Distantly-Supervised Joint Extraction with Noise-Robust Learning

Anonymous ACL submission

Abstract

Joint entity and relation extraction is a process that identifies entity pairs and their relations using a single model. We focus on the problem of joint extraction in distantly-labeled data, whose labels are generated by aligning entity mentions with corresponding entity and relation tags using a Knowledge Base (KB). One key challenge is the presence of noisy labels arising from both incorrect entity and relation annotations, which significantly impairs the quality of supervised learning. Existing approaches, either considering only one source of noise or making decisions using external knowledge, cannot well-utilize significant information in the training data. We propose DENRL, a generalizable framework that 1) incorporates a lightweight transformer backbone into a sequence labeling scheme for joint tagging, and 2) employs a noise-robust framework that regularizes the tagging model with significant relation patterns and entity-relation dependencies, then iteratively self-adapts to instances with less noise from both sources. Surprisingly, experiments on two benchmark datasets show that DENRL, using merely its own parametric distribution and simple data-driven heuristics, outperforms strong baselines by a large margin with better interpretability.

1 Introduction

Joint extraction aims to detect entities along with their relations using a single model (see Figure 1), which is a critical step in automatic knowledge base construction (Yu et al., 2020). In order to cheaply acquire a large amount of labeled joint training data, distant supervision (DS) (Mintz et al., 2009) was proposed to automatically generate training data by aligning knowledge base (KB) with an unlabeled corpus. It assumes that if an entity pair have a relationship in a KB, all sentences that contain this pair express the corresponding relation.

Nevertheless, DS brings plenty of noisy labels which significantly degrade the performance of the

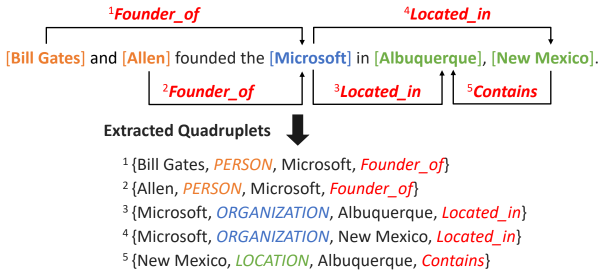


Figure 1: An example of joint extraction on a sentence with multiple relations that share the same entity, e.g., "*Microsoft*" in both the third and the forth relations.

joint extraction models. For example, given a sentence "*Bill Gates lived in Albuquerque*" and the sentence in Figure 1, DS may assign the relation type between "*Bill Gates*" and "*Albuquerque*" as *Place_lived* for both sentences. The words "*lived in*" in the first sentence is the pattern that explains the relation type, thus it is correctly labeled. While the second sentence is noisy due to the lack of corresponding relation pattern. Moreover, due to the ambiguity and limited coverage over entities in open-domain KBs, DS also generates noisy and incomplete entity labels. In some cases, DS may lead to over 30% noisy instances (Mintz et al., 2009), making it impossible to learn useful features.

Previous studies for handling such noisy labels consider either weakly-labeled entities, i.e., distantly-supervised named entity recognition (NER) (Shaalan, 2014), or noisy relation labels, i.e., distantly-supervised relation extraction (RE) (Rink and Harabagiu, 2010), where they focus on designing novel hand-crafted relation features (Yu et al., 2020), neural architectures (Chen et al., 2020), and tagging scheme (Dai et al., 2019) to improve relation extraction performance. Additionally, In-Context Learning (ICL) using external knowledge of Large Language Models (LLMs) (Pang et al., 2023) is popular. However, they are resource demanding, sensitive to prompt design, and may struggle with complex tasks.

To cheaply mitigate both noise sources, we propose **DENRL**—**D**istantly-supervised joint **E**xtraction with **N**oise-**R**obust **L**earning. DENRL assumes that 1) reliable relation labels, whose relation patterns significantly indicate the relationship between entity pairs, should be explained by a model, and 2) reliable relation labels also implicitly indicate reliable entity tags of the corresponding entity pairs. Specifically, DENRL applies *Bag-of-word Regularization* (BR) to guide a model to attend to significant relation patterns which explain correct relation labels, and *Ontology-based Logic Fusion* (OLF) that teaches underlying entity-relation dependencies with Probabilistic Soft Logic (PSL) (Bach et al., 2017). These two information sources are integrated to form a noise-robust loss, which regularizes a tagging model to learn from instances with correct entity and relation labels. Next, if a learned model clearly locates the relation patterns and understands entity-relation logic of candidate instances, they are selected for subsequent adaptive learning. We further sample negative instances that contain corresponding head or tail entities of recognized patterns in those candidates to reduce entity noise. We iteratively learn an interpretable model and select high-quality instances. These two-fold steps are mutually reinforced—a more interpretable model helps select a higher quality subset, and vice versa.

Given the superiority of unified joint extraction methods, we introduce a sequence labeling (Zheng et al., 2017) method to tag entities and their relations simultaneously as token classification. We incorporate a GPT-2 (Radford et al., 2019) backbone that learns rich feature representations into the tagging scheme to benefit the information propagation between relations and entities. The transformer attention mechanism builds direct connection between words and contributes to extracting long-range relations (Li et al., 2022, 2023a). Its multi-head attention weights indicate interactions between each pair of words, which is further leveraged by self-matching to produce position-aware representations. These representations are finally used to decode different tagging results and extract all entities together with their relations.

2 Joint Extraction Architecture

We incorporate a pre-trained GPT-2 backbone into our sequence tagging scheme to jointly extract entities and their relations (see Figure 3).

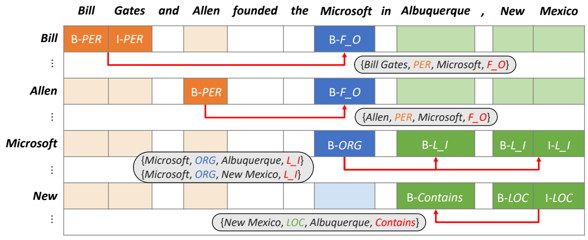


Figure 2: A example of our tagging scheme. For each head entity, we fill a T-tag sequence to represent corresponding relations. *PER*, *ORG*, *LOC* are abbreviations for entity *PERSON*, *ORGANIZATION*, *LOCATION*; *F_O*, *L_I* for relation *Founder_of*, *Located_in*.

2.1 Tagging Scheme

To extract both entities (mention and type) and relations, we tag quadruplets $\{e_1, tag_1, e_2, re\}$ for each start position p and define "BIO" signs to encode positions (see Figure 2). Here, e_1 is the detected entity at p (head entity), tag_1 is the entity type of e_1, e_2 is other detected entity that has relationship with e_1 (tail entity), and re is the predicted relation type between e_1 and e_2. For a T-token sentence, we annotate T different tag sequences according to different start positions.

For each tag sequence, if p is the start of an entity (this sequence is an instance), the entity type is labeled at p, other entities which have relationship to the entity at p are labeled with relation types. The rest of tokens are labeled "O" (Outside), meaning they do not correspond to the head entity. In this way, each tag sequence will produce a relation quadruplet. For example, if p is 7, the head entity is "*Microsoft*" and its tag is *ORG*. Other entities, such as "*Albuquerque*" and "*New Mexico*", are labeled as *L_I* and *L_I* indicating their (unidirectional) relations with "*Microsoft*". If p is 9, the head entity "*Albuquerque*" has no relationship with other entities, thus only the entity type *LOC* is labeled. If p is 13, all tokens are labeled as "O" because there is no entity at the head position to attend to.

We define instances that contain at least one relation as positive instances (e.g., p is 7), and those without relations as negative instances (e.g., p is 9). "BIO" (Begin, Inside, Outside) signs are used to indicate the position information of tokens in each entity for both entity and relation type annotation to extract multi-word entities. Note that we do not need the tail entity type, because every entity will be queried and we are able to obtain all entity types as well as their relations from the T tag sequences.

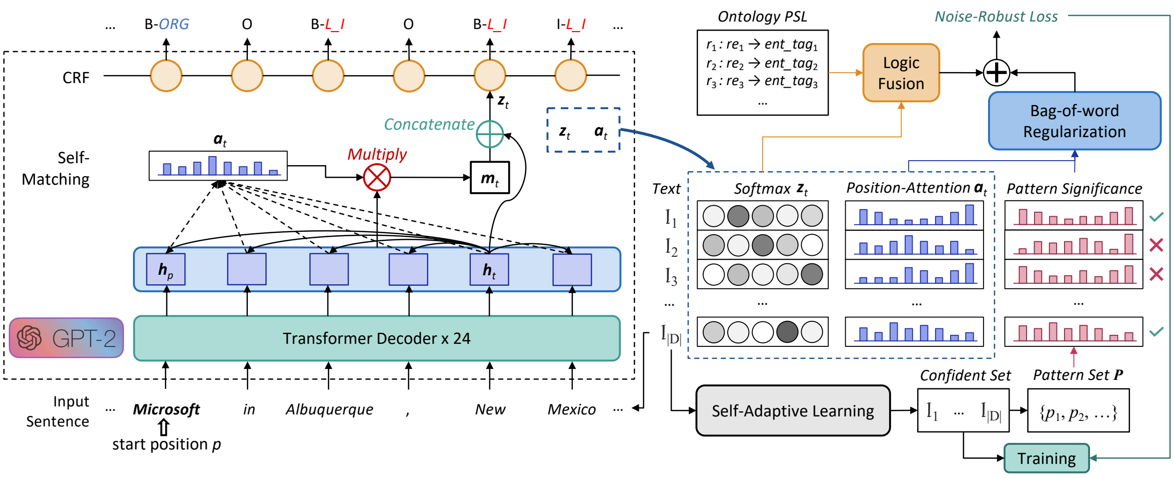


Figure 3: An overview of DENRL framework. The left part is our position-attentive joint tagging model, which receives a sentence input and different start position p to extract all entities and relations. $\boldsymbol{a}_t$ are position-attention weights and $\boldsymbol{z}_t$ are sequence scores. The right part is our noise-robust learning mechanism, which employs BR (on $\boldsymbol{a}_t$) and OLF (on $\boldsymbol{z}_t$) to guide the model to attend to significant patterns and entity-relation dependencies. Then, a fitness score u for each training instance is calculated to select and build new distributed training set as well as confident pattern set. These two steps are run iteratively as self-adaptive learning.

2.2 Tagging Model

GPT-2 with Self-Matching We follow GPT-2 (Radford et al., 2019) to use a multilayer transformer (Vaswani et al., 2017) that takes an input sequence $\mathcal{S} = \{w_1, ...w_T\}$ and converts it into token-level representations $\boldsymbol{h}^0 = \{\boldsymbol{h}_t\}_{t=1}^T$, where $\boldsymbol{h}_t \in \mathbb{R}^d$ is a d-dimensional vector corresponding to the t-th token in $\mathcal{S}$. The model applies L transformer layers over the hidden vectors to produce contextual representations: $\boldsymbol{h}^l = \textsc{Transformer}^{(l)}(\boldsymbol{h}^{l-1}), l \in [1, L]$. Each layer contains a Multi-Head Self-Attention (MHSA) layer followed by a Feed-Forward Network (FFN) over previous hidden state $\boldsymbol{h}^{l-1}$. The final representations $\boldsymbol{h}^L \in \mathbb{R}^{T \times d}$ integrate the contextual information of all previous tokens but are inadequate for decoding a T-tag sequence, since for each position p we still need to encode e_1 and its overlapping relations re with other entities e_2.

We define Self-Matching (Tan et al., 2018) that calculates position-attention $\boldsymbol{a}_t$ between tokens at start position p as well as each target position t:

$$\begin{aligned} \boldsymbol{a}_t &= \mathrm{softmax}(\{a_j^t\}_{j=1}^T) \\ s.t.\ \ a_j^t &= \boldsymbol{w}^\top(\boldsymbol{h}_p^L + \boldsymbol{h}_t^L + \boldsymbol{h}_j^L) \end{aligned} \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^d$ is a parameter to be learned, $\boldsymbol{h}_p$, $\boldsymbol{h}_t$, $\boldsymbol{h}_j \in \mathbb{R}^d$ are hidden states at position p, t, j, respectively. a_j^t is the score computed by comparing $\boldsymbol{h}_p$ and $\boldsymbol{h}_t$ with each hidden state $\boldsymbol{h}_j$. $\boldsymbol{a}_t \in \mathbb{R}^T$ is the softmax attention produced by normalizing a_j^t. The start hidden state $\boldsymbol{h}_p$ serves as comparing with the sentence representations to encode position information, and $\boldsymbol{h}_t$ matches the sentence representations against itself to collect context information. The position-aware representation $\boldsymbol{m}_t \in \mathbb{R}^{T \times d}$ is an attention-weighted sentence vector:

$$\boldsymbol{m}_t = \boldsymbol{a}_t^\top \boldsymbol{h}^L \tag{2}$$

We concatenate $\boldsymbol{h}_t$ and $\boldsymbol{m}_t$ to generate position-aware and context-aware representations $\{\boldsymbol{x}_t\}_{t=1}^T$:

$$\boldsymbol{x}_t = [\boldsymbol{h}_t; \boldsymbol{m}_t] \tag{3}$$

For each start position, self-matching produces different sentence representations and thus can model different tag sequences of a sentence.

CRF Decoder CRF (Lafferty et al., 2001) considers the correlations between labels in neighborhoods and jointly decodes the best chain of labels, which benefits sequence labeling models. For each position-aware representation $\boldsymbol{x}_t$, the input sequence scores $\boldsymbol{Z} = \{\boldsymbol{z}_t\}_{t=1}^T$ is generated by:

$$\boldsymbol{z}_t = \boldsymbol{W}^x \boldsymbol{x}_t \tag{4}$$

where $\boldsymbol{z}_t \in \mathbb{R}^V$ is tag score of the t-th token, V is the number of distinct tags, and z_t^j is the score of the j-th tag at position t.

For a sequence of labels $\boldsymbol{y} = \{y_1, ..., y_T\}$, the decoding score $score(\boldsymbol{Z}, \boldsymbol{y})$ is the sum of transition score from tag y_t to tag y_{t+1}, plus the input score $z_t^{y_t}$ for each token position t. The conditional probability $p(\boldsymbol{y}|\boldsymbol{Z})$ is the softmax of $score(\boldsymbol{Z}, \boldsymbol{y})$

over all possible label sequences $\boldsymbol{y}'$ for $\boldsymbol{Z}$. We maximize the log-likelihood of correct tag sequences during training:

$$\mathcal{L}_c = \sum\nolimits_i \log p(\boldsymbol{y}|\boldsymbol{Z}) \quad (5)$$

Decoding searches for the tag sequence $\boldsymbol{y}^*$ that maximizes the decoding score. The best tag sequence $\boldsymbol{y}^*$ is computed using the Viterbi algorithm.

3 Noise-Robust Learning

To reduce the impact of noisy labels on tagging performance, we introduce *Bow Regularization* (BR) to attend to confident relation patterns for reducing relation noise and *Ontology-based Logic Fusion* (OLF) to increase entity-relation coherence for reducing entity noise. Finally, we employ *Self-Adaptive Learning* (SAL) to iteratively train on instances that can be explained by the model.

3.1 Bag-of-word Regularization (BR)

Assuming reliable relation patterns are explainable to a model itself, we propose average BoW frequency as an instance-level pattern oracle to guide the model's position-attention for joint tagging. For an input sentence $\mathcal{S}$, an entity pair (e_1, e_2) in $\mathcal{S}$, a relation label re, and a relation pattern p that explains the relation re of e_1 and e_2, we define BoW frequency as the corresponding guidance score $\boldsymbol{a}^p$, i.e., Pattern Significance, conditional on pattern p. Take the relation *Contains* as an example, its BoW is a set of tokens {*"capital", "section", "of", "areas", "in", ...*} which appear in a corresponding pattern set {*"capital of", "section in", "areas of", ...*}. The motivation is to guide the model to explore new high-quality patterns such as "*section of*", "*areas in*", etc. The guidance $\boldsymbol{a}^{\mathcal{I}}$ for an instance $\mathcal{I}$ is the average of $\boldsymbol{a}^p$ regarding all patterns m corresponding to each relation re in $\mathcal{S}$:

$$\begin{aligned}\boldsymbol{a}^p &= \text{softmax}(\{\text{BoW}_t\}_{t=1}^T) \\ \boldsymbol{a}^{\mathcal{I}} &= \text{AvgPooling}\left(\boldsymbol{a}^{p_1}, \cdots, \boldsymbol{a}^{p_{|R_{\mathcal{I}}|}}\right)\end{aligned} \quad (6)$$

where BoW_t represents the BoW frequency of w_t under relation re if w_t belongs to entity words or corresponding relation pattern words, e.g., $f(\text{"of"}|\textit{Contains}) = 2$. $|R_{\mathcal{I}}|$ is the number of distinct relation types in instance $\mathcal{I}$.

We expect a joint tagger to approximate its position-attention $\boldsymbol{a}^{\mathcal{S}}$ to $\boldsymbol{a}^{\mathcal{I}}$, where $\boldsymbol{a}^{\mathcal{S}} =$ AvgPooling $(\boldsymbol{a}_1, \ldots, \boldsymbol{a}_T)$ is the average pooling of model's position-attention $\boldsymbol{a}_t$ defined in Equation (1) for each position j in $\mathcal{S}$. We apply Mean Squared Error (MSE) as the optimized function:

$$\mathcal{L}_{BR} = \text{MSE}(\boldsymbol{a}^{\mathcal{I}}, \boldsymbol{a}^{\mathcal{S}}) = \sum (\boldsymbol{a}^{\mathcal{I}} - \boldsymbol{a}^{\mathcal{S}})^2 \quad (7)$$

3.2 Ontology-Based Logic Fusion (OLF)

Probabilistic Soft Logic (PSL) (Bach et al., 2017) uses soft truth values for predicates in an interval between $[0, 1]$, which represents our token classification probability $p(y_t|w_t)$ as a convex optimization problem. We adapt PSL to entity-relation dependency rules according to data ontology. For example, if the predicted relation type is *Founder_of*, the head entity type is expected to be *PERSON*. Training instances that violate any of these rules are penalized to enhance comprehension of entity-relation coherence. Suppose BR guides a model to recognize confident relations, OLF further helps explore instances with reliable entity labels, especially when no relations exist in them.

Particularly, we define *Logic Distance* based on a model's softmax scores over the head entity given its predicted relation type to measure how severely it violates logic rules. For a training instance, we define an *atom* l as each tag and the *interpretation* $I(l)$ as soft truth value for the atom. For each rule r : RELATION $\rightarrow$ ENTITY, the distance to satisfaction $d_r(I)$ under the interpretation I is:

$$d_r(I) = \max\{0, I(l_{re}) - I(l_{ent})\} \quad (8)$$

PSL determines a rule r as satisfied when the truth value of $I(l_{re}) - I(l_{ent}) \geq 0$. For each instance $\mathcal{I}$, we set l_{ent} as (head) entity type and l_{re} as relation type. This equation indicates that the smaller $I(l_{ent})$ is, the larger penalty it has. We compute the distance to satisfaction for each rule r and use the smallest one as penalty because at least one rule needs to be satisfied.

We learn a distance function $\mathcal{D}(\cdot,\cdot)$ that minimizes all possible PSL rule grounding results, as described in Algorithm 1. $\mathcal{D}(\cdot,\cdot)$ should return 0 if at least one PSL rule is satisfied. The prediction probability $p(y|e_1)$ over head entity e_1 is regarded as the interpretation $I(l_{ent})$ of ground atom l_{ent}, so as $p(y|e_2)$ over tail entity e_2 for $I(l_{re})$ of l_{re}. If no rules is satisfied, the distance is set as 0. We formulate the distance to satisfaction as a regularization term to penalize inconsistent predictions:

Algorithm 1 Logic Distance Calculation $\mathcal{D}$

Input: Softmax $p(y|e_i)$, Prediction $\hat{y}_i$, $i \in \{1, 2\}$, PSL rules $\mathcal{R}$ w.r.t. ontology;

Output: Distance d;

```
Initialize d ← 1; Satisfied ← False;
for each r : l_re → l_ent ∈ R ∧ ŷ_2 == l_re do
    ȳ_1 ← l_ent;
    d' ← max {p(ŷ_2|e_2) − p(ȳ_1|e_1), 0};
    d ← min {d', d};
    Satisfied ← True;
if Satisfied == False then
    d ← 0.
```

$$\mathcal{L}_{OLF} = \sum \mathcal{D}(\mathcal{R}; \{(p(y|e_i), \hat{y}_i)\}) \tag{9}$$

where $p(y|e_i)$ is the softmax probability of $\boldsymbol{z}_{t_i}$ in Equation (4) for position t_i of e_i in $\mathcal{S}$, and $\mathcal{L}_{OLF}$ is the sum of $\mathcal{D}(\cdot, \cdot)$ over all entity-relation pairs (e_1, e_2) in instance $\mathcal{I}$. We finalize a noise-robust loss function by summing up (5), (7) and (9):

$$\mathcal{L} = \mathcal{L}_c + \alpha \mathcal{L}_{BR} + \beta \mathcal{L}_{OLF} \tag{10}$$

where α, β are two balancing hyper-parameters.

3.3 Self-Adaptive Learning (SAL)

Self-adaptive learning aims to iteratively select high-quality instances with informative relation patterns p and entity tags. In each training epoch, more precisely-labeled instance are needed to guide a model to attend to informative evidence for joint extraction. For instance selection, more versatile patterns are required to select trustable data and to discover more confident relation patterns. According to the attention mechanism and entity-relation logic, a trained tagger can tell the importance of each word for identifying the entity pair along with their relationship, and predict reasonable entity-relation label pairs. For an instance $\mathcal{I}$, if 1) the model's attention weights do not match the target attention that explains the relation types in $\mathcal{I}$, or 2) its confidence distribution over entity and relation tags violates the logic dependencies, this instance is likely a false alarm. We add up both BR and OLF loss for an instance $\mathcal{I}$ to measure its *fitness* $u(\mathcal{I})$, i.e., how likely it is correctly labeled:

$$u = \sigma(\mathrm{MSE}(\boldsymbol{a}^{\mathcal{I}}, \boldsymbol{a}^{\mathcal{S}}) - \mathcal{D}(\mathcal{R}; \mathcal{I})) \tag{11}$$

where σ is the sigmoid function that bounds u in the range $[0, 1]$. The higher u is, the more confident an instance $\mathcal{I}$ is. We compute fitness scores for all training instances and select those whose score is larger than a predefined threshold τ.

Because trustable relation labels also indicate trustable entity tags, we further consider *Entity Selection* (ES), i.e., selecting negative instances containing either the head or tail entity corresponding to each relation pattern in the selected positive candidates. Specifically, we consider relation pattern p as the text between two entities in an instance. We build an initial trustable pattern set $\mathcal{P}$ by counting all patterns up and selecting the top 10% frequent patterns for each relation type. Next, we redistribute the training dataset $\mathbf{D}$ based on $\mathcal{P}$, where all positive instances that match patterns in $\mathcal{P}$ as well as negative instances that contain the head entity or tail entity of these patterns are retained to train the model for a few epochs. Finally, we select more reliable instances according to fitness scores over $\mathbf{D}$, from which we extract new trustable patterns to enrich $\mathcal{P}$. These new confident instances are learned in the subsequent iteration. We repeat the above procedure until the validation F1 converges.

4 Experiments

4.1 Datasets and Evaluation

We evaluate the performance of DENRL on two public datasets: (1) **NYT** (Riedel et al., 2010). We use the human-annotated test dataset (Jia et al., 2019) including 1,024 sentences with 3,280 instances and 3,880 quadruplets. The training data is automatically generated by DS (aligning entity pairs from Freebase with handcrafted rules), including 235k sentences with 692k instances and 353k quadruplets. (2) **Wiki-KBP** (Ling and Weld, 2012). Its test set is manually annotated in 2013 KBP slot filling assessment results (Ellis et al., 2013) containing 289 sentences with 919 instances and 1092 quadruplets. The training data is generated by DS (Liu et al., 2017) including 75k sentences with 145k instances and 115k quadruplets.

We evaluate the extracted quadruplets for each sentence in terms of Precision (Prec.), Recall (Rec.), and F1. A quadruplet $\{e_1, tag_1, e_2, re\}$ is marked correct if the relation type re, two entities e_1, e_2, and head entity type tag_1 are all matched. Note that negative quadruplets with "None" relation are also considered for evaluating prediction accuracy. We build a validation set by randomly sampling 10% sentences from the test set.

Method	NYT			Wiki-KBP		
	Prec.	Rec.	F1	Prec.	Rec.	F1
LSTM-CRF (Zheng et al., 2017)	66.73	35.02	45.93	40.14	35.27	37.55
PA-LSTM-CRF (Dai et al., 2019)	37.90	**76.25**	50.63	35.82	45.06	39.91
OneIE (Lin et al., 2020)	52.33	64.40	57.74	36.25	46.51	40.74
PURE (Zhong and Chen, 2021)	53.11	65.84	58.79	38.20	44.89	41.28
CoType (Ren et al., 2017)	51.17	55.92	53.44	35.68	46.39	40.34
CNN+RL (Feng et al., 2018)	40.72	58.39	47.98	36.20	44.57	39.95
ARNOR (Jia et al., 2019)	59.64	60.78	60.20	39.37	47.13	42.90
FAN (Hao et al., 2021)	58.22	64.16	61.05	38.81	47.14	42.57
SENT (Ma et al., 2021)	63.88	62.12	62.99	41.37	46.72	43.88
LLM-ICL (Pang et al., 2023)	61.81	58.79	60.26	40.52	45.60	42.91
DENRL (triplet)	**70.72**$_{\pm 0.49}$	66.49$_{\pm 0.50}$	**68.60**$_{\pm 0.49}$	**42.57**$_{\pm 0.32}$	**50.81**$_{\pm 0.28}$	**46.29**$_{\pm 0.30}$
DENRL	70.02$_{\pm 0.45}$	65.84$_{\pm 0.32}$	67.87$_{\pm 0.38}$	41.89$_{\pm 0.27}$	50.14$_{\pm 0.31}$	45.65$_{\pm 0.29}$

Table 1: Evaluation results on NYT and Wiki-KBP datasets. Baselines include normal RE methods (the 1st part), DS RE methods (the 2nd part), and ICL method (the 3rd part). We run the model 5 times to get the average results.

4.2 Baselines

We compare DENRL with the following baselines:

LSTM-CRF (Zheng et al., 2017) that converts joint extraction to a sequence labeling problem based on a novel tagging scheme.

PA-LSTM-CRF (Dai et al., 2019), which uses sequence tagging to jointly extract entities and overlapping relations.

OneIE (Lin et al., 2020), a table filling approach that uses an RNN table encoder to learn sequence features for NER and a pre-trained BERT sequence encoder to learn table features for RE.

PURE (Zhong and Chen, 2021), a pipeline approach that uses pre-trained BERT entity model to first recognize entities and then employs a relation model to detect underlying relations.

CoType (Ren et al., 2017), a feature-based method that handles noisy labels based on multi-instance learning, assuming at least one mention is correct.

CNN+RL (Feng et al., 2018) that trains an instance selector and a CNN classifier using reinforcement learning.

ARNOR (Jia et al., 2019) which uses attention regularization and bootstrap learning to reduce noise for distantly-supervised RE.

FAN (Hao et al., 2021), an adversarial method including a BERT encoder to reduce noise for distantly-supervised RE.

SENT (Ma et al., 2021), a negative training method that selects complementary labels and re-labels the noisy instances with BERT for distantly-supervised RE.

LLM-ICL (Pang et al., 2023), we follow the basic prompt with two demonstration examples, each as a pair of input text and extracted triplets.

4.3 Implementation Details

For DENRL, we use the *gpt2-medium* as the sentence decoder. For baselines using LSTM, we consider a single layer with a hidden size of 256. For baselines using pre-trained BERT, we use the *bert-large-cased*. For LLM-ICL, we use Llama2-7B (Touvron et al., 2023). We tune hyperparameters on the validation set via grid search. Specifically in regularization training, we find optimal parameters α and β as 1 and 0.5 for our considered datasets. We implement DENRL and all baselines in PyTorch, using the AdamW (Loshchilov and Hutter, 2019) optimizer with a learning rate of 5e-4, a dropout rate of 0.2, and a batch size of 8. For instance selection, an empirical fitness threshold is set to 0.5 with the best validation F1. We take a maximum of 5 new patterns in a loop for each relation type. In the SAL stage, we run 5 epochs in the first loop, and 1 epoch in every rest loop until the validation performance converges.

4.4 Overall Results

As shown in Table 1, DENRL (triplet) denotes ignoring head entity type tag_1 when computing correctness, because all baselines only extract triplets $\{e_1, e_2, re\}$. The results of triplet and quadruplet have little difference, indicating that DENRL predicts precise entity types. DENRL significantly outperforms all baselines in precision and F1 metric. Specifically, it achieves roughly 5∼20% F1 improvement on NYT (3∼6% on Wiki-KBP) over the other denoising methods—CoType, CNN+RL, ARNOR, FAN, SENT. Compared to LSTM-CRF that also trains on selected subsets,

Component	Prec.	Rec.	F1
GPT-2+FC	44.28	72.80	55.07
GPT-2+CRF	45.11	**75.19**	56.40
+IDR	**73.12**	48.86	58.58
+BR	69.24	53.67	60.47
+OLF	71.37	55.80	62.63
+SAL (DENRL)	70.72	66.49	**68.60**

Table 2: Evaluation of components in DENRL. GPT-2+FC and GPT-2+CRF are two backbone models. IDR denotes initial data redistributing using initial pattern set. BR and OLF (in this case) are only for the first loop, SAL stands for self-adaptive learning.

DENRL achieves 31% recall improvements on NYT (15% on Wiki-KBP) with still better precision, suggesting that we explore more diverse entity and relation patterns. Compared to the sequence tagging approach PA-LSTM-CRF, DENRL achieves improvements of 32% in precision and over 18% F1 improvement. DENRL also outperforms baselines using pre-trained transformers (OneIE, PURE, FAN, SENT) or LLMs (LLM-ICL), showing our noise-robust learning effectively reduces the impact of mislabeled instances on joint extraction performance.

4.5 Ablation Study

We investigate the effectiveness of several components of DENRL on NYT dataset, as shown in Table 2. Before noise reduction, we first evaluate the impact of CRF layer by substituting it with a FC layer. We found it improves the final performance by over 1% F1. We then build an initial redistributed dataset (via IDR), which helps joint model earn over 2% improvement in F1 and a sharp 28% precision increase compared to GPT-2+CRF. This suggests the original DS dataset contains plenty of noise, thus a simple filtering method would effectively improve the performance.

However, this initial data induces poor recall performance, which means a large proportion of true positives with long-tail patterns are mistakenly regarded as false negatives. Assuming that some relation patterns in the training data are too rare to guide the model learn to attend them, we employ BR to training and achieves 5% recall increases with a slight decline in precision, inducing another 2% F1 improvement. This shows the effect of guiding the model to understand important feature words for identifying relations.

After we introduce OLF to training, both precision and recall improves about 2%, leading to another 2% F1 improvement, proving that logic rules guide a model to learn the entity-relation dependencies and further reduce entity labeling noise.

After we obtain an initial model trained by BR and OLF, we continue SAL where DENRL collects more confident long-tail patterns to mitigate false negatives and finally achieves 6% F1 improvement.

Method	Prec.	Rec.	F1
w/o ES	67.82	**67.45**	67.63
DENRL	**70.72**	66.49	**68.60**

Table 3: Comparison of Precision, Recall, and F1 after using Entity Selection (ES) during SAL.

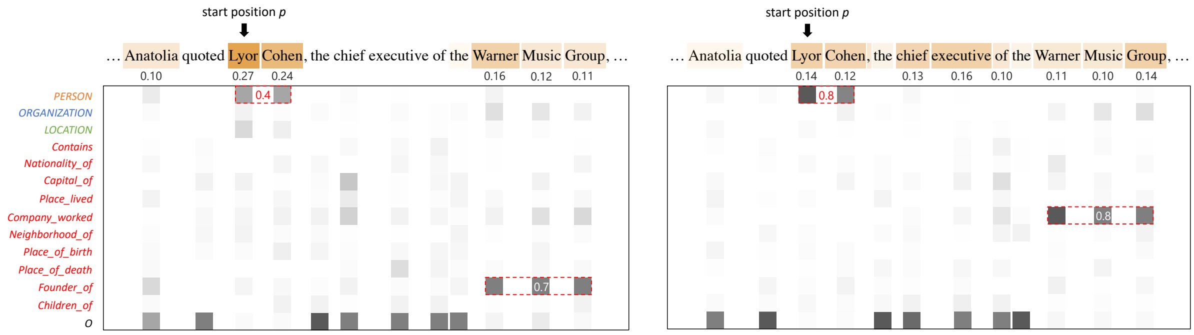
RELATION: *Contains* (left: u, right: pattern)	
0.749	e_2, *section of* e_1
0.692	e_2, *the capital of* e_1
...	...
0.548	e_2, *district of* e_1
0.554	e_2 *and other areas of* e_1
0.539	e_2 *and elsewhere in the* e_1
RELATION: *Company_worked* (left: u, right: pattern)	
0.667	e_1, *the chief executive of* e_1
0.673	e_2 *attorney general,* e_1
...	...
0.595	e_1, *the president of the* e_2
0.513	e_1, *an economist at the* e_2
0.526	e_1, *the chairman and chief executive of* e_2

Table 4: Pattern examples including high-frequency and top long-tail patterns (right) and corresponding average fitness scores (left).

4.6 Interpretability Study

To understand the effect of attention and logic guidance, we select some instances from the test set and visualize their attention weights, as well as the model's softmax probability distribution over all labels. As shown in Figure 4, GPT-2+CRF, which is trained on original noisy data without BR or OLF, only focuses on entity pairs and makes wrong predictions. Its logic distance for r : *Founder_of* $\rightarrow$ *PERSON* is $d_r(I) = \max\{0, 0.7 - 0.4\} = 0.3$. While DENRL precisely captures important words and correctly predicts the relation. The logic distance for r : *Company_worked* $\rightarrow$ *PERSON* is $d_r(I) = \max\{0, 0.8 - 0.8\} = 0 < 0.3$, suggesting the effect of OLF.

To show that BR explores versatile patterns to enrich pattern set $\mathcal{P}$, we summarize both high-frequency patterns obtained by IDR and meaningful long-tail patterns discovered during SAL, and statistic their average fitness (see Table 4). Some long-tail patterns are not similar syntactically but still have over 0.5 average fitness scores, meaning the model learns useful semantic correlations between related feature words.



(a) GPT-2+CRF (b) DENRL

Figure 4: Attention heat maps (top) and softmax probability heat maps (bottom). In this case, e_1: Lyor Cohen, e_2: Warner Music Group, and re: *Company_worked*. GPT-2+CRF misclassifies the relation as *Founder_of*, because it only attends to entities. DENRL is able to locate relation indicators and make correct predictions.

We further check the performance of DENRL on negative test cases that do not contain relations from NYT dataset. After selecting confident candidates in each epoch, we further choose additional trustable negative instances that contain either the head or tail entity corresponding to each relation pattern in the selected positive candidates during bootstrap. We compare the results between methods with and without entity selection, as shown in Table 3. The improved performance with ES demonstrates that a trustable relation pattern also indicates reliable entity labels, and partially explains the overall superiority of DENRL.

5 Related Work

Entities and relations extraction is important to construct a KB. Traditional methods treat this problem as two separated tasks, i.e., NER and RE. Joint extraction detects entities and their relations using a single model which effectively integrates the information of entities and relations, and therefore achieve better results in both subtasks (Zheng et al., 2017). Among them, unified methods tag entities and relation simultaneously, e.g., (Zheng et al., 2017) proposes a novel tagging scheme which converts joint extraction to a sequence labeling problem; (Dai et al., 2019) introduces query position and sequential tagging to extract overlapping relations. These methods avoid producing redundant information compared to the parameter-sharing neural models (Gupta et al., 2016), and require no hand-crafted features that are used in the structured systems (Yu et al., 2020; Ren et al., 2017).

Previous studies on distantly-supervised NER rely on simple tricks such as early stopping (Liang et al., 2020) and multi-type entity labeling (Shang et al., 2018; Meng et al., 2021). For distantly-supervised RE, existing methods include multi-instance learning (Lin et al., 2016) that models noise problem on a bag of instances, reinforcement learning (RL) (Nooralahzadeh et al., 2019; Hu et al., 2021), adversarial (Chen et al., 2021; Hao et al., 2021) or probabilistic learning (Liu et al., 2022; Li et al., 2023b) that selects trustable instances, and pattern-based methods (Ratner et al., 2016; Shang et al., 2022) that directly model the DS labeling process to find noise patterns, e.g., (Feng et al., 2018) proposes a pattern extractor based on RL and use extracted patterns as features for RE.

In recent years, PSL rules have been applied to machine learning topics, including model interpretability (Hu et al., 2016), probability reasoning (Dellert, 2020), sentiment analysis (Gridach, 2020), and temporal relation extraction (Zhou et al., 2021). We are the first to model entity-relation dependencies by designing ontology-based PSL.

6 Conclusions

We propose DENRL, a noise-robust learning framework for distantly-supervised joint extraction, which consists of a transformer backbone, a new loss function and a self-adaptive learning step. Specifically, we use Bag-of-word regularization and logic fusion to learn important relation patterns and entity-relation dependencies. The regularized model is able to select trustable instances and build a versatile relation pattern set. A self-adaptive learning procedure then iteratively improves the model and dynamically maintains trustable pattern set to reduce both entity and relation noise. In the future, we aim to explore more complex patterns when configuring pattern sets. We will also evaluate our framework on other tasks such as event extraction and open information extraction.

Limitations

In this work we incorporate a GPT-2 backbone into a sequence tagging scheme for distantly-supervised joint extraction. While our current framework considers GPT-2, it's designed with flexibility in mind. It can be easily adapted to other transformers such as BERT, XLNet, and even LLMs like Llama2, as the only difference is the computation of the transformer final representations, which is the very first step before our architecture designs.

Though achieving state-of-the-art performance compared to other DS methods, DENRL can be computation-costly due to the position-attentive loss computed on multiple start positions. We further conduct an efficiency analysis in Appendix A, demonstrating a relatively small training overhead of DENRL compared to other DS methods using transformers.

On the other hand, we focus on relations within a sentence and regard words between an entity pair as relation patterns. In our future work, we aim to consider relations beyond the sentence boundary for DS joint extraction to better adapt to real-world information extraction scenarios.

Furthermore, although our OLF is a one-time effort and can benefit future training, it is still handcrafted based on ontology, and we aim to design a probabilistic method such as model uncertainty to quantify more comprehensive underlying relation-entity dependencies in the future.

References

Stephen H. Bach, Matthias Broecheler, Bert Huang, and Lise Getoor. 2017. Hinge-loss markov random fields and probabilistic soft logic. *J. Mach. Learn. Res.*, 18:109:1–109:67.

Miao Chen, Ganhui Lan, Fang Du, and Victor Lobanov. 2020. Joint learning with pre-trained transformer on named entity recognition and relation extraction tasks for clinical analytics. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 234–242, Online. Association for Computational Linguistics.

Tao Chen, Haochen Shi, Liyuan Liu, Siliang Tang, Jian Shao, Zhigang Chen, and Yueting Zhuang. 2021. Empower distantly supervised relation extraction with collaborative adversarial training. In *AAAI 2021*, pages 12675–12682. AAAI Press.

Dai Dai, Xinyan Xiao, Yajuan Lyu, Shan Dou, Qiaoqiao She, and Haifeng Wang. 2019. Joint extraction of entities and overlapping relations using position-attentive sequence labeling. In *AAAI 2019*, pages 6300–6308. AAAI Press.

Johannes Dellert. 2020. Exploring probabilistic soft logic as a framework for integrating top-down and bottom-up processing of language in a task context. *CoRR*, abs/2004.07000.

Joe Ellis, Jeremy Getman, Justin Mott, Xuansong Li, Kira Griffitt, Stephanie M. Strassel, and Jonathan Wright. 2013. Linguistic resources for 2013 knowledge base population evaluations. In *Proceedings of the Sixth Text Analysis Conference, TAC 2013, Gaithersburg, Maryland, USA, November 18-19, 2013*. NIST.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *AAAI 2018*, pages 5779–5786. AAAI Press.

Mourad Gridach. 2020. A framework based on (probabilistic) soft logic and neural network for NLP. *Appl. Soft Comput.*, 93:106232.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2537–2547, Osaka, Japan. The COLING 2016 Organizing Committee.

Kailong Hao, Botao Yu, and Wei Hu. 2021. Knowing false negatives: An adversarial training method for distantly supervised relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9661–9672, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xuming Hu, Chenwei Zhang, Yawen Yang, Xiaohe Li, Li Lin, Lijie Wen, and Philip S. Yu. 2021. Gradient imitation reinforcement learning for low resource relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2737–2746, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Zhiting Hu, Xuezhe Ma, Zhengzhong Liu, Eduard H. Hovy, and Eric P. Xing. 2016. Harnessing deep neural networks with logic rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Wei Jia, Dai Dai, Xinyan Xiao, and Hua Wu. 2019. ARNOR: Attention regularization based noise reduction for distant supervision relation classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1399–1408, Florence, Italy. Association for Computational Linguistics.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Shuyang Li, Yufei Li, Jianmo Ni, and Julian McAuley. 2022. SHARE: a system for hierarchical assistive recipe editing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11077–11090, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yufei Li, Yanchi Liu, Haoyu Wang, Zhengzhang Chen, Wei Cheng, Yuncong Chen, Wenchao Yu, Haifeng Chen, and Cong Liu. 2023a. Glad: Content-aware dynamic graphs for log anomaly detection. *arXiv preprint arXiv:2309.05953*.

Yufei Li, Xiao Yu, Yanchi Liu, Haifeng Chen, and Cong Liu. 2023b. Uncertainty-aware bootstrap learning for joint extraction on distantly-supervised data. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1349–1358, Toronto, Canada. Association for Computational Linguistics.

Chen Liang, Yue Yu, Haoming Jiang, Siawpeng Er, Ruijia Wang, Tuo Zhao, and Chao Zhang. 2020. BOND: bert-assisted open-domain named entity recognition with distant supervision. In *KDD '20: The 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, Virtual Event, CA, USA, August 23-27, 2020*, pages 1054–1064. ACM.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural relation extraction with selective attention over instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133, Berlin, Germany. Association for Computational Linguistics.

Ying Lin, Heng Ji, Fei Huang, and Lingfei Wu. 2020. A joint neural model for information extraction with global features. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7999–8009, Online. Association for Computational Linguistics.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *AAAI 2012*. AAAI Press.

Liyuan Liu, Xiang Ren, Qi Zhu, Shi Zhi, Huan Gui, Heng Ji, and Jiawei Han. 2017. Heterogeneous supervision for relation extraction: A representation learning approach. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 46–56, Copenhagen, Denmark. Association for Computational Linguistics.

Ruri Liu, Shasha Mo, Jianwei Niu, and Shengda Fan. 2022. CETA: A consensus enhanced training approach for denoising in distantly supervised relation extraction. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 2247–2258, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Ruotian Ma, Tao Gui, Linyang Li, Qi Zhang, Xuanjing Huang, and Yaqian Zhou. 2021. SENT: Sentence-level distant relation extraction via negative training. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6201–6213, Online. Association for Computational Linguistics.

Yu Meng, Yunyi Zhang, Jiaxin Huang, Xuan Wang, Yu Zhang, Heng Ji, and Jiawei Han. 2021. Distantly-supervised named entity recognition with noise-robust learning and language model augmented self-training. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 10367–10378. Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Farhad Nooralahzadeh, Jan Tore Lønning, and Lilja Øvrelid. 2019. Reinforcement-based denoising of distantly supervised NER with partial annotation. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP, DeepLo@EMNLP-IJCNLP 2019, Hong Kong, China, November 3, 2019*, pages 225–233. Association for Computational Linguistics.

Chaoxu Pang, Yixuan Cao, Qiang Ding, and Ping Luo. 2023. Guideline learning for in-context information extraction. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15372–15389, Singapore. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Alexander J. Ratner, Christopher De Sa, Sen Wu, Daniel Selsam, and Christopher Ré. 2016. Data programming: Creating large training sets, quickly. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 3567–3575.

Xiang Ren, Zeqiu Wu, Wenqi He, Meng Qu, Clare R. Voss, Heng Ji, Tarek F. Abdelzaher, and Jiawei Han. 2017. Cotype: Joint extraction of typed entities and relations with knowledge bases. In *WWW 2017*, pages 1015–1024. ACM.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III*, volume 6323 of *Lecture Notes in Computer Science*, pages 148–163. Springer.

Bryan Rink and Sanda M. Harabagiu. 2010. UTD: classifying semantic relations by combining lexical and semantic resources. In *Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval@ACL 2010, Uppsala University, Uppsala, Sweden, July 15-16, 2010*, pages 256–259. The Association for Computer Linguistics.

Khaled Shaalan. 2014. A survey of arabic named entity recognition and classification. *Comput. Linguistics*, 40(2):469–510.

Jingbo Shang, Liyuan Liu, Xiaotao Gu, Xiang Ren, Teng Ren, and Jiawei Han. 2018. Learning named entity tagger using domain-specific dictionary. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2054–2064. Association for Computational Linguistics.

Yuming Shang, Heyan Huang, Xin Sun, Wei Wei, and Xian-Ling Mao. 2022. A pattern-aware self-attention network for distant supervised relation extraction. *Inf. Sci.*, 584:269–279.

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. In *AAAI*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Tingwen Liu, Yubin Wang, Bin Wang, and Sujian Li. 2020. Joint extraction of entities and relations based on a novel decomposition strategy. In *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2282–2289. IOS Press.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1227–1236, Vancouver, Canada. Association for Computational Linguistics.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 50–61, Online. Association for Computational Linguistics.

Yichao Zhou, Yu Yan, Rujun Han, J. Harry Caufield, Kai-Wei Chang, Yizhou Sun, Peipei Ping, and Wei Wang. 2021. Clinical temporal relation extraction with probabilistic soft logic regularization and global inference. In *AAAI 2021*, pages 14647–14655. AAAI Press.

A Efficiency Analysis

DENRL considers the position-attentive loss calculated through traversing transformer logits on different start positions. However, it's crucial to underscore that our method does not significantly inflate the training time. For a sentence of n tokens, the computationally-intensive transformer self-attention operations, which typically have an $O(n^2)$ complexity, are executed just once per sentence. The resultant hidden outputs are then used to perform self-matching and CRF decoding regarding each start token, which also has an $O(n^2)$ complexity but with only few extra trainable parameters introduced. This layered approach ensures that the overall computational overhead remains manageable.

Table 5 reports the average GPU hours per training epoch for each method on the NVIDIA A6000 Ada server. We observe that DS methods consume more time compared to their normal counterpart, for example, ARNOR takes up to $\times 1.6$ the overhead of LSTM-CRF. DENRL, although requires more time training the joint model compared to

Method	NYT	Wiki-KBP
BERT+CRF	0.78	0.70
T5+CRF	0.89	0.82
GPT-2+CRF	0.94	0.88
LSTM-CRF	**0.27**	**0.21**
PA-LSTM-CRF	0.35	0.33
OneIE	0.32	0.28
PURE	0.85	0.79
ARNOR	0.43	0.39
FAN	1.62	1.59
SENT	1.43	1.36
DENRL	1.39	1.07

Table 5: Comparison of training overhead (GPU hours) between baselines and SAL training of DENRL with different backbones. **Bold** and underline denote most efficient and time-consuming methods.

GPT-2+CRF, is more efficient than DS methods using transformers (e.g., FAN, SENT).