet under the 'w/o PT w/ Inst.' setting, with the selected comparison baseline being the performance of Mirror under the same setting. The results are shown in the Table 7. As can be seen, LDNet$_{MR}$ still outperforms Mirror under the same setting, demonstrating the effectiveness of the Multi-aspect Relation Modeling mechanism. The presence of Multi-aspect Relation Modeling reduces decision confusion while also creating a more suitable environment for the label drop mechanism. So it can be shown in the table that LDNet$_{MT-}$ achieves the best performance among the three: LDNet$_{MR}$, LDNet$_{LD}$, and LDNet$_{MT-}$.

5 Conclusion and Discussion

In this paper, we propose LDNet, a novel network that combines multi-aspect relation modeling and a label drop mechanism. LDNet assigns different relations to different levels for understanding and decision-making, thereby reducing decision confusion. By introducing the label drop mechanism, LDNet alleviates the influence of irrevelant relations. Experimental results show that LDNet achieves highly competitive performance across 9 tasks, in both single-modal and multi-modal, few-shot and zero-shot settings, which verifies its effetiveness and universality.

Limitations

1) The total quantity and variety of MIE datasets are not enough, so LDNet cannot be pre-trained on a relatively large-scale dataset for MIE as it can be for IE, and since LDNet is a universal IE and UIE solution, its performance on certain multi-modal datasets may not be as good as models specifically designed for multi-modal tasks. 2) Due to the maximum input length constraint, LDNet may experience a performance decline in document-level information extraction.

Ethical Considerations

If the model is able to extract information of high quality, it may be able to extract personal privacy information such as names, addresses, and phone numbers from large text datasets. This information could potentially be used for illegal monitoring, harassment, and other malicious purposes. Establishing appropriate privacy protection mechanisms and usage restrictions can be applied to ensure that the extracted information is only used for legitimate purposes and not abused.

References

Peggy M. Andersen, Philip J. Hayes, Alison K. Huettner, Linda M. Schmandt, Irene B. Nirenburg, and Steven P. Weinstein. 1992. Automatic extraction of facts from press releases to generate news stories. In *ANLC*, page 170–177.

F. Chen and Yujian Feng. 2023. Chain-of-thought prompt distillation for multimodal named entity and multimodal relation extraction. *ArXiv*, abs/2306.14122.

Xiang Chen, Ningyu Zhang, Lei Li, Yunzhi Yao, Shumin Deng, Chuanqi Tan, Fei Huang, Luo Si, and Huajun Chen. 2022. Good visual guidance make a better extractor: Hierarchical visual prefix for multimodal entity and relation extraction. In *NAACL*, pages 1607–1618.

Yew Ken Chia, Lidong Bing, Sharifah Mahani Aljunied, Luo Si, and Soujanya Poria. 2022. A dataset for hyper-relational extraction and a cube-filling approach. In *EMNLP*, pages 10114–10133.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

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

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *IWP@IJCNLP*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Hao Fei, Shengqiong Wu, Jingye Li, Bobo Li, Fei Li, Libo Qin, Meishan Zhang, Min Zhang, and Tat-Seng Chua. 2022. Lasuie: Unifying information extraction with latent adaptive structure-aware generative language model. In *NeurIPS*.

Ralph Grishman. 2019. Twenty-five years of information extraction. *Nat. Lang. Eng.*, 25(6):677–692.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2021. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. *CoRR*, abs/2111.09543.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Xuming Hu, Zhijiang Guo, Zhiyang Teng, Irwin King, and Philip S. Yu. 2023. Multimodal relation extraction with cross-modal retrieval and synthesis. In *ACL*, pages 303–311.

Martin Josifoski, Nicola De Cao, Maxime Peyrard, Fabio Petroni, and Robert West. 2022. GenIE: Generative information extraction. In *NAACL*, pages 4626–4643.

Sarvnaz Karimi, Alejandro Metke-Jimenez, Madonna Kemp, and Chen Wang. 2015. Cadec: A corpus of adverse drug event annotations. *J. Biomed. Informatics*, 55:73–81.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL*, pages 7871–7880.

Jiajia Li, Lu Yang, Mingni Tang, Cong Chen, Zuchao Li, Ping Wang, and Hai Zhao. 2024a. The music maestro or the musically challenged, a massive music evaluation benchmark for large language models. *Preprint*, arXiv:2406.15885.

Jingye Li, Hao Fei, Jiang Liu, Shengqiong Wu, Meishan Zhang, Chong Teng, Donghong Ji, and Fei Li. 2022. Unified named entity recognition as word-word relation classification. In *AAAI*, pages 10965–10973.

Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. 2019. Visualbert: A simple and performant baseline for vision and language. *CoRR*, abs/1908.03557.

Zixuan Li, Yutao Zeng, Yuxin Zuo, Weicheng Ren, Wenxuan Liu, Miao Su, Yucan Guo, Yantao Liu, Xiang Li, Zhilei Hu, et al. 2024b. Knowcoder: Coding structured knowledge into llms for universal information extraction. *arXiv preprint arXiv:2403.07969*.

Xiaobo Liang, Lijun Wu, Juntao Li, Yue Wang, Qi Meng, Tao Qin, Wei Chen, Min Zhang, and Tie-Yan Liu. 2021. R-drop: Regularized dropout for neural networks. In *NeurIPS*, pages 10890–10905.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *ACL*, pages 7999–8009.

Jingjing Liu, Panupong Pasupat, Scott Cyphers, and James R. Glass. 2013. Asgard: A portable architecture for multilingual dialogue systems. In *ICASSP*, pages 8386–8390.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

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*, pages 13452–13460.

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

Jie Lou, Yaojie Lu, Dai Dai, Wei Jia, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2023. Universal information extraction as unified semantic matching. *CoRR*, abs/2301.03282.

Di Lu, Leonardo Neves, Vitor Carvalho, Ning Zhang, and Heng Ji. 2018. Visual attention model for name tagging in multimodal social media. In *ACL*, pages 1990–1999.

Yaojie Lu, Qing Liu, Dai Dai, Xinyan Xiao, Hongyu Lin, Xianpei Han, Le Sun, and Hua Wu. 2022. Unified structure generation for universal information extraction. In *ACL*, pages 5755–5772.

Yi Luan, Luheng He, Mari Ostendorf, and Hannaneh Hajishirzi. 2018. Multi-task identification of entities, relations, and coreference for scientific knowledge graph construction. In *EMNLP*, pages 3219–3232.

Alexis Mitchell, Stephanie Strassel, Shudong Huang, and Ramez Zakhary. 2005. Ace 2004 multilingual training corpus.

Giovanni Paolini, Ben Athiwaratkun, Jason Krone, Jie Ma, Alessandro Achille, Rishita Anubhai, Cícero Nogueira dos Santos, Bing Xiang, and Stefano Soatto. 2021. Structured prediction as translation between augmented natural languages. In *ICLR*.

Tianshuo Peng, Zuchao Li, Ping Wang, Lefei Zhang, and Hai Zhao. 2023a. A novel energy based model mechanism for multi-modal aspect-based sentiment analysis. *Preprint*, arXiv:2312.08084.

Tianshuo Peng, Zuchao Li, Lefei Zhang, Bo Du, and Hai Zhao. 2023b. FSUIE: A novel fuzzy span mechanism for universal information extraction. In *ACL*, pages 16318–16333.

Yang Ping, JunYu Lu, Ruyi Gan, Junjie Wang, Yuxiang Zhang, Pingjian Zhang, and Jiaxing Zhang. 2023. UniEX: An effective and efficient framework for unified information extraction via a span-extractive perspective. In *ACL*, pages 16424–16440.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Ion Androutsopoulos, Suresh Manandhar, Mohammad AL-Smadi, Mahmoud Al-Ayyoub, Yanyan Zhao, Bing Qin, Orphée De Clercq, Véronique Hoste, Marianna Apidianaki, Xavier Tannier, Natalia Loukachevitch, Evgeniy Kotelnikov, Nuria Bel, Salud María Jiménez-Zafra, and Gülşen Eryiğit. 2016. SemEval-2016 task 5: Aspect based sentiment analysis. In *SemEval-2016*, pages 19–30.

Maria Pontiki, Dimitris Galanis, Haris Papageorgiou, Suresh Manandhar, and Ion Androutsopoulos. 2015. SemEval-2015 task 12: Aspect based sentiment analysis. In *SemEval 2015*, pages 486–495.

Maria Pontiki, Dimitris Galanis, John Pavlopoulos, Harris Papageorgiou, Ion Androutsopoulos, and Suresh Manandhar. 2014. SemEval-2014 task 4: Aspect based sentiment analysis. In *SemEval 2014*, pages 27–35.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *ACL*, pages 784–789.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *ECML-PKDD*, pages 148–163.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *HLT-NAACL*, pages 1–8.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2024. Gollie: Annotation guidelines improve zero-shot information-extraction. *Preprint*, arXiv:2310.03668.

Taneeya Satyapanich, Francis Ferraro, and Tim Finin. 2020. Casie: Extracting cybersecurity event information from text. In *AAAI*, pages 8749–8757.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, pages 1631–1642.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *JMLR*, pages 1929–1958.

Jianlin Su, Ahmed Murtadha, Shengfeng Pan, Jing Hou, Jun Sun, Wanwei Huang, Bo Wen, and Yunfeng Liu. 2022. Global pointer: Novel efficient span-based approach for named entity recognition. *CoRR*, abs/2208.03054.

Lin Sun, Kai Zhang, Qingyuan Li, and Renze Lou. 2024. Umie: Unified multimodal information extraction with instruction tuning. In *AAAI*, volume 38, pages 19062–19070.

Qingyu Tan, Ruidan He, Lidong Bing, and Hwee Tou Ng. 2022. Document-level relation extraction with adaptive focal loss and knowledge distillation. In *ACL*, pages 1672–1681.

Jinhao Tian, Zuchao Li, Jiajia Li, and Ping Wang. 2023. N-gram unsupervised compoundation and feature injection for better symbolic music understanding. *Preprint*, arXiv:2312.08931.

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

David Wadden, Ulme Wennberg, Yi Luan, and Hannaneh Hajishirzi. 2019. Entity, relation, and event extraction with contextualized span representations. In *EMNLP-IJCNLP*, pages 5784–5789.

Christopher Walker, Stephanie Strassel, Julie Medero, and Kazuaki Maeda. 2006. Ace 2005 multilingual training corpus.

Jianqiang Wan, Sibo Song, Wenwen Yu, Yuliang Liu, Wenqing Cheng, Fei Huang, Xiang Bai, Cong Yao, and Zhibo Yang. 2024. Omniparser: A unified framework for text spotting key information extraction and table recognition. In *CVPR*, pages 15641–15653.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.

Chenguang Wang, Xiao Liu, Zui Chen, Haoyun Hong, Jie Tang, and Dawn Song. 2022a. DeepStruct: Pretraining of language models for structure prediction. In *ACL*, pages 803–823.

Jue Wang and Wei Lu. 2020. Two are better than one: Joint entity and relation extraction with table-sequence encoders. In *EMNLP*, pages 1706–1721.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, Jihua Kang, Jingsheng Yang, Siyuan Li, and Chunsai Du. 2023. Instructuie: Multi-task instruction tuning for unified information extraction. *CoRR*, abs/2304.08085.

Xinyu Wang, Jiong Cai, Yong Jiang, Pengjun Xie, Kewei Tu, and Wei Lu. 2022b. Named entity and relation extraction with multi-modal retrieval. In *EMNLP*, pages 5925–5936.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Trans. Assoc. Comput. Linguistics*, 7:625–641.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL*, pages 1112–1122.

Zhiwei Wu, Changmeng Zheng, Yi Cai, Junying Chen, Ho-fung Leung, and Qing Li. 2020. Multimodal representation with embedded visual guiding objects for named entity recognition in social media posts. In *MM*, pages 1038–1046.

Jianfei Yu, Jing Jiang, Li Yang, and Rui Xia. 2020. Improving multimodal named entity recognition via entity span detection with unified multimodal transformer. In *ACL*, pages 3342–3352.

Xin Cong. Bowen Yu, Mengcheng Fang, Tingwen Liu, Haiyang Yu, Zhongkai Hu, Fei Huang, Yongbin Li, and Bin Wang. 2023. Universal information extraction with meta-pretrained self-retrieval. *Preprint*, arXiv:2306.10444.

Dong Zhang, Suzhong Wei, Shoushan Li, Hanqian Wu, Qiaoming Zhu, and Guodong Zhou. 2021. Multimodal graph fusion for named entity recognition with targeted visual guidance. In *AAAI*, pages 14347–14355.

Qi Zhang, Jinlan Fu, Xiaoyu Liu, and Xuanjing Huang. 2018a. Adaptive co-attention network for named entity recognition in tweets. In *AAAI*, pages 5674–5681.

Qi Zhang, Jinlan Fu, Xiaoyu Liu, and Xuanjing Huang. 2018b. Adaptive co-attention network for named entity recognition in tweets. In *AAAI*, pages 5674–5681.

Zheng Zhang, Zili Zhou, and Yanna Wang. 2022. SSEGCN: Syntactic and semantic enhanced graph convolutional network for aspect-based sentiment analysis. In *NAACL*, pages 4916–4925.

Changmeng Zheng, Junhao Feng, Yi Cai, Xiaoyong Wei, and Qing Li. 2023. Rethinking multimodal entity and relation extraction from a translation point of view. In *ACL*, pages 6810–6824.

Changmeng Zheng, Zhiwei Wu, Junhao Feng, Ze Fu, and Yi Cai. 2021. Mnre: A challenge multimodal dataset for neural relation extraction with visual evidence in social media posts. In *ICME*, pages 1–6.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *NAACL*, pages 50–61.

Tong Zhu, Junfei Ren, Zijian Yu, Mengsong Wu, Guoliang Zhang, Xiaoye Qu, Wenliang Chen, Zhefeng Wang, Baoxing Huai, and Min Zhang. 2023. Mirror: A universal framework for various information extraction tasks. In *EMNLP*, pages 8861–8876.

A Methodology

A.1 Schema Labels

When it comes to the specific implementation, the schema labels are divided into two parts: a set of special tokens `[LM]`, `[LR]`, `[LC]`, `[TL]`, `[TP]`, and `[B]` and their corresponding label text.

`[LM]`, `[LR]`, and `[LC]` are the special tokens that represent labels of entity mention, relation, and text classification respectively. Only one of the three special tokens `[TL]`, `[TP]`, and `[B]` will appear in a single data instance. `[TP]` is used for the MRC task, `[B]` is used for the classification task, and `[TL]` is used for all other tasks.

The special tokens `[LM]`, `[LR]`, `[LC]` will be followed by their corresponding label text, which are the tokenized strings for the entity mention, relation, and text classification labels. All the labels that appear in the dataset will be included in the schema labels, such as `[LC]`, "`_correct`", `[LC]`, "`_wrong`" for the classification task with "correct" and "wrong" as the two labels.

We use the special tokens in the schema labels as the trigger words and utilize them to guide the relation extraction process. A relation will only be extracted when the trigger word is activated. The label drop operation sets the elements in the label vector corresponding to the special tokens that in gold spans to 1, while the other schema label elements remain 0. For example, in the classification task, if a data instance is classified as "correct", the element corresponding to the `[LC]` token before "_correct" token will be set to 1, while the element corresponding to the "_correct" token will remain 0.

A.2 Handling of Unknown Schemas

If the schema is unknown, LDNet removes the schema from the original NER input format, transforming it from the format of *instruction + schema + text*:

```
[I] Please identify possible entities
from the given text and determine
their types [LM] person [LM] location
[LM] organization [TL] Jerry Smith
is a friend of Tom
```

to:

```
[I] Please identify possible entities
from the given text and determine
their types [TP] Jerry Smith
is a friend of Tom
```

Here, `[I] Please identify possible entities from the given text and determine their types` is the instruction part, with `[I]` being a special token indicating the start of the instruction; `[LM] person [LM] location [LM] organization` represents the schema, with `[LM]` being a special token representing an entity type, such as `[LM] person`, indicating the person type; `[TL] Jerry Smith is a friend of Tom` is the text part, with `[TL]` being a special token indicating that the text following it requires not only span extraction but also the extraction of types within the schema. For instance, to extract the entity `Jerry Smith`, it's necessary to extract both its span and its corresponding type, **person**, which is associated with the span of the `[LM]` token in `[LM] person` in the schema. In the input without a schema, `[TP]` indicates that only entity spans need to be extracted from the text, without requiring the extraction of additional information such as types in schemas.

Thus, LDNet handles unknown schemas by processing the input as follows:

```
[I] Please identify possible entities
from the given text and determine their
types
[TP] Jerry Smith is a friend of Tom
```

From the text *"Jerry Smith is a friend of Tom"*, LDNet directly extracts the spans of entities such as `Jerry Smith` and `Tom` as the output.

A.3 Handling of Discontinuous NER and Nested NER

Discontinuous NER LDNet handles discontinuous NER in a manner similar to how it handles regular NER. Suppose the input is:

Parameter	Value	Parameter	Value
warmup proportion	0.1	batch size	4
pre-training epochs	3	PLM learning rate	2e-5
fine-tuning epochs	20	PLM weight decay	0.1
fine-tuning epoch patience	3	others learning rate	1e-4
few-shot epochs	200	max gradient norm	1.0
few-shot epoch patience	10	d_h	1024
$d_v^\diamondsuit$	1024	batch size$^\diamondsuit$	32
PLM learning rate$^\diamondsuit$	3e-5	others learning rate$^\diamondsuit$	1e-4
fine-tuning epochs$^\diamondsuit$	20	PLM weight decay$^\diamondsuit$	0.1
warmup proportion$^\diamondsuit$	0.01	$\alpha^\diamondsuit$	0.5

Table 8: The parameters marked with $^\diamondsuit$ are for the multi-modal experiments.

NER Pre-training Dataset	Instruction	Instance	RE Pre-training Dataset	Instruction	Instance
AnatEM	42	5,861	ADE_corpus	9	3,417
bc2gm	42	12,500	FewRel	9	20,000
bc4chemd	42	20,000	GIDS	9	8,526
bc5cdr	42	4,560	kbp37	9	15,807
Broad_Tweet_Corpus	42	5,334	New-York-Times-RE	9	20,000
FabNER	42	9,435	NYT11HRL	9	20,000
FindVehicle	42	20,000	semeval	9	8,000
GENIA	42	15,023	WebNLG	9	5,019
HarveyNER	42	3,967	Wiki-ZSL	9	23,107
MultiNERD	42	20,000	MNRE$^\diamondsuit$	9	12,248
NCBIdiease	42	5,432	MRC Pre-training Dataset	Instruction	Instance
ontoNotes5	42	20,000	BiPaR	11,524	11,668
TweetNER7	42	7,103	ms_marco_v2.1	20,000	20,000
WikiANN_en	42	20,000	newsqa	19,659	20,000
WNUT-16	42	2,394	squad_v2	19,998	20,000
Twitter2015$^\diamondsuit$	42	4,000	SubjQA	4,060	13,990
Twitter2017$^\diamondsuit$	42	3,376	EE Pre-training Dataset	Instruction	Instance
			PHEE	40	2,898

Table 9: The datasets marked with $^\diamondsuit$ are for the multi-modal experiments.

```
[I] Please identify possible entities
from the given text
and determine their types
[LM] person [LM] title
[LM] organization [TL] The CEO of Tesla
, Elon Musk, made an announcement today
```

LDNet will extract the following relations:

- TA Relation: Between `[LM]` before **person** and `CEO of Tesla`, and `[LM]` before **person** and `Elon Musk`.
- A2A Relation: Between `CEO of Tesla` and `Elon Musk`.
- AS Relation: Between `CEO of Tesla` and `CEO of Tesla`, and `Elon Musk` and `Elon Musk`.

The closed loop formed by the TA relation between `[LM]` before **person** and `CEO of Tesla`, the AS relation between `CEO of Tesla` and `CEO of Tesla`, the A2A relation between `CEO of Tesla` and `Elon Musk`, the AS relation between `Elon Musk` and `Elon Musk`, and the TA relation between `[LM]` before **person** and `Elon Musk` allows LDNet to extract `CEO of Tesla, Elon Musk` as a discontinuous entity of the **person** type.

Nested NER The input format for nested NER is:

```
[I] Please identify possible entities
from the given text and
determine their types
[LM] organization [LM] title
[LM] person [LM] location
[TL] Apple CEO Tim Cook
gave a speech at Stanford University
in California
```

Classification Pre-training Dataset	Instruction	Instance	Classification Pre-training Dataset	Instruction	Instance
ag_news	5	5,000	ANLI	29	15,000
ARC	3,361	3,370	CoLA	43	5,000
CosmosQA	4,483	5,000	cos_e	5,000	5,000
dbpedia	6	5,000	DREAM	3,842	5,000
hellaswag	20	5,000	IMDB	26	5,000
MedQA	5,000	5,000	MNLI	29	5,000
MRPC	40	3,668	MultiRC	4,999	5,000
OpenBookQA	4,835	4,957	QASC	4,832	5,000
QNLI	31	5,000	QQP	40	5,000
RACE	4,482	5,000	RACE-C	4,782	5,000
ReClor	3,368	4,638	RTE	29	2,490
SciQ	4,989	5,000	SNLI	29	5,000
SST-2	26	5,000	Winogrande	20	5,000
WNLI	31	635			

Table 10: The Classification Pre-training Datasets.

Model	TANL	UIE	DeepStruct	InstructUIE	USM	Mirror	LDNet
Computational Complexity	$O(n^3)$	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n^2)$
PLM	T5-base	T5-large	GLM	FlanT5	RoBERTa-large	DeBERTa-v3-large	DeBERTa-v3-large
PLM Params	220M	770M	10B	11B	372M	304M	304M

Table 11: Computational complexity and model parameters of various models.

In this example, `Apple` is an organization entity representing the Apple company, and `Apple CEO` is a title entity representing the CEO position of Apple.

After obtaining the TA, A2A, and AS relation probability matrices, LDNet will extract the following relations:

- TA Relation: Between `[LM]` before **organization** and `Apple`, between `[LM]` before **title** and `Apple`, and `[LM]` before **title** and `CEO`.
- AS Relation: Between `Apple` and `Apple`; between `Apple` and `CEO`.

The closed loop formed by the TA relation between `[LM]` before **organization** and `Apple`, and the AS relation between `Apple` and `Apple` allows LDNet to extract `Apple` as an organization entity. Similarly, the TA relations between `[LM]` before **title** and `Apple`, the AS relation between `Apple` and `CEO`, and the TA relation between `[LM]` before **title** and `CEO` form a closed loop, allowing LDNet to extract `Apple CEO` as a title entity.

The A2A relation is not mandatory, as the information to be extracted may not involve two arguments, such as in NER tasks.

A.4 Underlying Logic of Label Drop Mechanism

The underlying logic of label drop mechanism is that if the i-th token does not exist in the gold answer, after training, the value of $\hat{l}^r_i$ at position i in $\hat{l}^r$ will close to 0, it suppresses the value of $s^r_{\cdot i}$ in the i-th column of the S^r matrix. During subsequent decoding, if the value of $s^r_{\cdot i}$ is suppressed below a threshold, LDNet will not consider extracting the relation between position i and other positions, therefore filtering out token pairs that is impossible to have relations. As shown in Figure 1, for the convenience of observation, we only depict the prediction of the A2A relation. And to better illustrate the concept of label drop, we hypothesize an extreme scenario where $\hat{l}^r$ is the same as $\mathbf{l}^r$. For instance, the probability of a relation existing between “playwright” and “born” is set to 0.

A.5 LDNet’s Contributions Relative to Prior Multi-modal Approaches

Multi-modal representation is a conventional approach. LDNet utilizes multi-modal representation to enable its application to multi-modal data. LDNet is a universal information extraction method, its capability is general and not differentiated by whether the data is multi-modal or unimodal. Compared to prior multi-modal approaches, the contribution of LDNet lies not in combining multi-modal

Task	Datasets	TANL	UIE	USM	FSUIE-base	UniEX-large	MetaRetriever	LDNet$_{deberta-v3-base}$	LDNet$_{deberta-v3-large}$
NER	ACE04	-	86.89	87.62	85.24	87.12	86.10	88.50	88.69
	ACE05	84.90	85.78	87.14	86.22	87.02	84.01	88.18	87.79
	CoNLL03	91.70	92.99	93.16	-	92.65	92.38	93.43	93.44
RE	ACE05	63.70	66.06	67.88	72.29	66.06	64.37	69.33	69.14
	CoNLL04	71.40	75.00	78.84	-	73.40	73.66	79.26	80.79
	NYT	-	93.54	94.07	-	-	-	94.15	94.96
	SciERC	-	36.53	37.36	-	38.00	35.77	37.49	46.85
EE	ACE05-Tgg	68.40	73.36	72.41	-	74.08	72.38	73.42	83.56
	ACE05-Arg	47.60	54.79	55.83	-	53.92	52.62	57.04	61.14
	CASIE-Tgg	-	69.33	71.73	-	71.46	69.76	72.77	73.36
	CASIE-Arg	-	61.30	63.26	-	62.91	60.37	63.35	63.88
ABSA	14-res	-	74.52	77.26	74.17	74.77	73.41	79.08	79.17
	14-lap	-	63.88	65.51	65.56	65.23	62.83	68.84	69.00
	15-res	-	67.15	69.86	70.63	68.58	65.85	74.47	71.18
	16-res	-	75.07	78.25	75.80	76.02	73.55	78.40	78.85

Table 12: Comparison to the performance of models with smaller pre-trained language models.

Task	Datasets	GoLLIE Baseline	GoLLIE	GoLLIE-13B	GoLLIE-34B	LDNet Baseline	LDNet$_{deberta-v3-base}$	LDNet$_{deberta-v3-large}$
NER	ACE05	89.10	88.10	89.40	**89.60**	85.70	88.18	87.79
	CoNLL03	92.90	92.80	93.00	93.10	92.73	93.43	**93.44**
RE	ACE05	63.80	63.60	67.50	**70.10**	67.86	69.33	69.14
EE	ACE05-Tgg	71.7	72.2	70.9	71.9	73.05	73.42	**83.56**
	ACE05-Arg	65.9	66.0	67.8	**68.6**	54.73	57.04	61.14
	CASIE-Tgg	33.9	59.3	62.2	65.5	71.60	72.77	**73.36**
	CASIE-Arg	47.9	50.0	52.6	55.2	61.04	63.35	**63.88**

Table 13: Comparison to GoLLIE models.

information, but rather in its Label Drop mechanism, which effectively filters out irrelevant token pairs.

Individual Contributions of Image Features Since images only provide clue information; the information extraction ultimately comes from the text. Therefore, it is not possible to explore the individual contributions of image features on the model's performance.

B Related Work

Dropout Strategy Dropout (Srivastava et al., 2014) is a powerful technique usually used to regularize the training of deep neural networks. R-Drop (Liang et al., 2021) forces the output distributions of different submodels, sampled by dropout, to be consistent with each other by minimizing their bidirectional KL-divergence. LDNet transfers the idea of dropout into IE tasks and applies label drop to remove unrelational token pairs, forcing the model to concentrate on relational ones.

Model Transfer Learning Averaging the predictions of all trained models is a simple yet effective way to enhance the performance of almost any machine learning algorithm. However, it can be cumbersome and computationally expensive. Therefore, (Hinton et al., 2015) proposed Knowledge Distillation (KD) to compress the knowledge of an ensemble into a single model. (Tan et al., 2022) utilizes KD in document-level relation extraction. The system trains a teacher model on the distantly-supervised data and uses the distributions generated by the teacher model as soft labels to pre-train the student model. The authors found that distilling with the MSE loss performs better than using KL-divergence. LDNet incorporates this idea and follows the same setting, using MSE loss to minimize the difference between the distributions of the teacher model and the student model.

Multi-modal Information Extraction Unlike traditional information extraction, which relies exclusively on single-modal data, Multi-modal Information Extraction (MIE) leverages auxiliary visual cues from images to supplement the missing context. MEGA (Zheng et al., 2021) is the first to propose the MNRE dataset and introduces a dual image alignment method to capture aligned information between visual object and textual object. UMT (Yu et al., 2020), which pioneers the use of Transformer in the MIE task, utilizes a multi-modal interaction module to integrate token representa-

Model	SQuAD 2.0 (EM/F1)	CoLA (Mcc)	QQP (Acc)	MNLI (Acc)	SST-2 (Acc)	QNLI (Acc)	RTE (Acc)	MRPC (Acc)
BERT-large	79.0 / 81.8	60.6	91.3	-	93.2	92.3	70.4	84.1
RoBERTa-large	86.5 / 89.4	68.0	92.2	90.2	96.4	93.9	86.6	88.8
DeBERTa v3-large	89.0 / 91.5	75.3	93.0	91.9	96.9	96.0	92.7	92.2
Mirror	40.4 / 67.4	63.9	84.8	85.9	93.6	91.6	85.9	89.2
LDNet	42.5 / 72.0	74.2	86.2	87.2	94.8	92.8	89.2	91.0

Table 14: Results on MRC and classification tasks.

Modality	**Methods**	Twitter-2015			Twitter-2017			MNRE		
		Prec.	Rec.	F1	Prec.	Rec.	F1	Prec.	Rec.	F1
Text+Image	AdapCoAtt (Zhang et al., 2018a)	69.87	74.59	72.15	85.13	83.20	84.10	-	-	-
	VisualBERT (Li et al., 2019)	68.84	71.39	70.09	84.06	85.39	84.72	57.15	59.48	58.30
	OCSGA (Wu et al., 2020)	74.71	71.21	72.92	-	-	-	-	-	-
	UMT (Yu et al., 2020)	71.67	75.23	73.41	85.28	85.34	85.31	62.93	63.88	63.46
	UMGF (Zhang et al., 2021)	74.49	75.21	74.85	86.54	84.50	85.51	64.38	66.23	65.29
	MEGA (Zheng et al., 2021)	70.35	74.58	72.35	84.03	84.75	84.39	64.51	68.44	66.41
	HVPNeT (Chen et al., 2022)	73.87	76.82	75.32	85.84	87.93	86.87	83.64	80.78	81.85
	MoRe (Wang et al., 2022b)	-	-	-	-	-	-	65.25	67.32	66.27
	TMR (Zheng et al., 2023)	75.26	76.49	75.87	88.12	88.38	88.25	90.48	87.66	89.05
	LDNet	76.79	75.56	76.17	87.51	87.64	87.57	75.45	76.15	75.79

Table 15: Comparison of LDNet's performance on MIE tasks with some MIE baselines.

tions with visual representations. MoRe (Wang et al., 2022b) enhances textual information retrieval by leveraging images and titles from search engines, thereby improving the accuracy of multimodal RE and NER tasks. Building on MoRe, MRE-RS (Hu et al., 2023) retrieves textual and visual evidence at different levels and further proposes a novel method to synthesize information for improved reasoning across the same and different modalities. CoTPD (Chen and Feng, 2023) demonstrates the elicitation of reasoning abilities from LLMs using CoT prompts across various dimensions and introduces a conditional prompt distillation method to transfer commonsense reasoning to a student model, enhancing its performance on text-only inputs. There are also some recently new methods, such as UMIE (Sun et al., 2024) and OmniParser (Wan et al., 2024).

C Experiments details

C.1 Hyperparameters and Pre-training Datasets

Specifically, we use vit-large-patch32-224-in21k as our image backbone. The experiments can be run using only 1 NVIDIA RTX 3090 with 24 GB memory. The hyperparameters and pre-training datasets are in Table 8, Table 9 and Table 10.

C.2 Experimental Configurations

The LDNet configurations 'w/ PT w/o Inst.' and 'w/ PT w/ Inst.' are not considered variants of the model; the only difference lies in whether the instruction part of the model input is an empty string. In 'w/ PT w/o Inst.' configuration, the instruction part is an empty string, while in 'w/ PT w/ Inst.', the instruction part is not empty and is a string similar to 'Please determine the two entities mentioned in the text and specify the nature of their relationship.'

We here give reasons why performance is often better in 'without PT and Inst.' configurations. The situation that configurations without PT and Inst. perform better is related to the dataset and the backbone model, DeBERTa-v3-large. For certain datasets like CoNLL04, 14-lap, and 15-res, better performance is observed under the 'with PT' configuration. On the other hand, some datasets, such as ACE04 and ACE05, are inherently large, and fine-tuning on their training sets alone yields good results, and the absence of pre-training does not introduce interference from other data, leading to better performance without PT. As shown in Table 2, the performance difference between

Dataset	Strategy	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
CoNLL03	w / PT & Inst.	94.74	94.35	94.29	94.49	95.04	94.68	94.08	94.56	94.29	94.60
	w / PT & w/o Inst.	94.69	94.11	95.06	93.49	94.57	94.62	94.92	94.65	94.88	95.11
NYT	w / PT & Inst.	93.72	93.51	93.36	93.16	93.59	93.07	93.38	93.48	92.86	94.43
	w / PT & w/o Inst.	93.35	93.86	93.17	93.39	93.96	93.70	93.54	93.25	93.54	94.41
ACE05-Tgg	w / PT & Inst.	71.79	72.59	68.33	74.45	67.23	70.07	69.14	75.58	68.66	78.74
	w / PT & w/o Inst.	67.10	66.67	69.57	68.61	76.43	79.55	70.92	71.25	78.61	80.23
ACE05-Arg	w / PT & Inst.	49.80	48.73	60.42	50.24	58.95	47.06	49.64	51.22	42.49	55.78
	w / PT & w/o Inst.	50.95	46.15	42.93	52.22	50.00	60.47	48.70	54.94	55.77	62.76
14-res	w / PT & Inst.	84.62	83.86	84.29	85.58	84.82	83.93	85.37	84.75	85.10	83.58
	w / PT & w/o Inst.	84.54	82.97	84.21	84.88	84.60	84.82	84.49	84.13	82.66	84.79

Table 16: Results of the label drop mechanism with different drop rates.

Dataset	Metric	10%	20%	30%	40%	50%	60%	70%	80%	90%	100%
Twitter-2015	P	76.22	76.52	78.22	77.61	76.36	75.93	75.56	79.45	76.74	76.79
	R	76.32	75.60	74.36	73.61	76.19	75.06	75.86	74.03	76.21	75.56
	F1	76.27	76.05	76.24	75.56	76.28	75.49	75.71	76.64	76.47	76.17
Twitter-2017	P	87.80	87.33	89.10	88.25	88.7	87.69	86.39	87.33	86.49	87.51
	R	87.34	86.75	86.68	86.75	87.27	87.49	87.86	87.79	87.19	87.64
	F1	87.57	87.04	87.88	87.50	87.99	87.59	87.12	87.56	86.84	87.57
MNRE	P	74.21	75.16	76.05	75.29	75.80	74.74	74.73	74.54	73.57	75.45
	R	73.79	74.97	72.99	72.49	74.72	74.41	74.04	72.37	72.61	76.15
	F1	74.00	75.06	74.49	73.86	75.26	74.57	74.39	73.44	73.09	75.79

Table 17: Results of the label drop mechanism with different drop rates on MIE datasets.

'with and without Inst.' is generally small. The distinction between these two configurations lies in whether the instruction part of LDNet's input is an empty string, and any difference should be related to DeBERTa-v3-large's instruction-following capability.

C.3 Method for Few-shot Experiments

Regarding the few-shot experiments, we employed fine-tuning. Following the settings of Mirror to ensure the fairness of the comparison, we fine-tuned for several epochs on the training set of the few-shot dataset before assessing few-shot capabilities.

C.4 Model Parameters and Computational Complexity

All models' computational complexities and the scales of the pre-trained language models used are listed in the Table 11.

Here, we define the following notations to analyze the computational complexity of the models:

- Expected answer length: k
- Text length: n
- Instruction length: l
- Schema length: s
- Hidden dimension: d
- Biaffine dimension: b
- Number of transformer layers: t

First, the computation for the hidden states of a transformer architecture model (excluding the computation for predicting token logits) is $(24nd^2 + 4dn^2) \cdot t$. The computation for predicting token logits is $2dvn$.

TANL, Deepstruct, UIE, and InstructUIE, which are generative unified information extraction methods, mainly focus on the computation involved in token prediction.

- **TANL** outputs a sequence that includes the answer to be extracted in the original input sequence, so its computation for sequence generation is $[(24nd^2+4dn^2) \cdot t + 2dvn] \cdot (n+k)$. Since the decoding algorithm used by TANL has a complexity of $O(n^2)$, its overall computational complexity is $O(n^3)$.
- **Deepstruct** outputs a sequence that is the answer sequence itself, so its computation is $[(24nd^2+4dn^2) \cdot t + 2dvn] \cdot k$, with a computational complexity of approximately $O(n^2)$.

w/ Label Drop		✗	✓
Twitter-2015	P	74.07	76.79
	R	60.57	75.56
	F1	66.64	76.17
Twitter-2017	P	84.42	87.51
	R	81.54	87.64
	F1	82.96	87.57
MNRE	P	74.04	75.45
	R	68.66	76.15
	F1	71.25	75.79

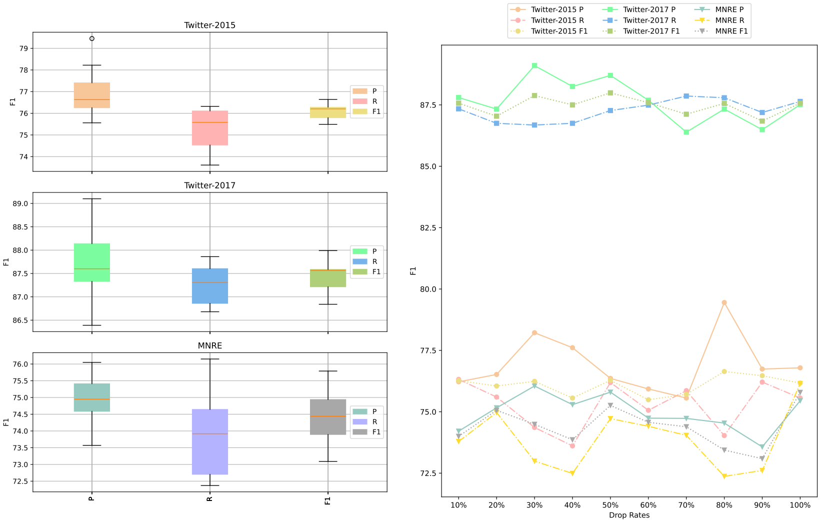

Figure 3: Results of the ablation study on the label drop mechanism on MIE tasks.

- **UIE** takes an input sequence that includes the schema and directly generates a Structured Extraction Language (SEL) as output. Its computation is $[(24(n+s)d^2 + 4d(n+s)^2) \cdot t + 2dv(n+s)] \cdot k$, with a computational complexity of approximately $O(n^2)$.

- **InstructUIE** includes the instruction, schema, and text in the input sequence. Its computation is $[(24(n+s+l)d^2 + 4d(n+s+l)^2) \cdot t + 2dv(n+s+l)] \cdot k$, with a computational complexity of approximately $O(n^2)$.

USM, Mirror, and LDNet belong to extractive unified information extraction methods, hence they do not include the computation for predicting token logits.

- **USM** includes schema and text in the input. In addition to encoder representations, it computes token-token linking scores, label-token linking scores, and token-label linking scores. Its computation is $(24(n+s)d^2+4d(n+s)^2) \cdot t + 8(n+s)d^2 + 2d(n+s)^2$, with a computational complexity of $O(n^2)$.

- **Mirror** includes instruction, schema, and text in the input. After computing the representations with the pre-trained model, it uses a biaffine transformation to generate score matrices. Its computation is $(24(n+s+l)d^2 + 4d(n+s+l)^2) \cdot t + 4ndb + 4nd + 6nb^2 + 6bn^2$, with a computational complexity of $O(n^2)$.

- **LDNet** includes instruction, schema, and text in the input. The Multi-aspect Relation Modeling module's computation is $(8nd^2+2dn^2) \cdot 3$. The label drop mechanism's computation is $3n^2$. Therefore, the total computation is $(24(n+s+l)d^2+4d(n+s+l)^2) \cdot t + (8nd^2 + 2dn^2) \cdot 3 + 3n^2$, with a computational complexity of $O(n^2)$.

C.5 Additional Experiments

Comparison to Models with Smaller PLMs We separately list the pretrained language model parameters smaller than the DeBERTa-v3-Large (304M) used by LDNet and compare them with LDNet using DeBERTa-v3-Base (86M). The results are shown in the Table 12.

Comparison to GoLLIE Models In the Table 13, we compare the results of GoLLIE and LDNet. For ease of comparison, we have included only the results where both models are evaluated.

MRC and Classification Results To demonstrate the compatibility of LDNet, we conducted experiments on SQuAD v2 (Rajpurkar et al., 2018) and the 7 GLUE datasets (Warstadt et al., 2019; Wang et al., 2019; Williams et al., 2018; Socher et al., 2013; Dolan and Brockett, 2005). As shown in Table 14, LDNet outperforms Mirror across all datasets. Additionally, LDNet surpasses BERT-large on SST-2 and QNLI, outperforms RoBERTa-large on CoLA, RTE, and MRPC, and achieves competitive results with DeBERTa v3-large on CoLA and MRPC. It is important to note that LDNet is a universal solution for IE and does not undergo full fine-tuning like Language Model Models (LLMs). Therefore, LDNet has limitations when it comes to tasks such as MRC and classification.

Consequently, it is reasonable to observe a slight performance gap in these tasks. However, it is worth noting that LDNet exhibits a smaller performance gap compared to Mirror.

Multi-modal Results We put detailed results of multi-modal experiments in Table 15. It can be seen that LDNet performs better than most baselines, and is slightly lower in some metrics on certain specific baselines. But these methods are all focused only on MIE, unlike LDNet which is a universal solution for both IE and MIE. And some methods like HVPNeT utilizes additional visual prefixes and uses a specially-designed pyramid structure, and TMR have introduced more datasets for training, so there may be some performance gap. Although LDNet performs slightly lower than TMR on some datasets, it surpasses TMR on the Twitter-2015 dataset. Moreover, LDNet is a universal information extraction solution that covers a broader range of tasks, such as Discontinuous NER and Hyper RE.

Specific Results of Label Drop Mechanism We put the specific results of the label drop mechanism with different drop rates in Table 16. It can be seen that after 100 rounds of fine-tuning, the performance of the two strategies on the CoNLL03, NYT, and 14-res datasets is not much different for LDNet, and in some drop rate cases, LDNet can perform better without the instruction, demonstrating the effectiveness of the label drop mechanism in the absence of specified instructions, which can be extended to other datasets without annotated instructions.

We also conduct ablation experiments on multi-modal datasets, and the experiments on multi-modal datasets are trained for 20 rounds, just like the main experiments. The results we release are the w/ PT & Inst. results. From Table 17 and Figure 3, we can see that the performance with the label drop mechanism is better than without it, which demonstrates the effectiveness of the label drop mechanism in MIE tasks. Under different drop rates, the F1 scores on the MIE datasets do not fluctuate greatly, indicating that the label drop mechanism still has robustness in MIE.