Combining Supervised Learning and Reinforcement Learning for Multi-Label Classification Tasks with Partial Labels

Zixia Jia[1], **Junpeng Li**[1], **Shichuan Zhang**[2], **Anji Liu**[3], **Zilong Zheng**[1*]
[1] State Key Laboratory of General Artificial Intelligence, BIGAI, Beijing, China
[2] Zhejiang University, Hangzhou, Zhejiang, China [3] UCLA
{jiazixia,zlzheng}@bigai.ai

Abstract

Traditional supervised learning heavily relies on human-annotated datasets, especially in data-hungry neural approaches. However, various tasks, especially multi-label tasks like document-level relation extraction, pose challenges in fully manual annotation due to the specific domain knowledge and large class sets. Therefore, we address the multi-label positive-unlabelled learning (MLPUL) problem, where only a subset of positive classes is annotated. We propose Mixture Learner for Partially Annotated Classification (MLPAC), an RL-based framework combining the exploration ability of reinforcement learning and the exploitation ability of supervised learning. Experimental results across various tasks, including document-level relation extraction, multi-label image classification, and binary PU learning, demonstrate the generalization and effectiveness of our framework.

1 Introduction

Multi-Label Classification (MLC) task treats a problem that allows instances to take multiple labels, and traditional Supervised Learning (SL) methods on MLC heavily rely on human-annotated data sets, especially neural approaches that are data-hungry and susceptible to over-fitting when lacking training data. However, in many MLC tasks that generally have dozens or hundreds of sizes of class sets, incompleteness in the acquired annotations frequently arises owing to the limited availability of expert annotators or the subjective nature inherent in human annotation processes. (Kanehira and Harada, 2016; Cole et al., 2021; Tan et al., 2022; Ben-Baruch et al., 2022). Therefore, we focus on the fundamentally important problem, typically termed Multi-Label Positive-Unlabelled Learning (MLPUL) (Kanehira and Harada, 2016; Teisseyre, 2021), which involves learning from a multi-label dataset in which only *a subset of positive classes* is definitely annotated, while all the remaining classes are unknown (which could be positives or negatives). For instance, as shown in Fig. 1A&B, human annotators find it hard to completely annotate all the relations due to the confusion of understanding relation definitions and long-context semantics in document-level relation extraction (DocRE) task (Huang et al., 2022; Tan et al., 2022).

Positive and unlabelled (PU) classification has received extensive attention in binary settings, with several recent MLPUL approaches adapting traditional binary PU loss functions to address multi-label classification tasks (Kanehira and Harada, 2016; Wang et al., 2022, 2024). These methods typically operate under the assumption that the prior distribution of positive labels can be inferred from fully labeled training samples or closely unbiased estimations. In a specific instance, (Wang et al., 2022) supposed that the actual positive classes are three times the number of observed labels in the DocRE task, and their model's performance is heavily influenced by the prior (Fig. 1C). However, estimating the prior distribution of labels in real-world scenarios poses significant challenges, as it is rarely feasible to ensure a comprehensive data set encompassing all label types (Chen et al., 2020; Hu et al., 2021; Yuan et al., 2023). Additionally, Li et al. (2023) noted that many long-tail label types tend to be omitted from training annotations. Consequently, we focus on addressing MLPUL without prior knowledge of class distribution. Moreover, MLC generally faces the challenge of imbalanced positive and negative labels, which is severely exacerbated by missing positive class annotations under MLPUL, as shown in Fig. 1B. Previous works typically adopted the re-balance factor to re-weight the loss functions, containing positive up-weight and negative under-weight (Li et al., 2020). We simply attempt these approaches and find they partly improve the model performance but still perform

* Correspondence to Zilong Zheng ⟨zlzheng@bigai.ai⟩.

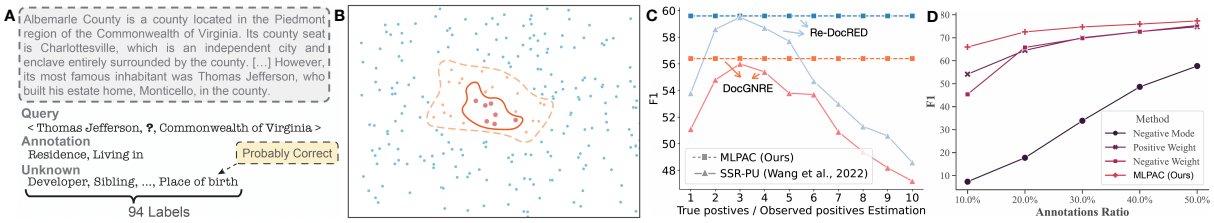

Figure 1: **A.** A partially annotated data sample in DocRE task. **B.** Severe imbalanced distribution of annotated positive (red scatters) and negative labels (blue and orange scatters) corresponding to the DocRED (Yao et al., 2019) dataset. Orange scatters are actually-positive labels reannotated by Re-DocRED dataset (Wu et al., 2022) **C.** Results of training on incomplete DocRED and testing on reannotated Re-DocRED and DocGNRE (Li et al., 2023). SSR-PU is sensitive to prior estimation, while ours is prior agnostic. **D.** Performance comparison in DocRE task.

unsatisfactorily when only a very small set (10%) of positive class annotations is available (Fig. 1D).

Previous works (Silver et al., 2016; Feng et al., 2018; Nooralahzadeh et al., 2019) have demonstrated the powerful exploration ability of Reinforcement Learning (RL). Furthermore, RL has shown great success on distant or partial annotations recently (Feng et al., 2018; Luo et al., 2021; Chen et al., 2023). Inspired by these successful RL attempts, we believe that the exploratory nature of RL has the potential ability to discover additional positive classes while mitigating the overfitting issues typically encountered in supervised learning, especially when the observed label distribution is severely biased, which holds promise in addressing MLPUL. Besides, recent works have shown that supervised learning can be remarkably effective for the RL process (Emmons et al., 2021; Park et al., 2021; Badrinath et al., 2024).

Based on this intuition, we introduce a novel framework termed Mixture Learner for Partially Annotated Classification (MLPAC), which combines the exploratory capacity of RL in tandem with the exploitation capabilities of supervised learning. Specifically, we design a policy network (as a multi-label classifier) and a critic network, along with two types of reward functions: global rewards calculated by a recall function, which evaluates the all-classes prediction performance for each instance, and local rewards provided by the critic network, which assesses the prediction quality of each individual class for a given instance. The local rewards are expected to narrow the exploration space of traditional RL and offer a preliminary yet instructive signal to guide the learning process, while the global rewards encourage the policy network to explore a broader spectrum of positive classes, consequently mitigating distribution bias stemming from imbalanced labels and incomplete annotations.

In addition, inspired by the traditional actor-critic RL algorithm (Bahdanau et al., 2016), we iteratively train the policy network and the critic network, which achieves dynamic reward estimation in our setting. The absence of fully annotated samples in both training and validation sets precludes the direct attainment of perfectly accurate rewards. Hence, we introduce label enhancement through collaborative policy and critic network efforts during iterative training, boosting label confidence and enhancing the critic network's reward estimation accuracy.

Moreover, our RL framework is concise and flexible, guaranteeing its generalization and adaptation to many tasks. Beyond the experiments on document-level relation extraction task (§4.1) in Natural Language Process (NLP) field, we also conduct sufficient experiments in multi-label image classification task (§4.2) in Computer Vision (CV) field and general PU learning setting in binary case (§4.3) to verify the generalization and effectiveness of our framework. All experimental results demonstrate the advantage and significant improvement of our framework.

2 Related Work

Multi-label Positive-Unlabelled Learning Methods Label correlation modeling, rank-based weighted loss function and enhanced incomplete labels are typical technologies that previous MLPUL methods adopted. Label correlation is usually learned or calculated from the label matrix of the training data under partial positive *and* negative samples under *multi-label partially observed labeling* (Rastogi and Mortaza, 2021; Kumar and Rastogi, 2022; Jiang et al., 2023; Yu et al., 2024). In the MLPUL setting, Teisseyre (2021) leveraged classifier chains to model high-order label corre-

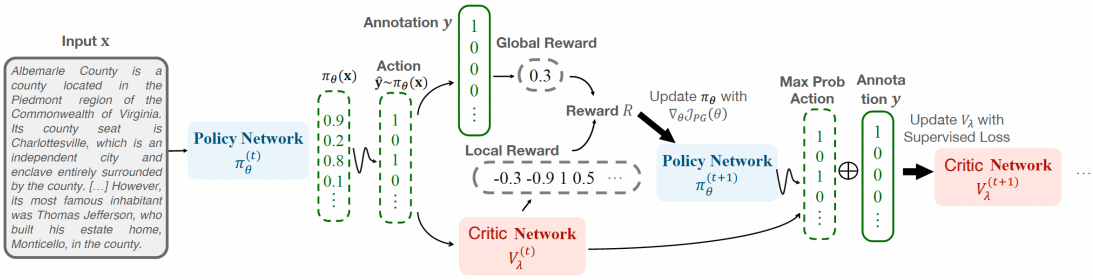

Figure 2: Illustration of our RL framework. $\bigoplus$ represents union operation. We iteratively update the policy network and critic network. The augmented training data are curated for the critic network.

lation with the assumption that prior probabilities could be estimated by predictions from the previous models in the chain. Because label correlations may bring correlation bias under the severe scarcity of positive labels, we did not explicitly model the correlations in our model.

Kanehira and Harada (2016) extended binary PU classification to the multi-label setting, dealing with multi-label PU ranking problem. They modeled MLPUL as cost-sensitive learning by designing a weighted rank loss for multi-label image classification tasks. Following this, Wang et al. (2022) proposed shift and squared ranking loss PU learning for the document-level relation extraction task by modifying the prior terms in the rank-based loss function. Both works supposed the knowledge of class prior distribution, but the prior is difficult to estimate in reality. Recently, encouraged by the binary PU methodology without class prior (Chen et al., 2020; Hu et al., 2021), Yuan et al. (2023) adopted the variational binary PU loss and additionally dealt with the catastrophic imbalanced positive and negative label distribution that MLPUL faced by introducing an adaptive re-balance factor and adaptive temperature coefficient in the loss function. Our approach also pays attention to solving the imbalanced problem in MLPUL and designs a framework without the knowledge of class priors. Different from their methods specifically designed for image classification, our framework is effective for a wide range of MLPUL tasks and is further adapted for imbalanced binary PU learning cases.

Besides the above methodologies, Chen et al. (2020); Yuan et al. (2023) proposed regularization term based on Mixup (Zhang et al., 2018) to enhance incomplete labels, alleviating the overfitting problem and increasing model robustness. The concurrent work to us, Wang et al. (2024) designs positive-augmentation and positive-mixup strategies to improve rank-based learning methods (Wang et al., 2022) for PU document-level relation extraction. These strategies are not uniquely applicable to their framework, and our method can also integrate them, which we leave to future work.

Reinforcement Learning under Weak Supervision There are many previous works leveraging Reinforcement Learning (RL) to solve tasks only with weak supervision (Feng et al., 2018; Zeng et al., 2018; Luo et al., 2021; Chen et al., 2023). In the NLP field, to precisely leverage distant data, Qin et al. (2018); Feng et al. (2018) train an agent as a noisy-sentence filter, taking performance variation on development or probabilities of selected samples as a reward and adopting policy gradient to update. Nooralahzadeh et al. (2019) expand their methods to NER task. Recent work of Chen et al. (2023) also conducts RL to remove the noisy sentence so as to improve the fault diagnosis system. Notably, RL learning on distantly supervised learning aims to filter false positives, whereas our goal is to identify false negatives. A closer related work to us is Luo et al. (2021), in which an RL method is designed to solve the PU learning problem. But unlike us, their agent is a *negative sample selector*, aiming to find negatives with high confidence and then couple them with partial positives to train a classifier. More related works under different weak supervision settings can be found in Appendix A.

3 RL-based Framework

We propose a novel RL framework to solve the MLPUL task. We formulate the multi-label prediction as the action execution in the Markov Decision Process (MDP) (Puterman, 1990). We design both *local* and *global* rewards for actions to guide the action decision process. The policy-based RL method is adopted to train our policy network. The overall RL framework is illustrated in Figure 2. We introduce the details in the following subsections.

3.1 Problem Setting

We mathematically formulate the MLPUL task. Given a multi-label dataset $\mathcal{X} = \{\mathbf{x}_i\}$, each $\mathbf{x}_i$ is labeled with a partially annotated multi-hot vector $\mathbf{y}_i = [y_i^1, \cdots, y_i^c, \cdots y_i^{|\mathcal{C}|}]$, where $y_i^c \in \{0, 1\}$ denotes whether class $c \in \mathcal{C}$ is TRUE for instance $\mathbf{x}_i$ and $|\mathcal{C}|$ is the cardinality of set $\mathcal{C}$. In terms of partial annotation, we assume that $\{c; y_i^c = 1\}$ is a subset of gold positive classes regarding to $\mathbf{x}_i$, *i.e.*, $\{c; y_i^c = 0\}$ is the set of UNKNOWN classes (which could be actually positive or negative). Typically, in the multi-label setting, the size of the label set is dozens or hundreds; thus $|\{c; y_i^c = 1\}| << |\{c; y_i^c = 0\}| \leq |\mathcal{C}|$. In some complicated tasks, such as Relation Extraction, we further define a special label <None> to $\mathbf{x}_i$ if $\mathbf{y}_i = [y_i^c = 0]_{c=1}^{|\mathcal{C}|}$. It should be mentioned that we do not have any fully annotated data, both the training and validation sets being partially annotated.

A straightforward approach (referred to as the "negative mode") for tackling this task involves treating all unlabeled classes as FALSE (set all $y_i^c = 0$ to $y_i^c = -1$) and subsequently reducing the multi-label problem to a number of independent binary classification tasks by employing conventional supervised learning (SL). However, due to incomplete positive labels and severely imbalanced label distribution, the negative mode is susceptible to overfit biased annotated labels, resulting in high precision but low recall on the test set. Based on the negative mode, we introduce an RL framework (MLPAC) that combines RL and SL to mitigate distribution bias and encourage the multi-label classifier to predict more potential positive classes.

3.2 Modeling

Typically, basic RL is modeled as an MDP $(S, A, \pi, \mathcal{T}, R)$ which contains a set of environment and agent states S, a set of actions A of the agent, the transition probabilities $\mathcal{T}$ from a state to another state under action a, and the reward R. The goal of an RL agent is to learn a policy π that maximizes the expected cumulative reward. In our problem setting, we formalize the multi-label positive-unlabelled learning as a **one-step** MDP problem: we do not consider state transitions because our action execution does not change the agent and environment. Our setting highly resembles the setting of contextual bandits (Chu et al., 2011), where actions only affect the reward but not the state. Our RL framework's policy π_θ is a multi-label classifier constructed by a neural network with parameter θ. We define the constituents in detail.

States A state s includes the potential information of an instance to be labeled. In our setting, this information consists of instance features, which are essentially continuous real-valued vectors derived from a neural network.

Actions Due to the multi-label setting, our agent is required to determine the label of each class c for one instance. There are two actions for our agent: setting the current class as TRUE ($\hat{y}_i^c = 1$) or FALSE ($\hat{y}_i^c = -1$). It is necessary to execute $|\mathcal{C}|$ (size of the class set) actions to label an instance completely.

Policy Our policy network outputs the probability $\pi_\theta(\hat{y}_i^c|\mathbf{x}_i) = P(a = \hat{y}_i^c|s = \mathbf{x}_i)$ for each action condition on the current state. We adopt the model structure commonly utilized in previous supervised studies as the architecture for our policy network.

Rewards Recall that our primary objective is to acquire a less biased label distribution compared to the supervised negative mode training approach using the partially annotated training dataset. We anticipate that our MLPAC possesses the capacity for balanced consideration of both *exploitation* and *exploration*. *Exploitation* ensures that our agent avoids straying from local optima direction and avoids engaging in excessively invalid exploratory behavior, while *exploration* motivates our agent to explore the action space somewhat randomly and adapt its policy, preventing overfitting to partial supervision. Inspired by the actor-critic RL algorithm (Bahdanau et al., 2016), we design our rewards function containing two parts: a **local reward** provided by a trainable critic network, which provides immediate value estimation of each action and a **global reward** regarding the overall performance of all classes predictions for each instance.

Specifically, inspired by Luo et al. (2021), the local reward calculates the reward of each class prediction for each instance according to the critic network confidence:

$$r_i^c(V_\lambda, \mathbf{x}_i, c) \tag{1}$$
$$= \begin{cases} \mathbb{C}(-1, \log \frac{p_{V_\lambda}^c(\mathbf{x}_i)}{1-p_{V_\lambda}^c(\mathbf{x}_i)}, 1) & \text{if } \hat{y}_i^c = 1, \\ \mathbb{C}(-1, \log \frac{1-p_{V_\lambda}^c(\mathbf{x}_i)}{p_{V_\lambda}^c(\mathbf{x}_i)}, 1) & \text{if } \hat{y}_i^c = -1. \end{cases}$$

where $p_{V_\lambda}^c(\mathbf{x}_i)$ and $1-p_{V_\lambda}^c(\mathbf{x}_i)$ are the probabilities of class c being TRUE and FALSE respectively for an

instance $\mathbf{x}_i$, calculated by a critic network V_λ with parameter λ, $\hat{y}_i^c$ denotes the prediction of policy network, and $\mathbb{C}(-1, \cdot, 1)$ is a clamping function: a) $\mathbb{C}(-1, x, 1) = -1$ if $x < -1$; b) $\mathbb{C}(-1, x, 1) = 1$ if $x > 1$; c) otherwise, $\mathbb{C}(-1, x, 1) = x$.

We train the critic network in a supervised fashion (under negative mode) to guide the policy network's exploration direction through the local reward function, thus equipping our framework with the ability of *exploitation*. From Equation 2, the action of our policy network could get a positive reward if the critic network has the same prediction tendency and a negative reward otherwise. The policy network is rewarded to "softly fit" the distribution learned by the critic network. To improve the accuracy of value estimation, we perform label enhancement to train our critic network (described in Section 3.4). Consequently, the local rewards offer a preliminary yet instructive signal to guide the learning process in our MLPAC framework, thereby preventing the MLPAC from engaging in excessively invalid exploratory behavior within the large action space, thereby enhancing the overall learning efficiency. Nevertheless, relying solely on these local rewards may potentially lead the MLPAC system to converge to a biased "negative mode" solution. To mitigate this risk, we introduce global rewards to stimulate more comprehensive exploration during the learning process.

As for *global* reward, we employ a straightforward yet highly effective scoring function, which is computed based on the recall metric. In detail, for the whole classes prediction $\hat{\mathbf{y}}_i$ of $\mathbf{x}_i$ with the observed ground truth $\mathbf{y}_i$, the recall score is:

$$recall(\mathbf{y}_i, \hat{\mathbf{y}}_i) = \frac{|\{y_i^c = 1 \wedge \hat{y}_i^c = 1,\ y_i^c \in \mathbf{y}_i, \hat{y}_i^c \in \hat{\mathbf{y}}_i\}|}{|\{y_i^c = 1,\ y_i^c \in \mathbf{y}_i\}|} \tag{2}$$

To enhance recall scores, our policy network is encouraged to predict a greater number of classes as TRUE, thereby alleviating the catastrophic label imbalanced challenge [1].

Note that in our reward design, the terms "local" and "global" are both used to characterize the "goodness" of the predictions of input instances (i.e., state-action pairs). They are both immediate rewards in the considered RL framework, as we formalize MLPUL as a one-step MDP. To calculate the final reward of the whole predictions of an instance, the local rewards of all predicted classes c are summed out, eventually weighted summed with global reward:

$$R(\mathbf{x}_i, \hat{\mathbf{y}}_i, V_\lambda, \mathbf{y}_i) = \frac{1}{|\mathcal{C}|} \sum_{c \in \mathcal{C}} r_i^c(V_\lambda, \mathbf{x}_i, c) + w * recall(\mathbf{y}_i, \hat{\mathbf{y}}_i), \tag{3}$$

where w is a weight controlling the scale balance between local reward and global reward.

3.3 Inference

The final predictions of each instance are decided according to the probabilities that our policy network output. We simply set classes whose probabilities are more than 0.5 ($\pi_\theta(\hat{y}_i^c|\mathbf{x}_i) > 0.5$) to as TRUE (*i.e.*, $\hat{y}_i^c = 1$).

3.4 Learning

We iteratively train our critic network and policy network in an end-to-end fashion. Since the critic network plays a critical role in guiding policy network learning, we employ label enhancement techniques during the training of the critic network to enhance the precision of value estimations. It is important to emphasize that we intentionally exclude the enhanced labels from participation in the calculation of the recall reward. This decision is motivated by the desire to maintain the precision of the global reward and prevent potential noise introduced by the enhanced labels.

It is widely acknowledged that the training process in RL can experience slow convergence when confronted with a vast exploration space. Inspired by previous RL-related works (Silver et al., 2016; Qin et al., 2018), we initiate our process by conducting pre-training for both our policy and critic networks before proceeding with the RL phase. Typically, pre-training is executed through a supervised method. In our settings, a range of trivial solutions for MLPUL can serve as suitable candidates for pre-training. In most cases, we simply utilize the negative mode for pre-training. However, as previously mentioned, the negative mode tends to acquire a biased label distribution due to severe label imbalance. Thus, we implement an early-stopping strategy during the pre-training phase to prevent convergence. The following introduces the detailed learning strategies and objectives.

Objective for Value Model Generally, a well-designed supervised objective urges models to

[1] Note that we do not punish the action of wrongly setting a class as TRUE. Thus, the policy network is pleased to predict all classes as TRUE if we only leverage the global reward function

Algorithm 1: Partially Annotated Policy Gradient Algorithm

Input: Observed data $\mathcal{X}$, partial labels $\mathcal{Y}$, pre-trained policy network π_{θ^0}, critic network V_λ, REINFORCE learning rate α, confidence threshold γ, sample steps T

Output: Optimal parameters θ^*

$e \leftarrow 0$, $\theta^* \leftarrow \theta^0$, enhanced annotation set $\bar{\mathcal{Y}} \leftarrow \mathcal{Y}$

while $e <$ *total training epoches* **do**

  Training set for critic network: $(\mathcal{X}, \bar{\mathcal{Y}})$

  Training set for policy network: $(\mathcal{X}, \mathcal{Y})$

  for $\mathcal{X}_{batch} \in \mathcal{X}$ *in total batches* **do**

    Update λ by minimizing Equation 4 with $\bar{\mathcal{Y}}_{batch}$ for critic network

    for step $t <$ sample steps T **do**

      For each $\mathbf{x}_i \in \mathcal{X}_{batch}$, sample $\hat{\mathbf{y}}_i$ w.r.t. $\hat{\mathbf{y}}_i \sim \pi_\theta(\hat{\mathbf{y}}_i|\mathbf{x}_i)$

      Compute $R(\mathbf{x}_i, \hat{\mathbf{y}}_i, V_\lambda, \mathbf{y}_i)$ according to Equation 3

    Update policy network using $\theta \leftarrow \theta + \alpha \nabla_\theta \mathcal{J}_{PG}(\theta)$

  $\bar{\mathcal{Y}} \leftarrow \{[\bar{y}_i^c]_{c=1}^{|\mathcal{C}|}\}$, where $\bar{y}_i^c = 1$ if ($y_i^c = 1$ or $\big(\pi_\theta(\hat{y}_i^c = 1|\mathbf{x}_i) > \gamma$ and $\hat{y}_i^c = V_\lambda(\mathbf{x}_i)_c = 1\big)$)

  if *eval*(π_θ) $>$ *eval*(π_{θ^*}) **then**

    $\theta^* \leftarrow \theta$

  $e \leftarrow e + 1$

return θ^*

learn expected outputs by learning from annotated data. This process typically refers to the *exploitation*, where the supervised model fits the distribution of label annotations. We denote the supervised objective by a general formulation:

$$\mathcal{L}_{SUP}(\theta) = \sum\nolimits_{\mathbf{x}_i \in \mathcal{X}} p(\mathbf{x}_i) \mathcal{D}(\mathbf{y}_i, \hat{\mathbf{y}}_i), \quad (4)$$

where $\mathcal{D}$ is a task-specific distance metric measuring the distance between annotation $\mathbf{y}_i$ and prediction $\hat{\mathbf{y}}_i$. Recall that we treat all the unknown classes as negatives to perform supervised learning.

Objective for Policy Model As stated in previous work (Qin et al., 2018), policy-based RL is more effective than value-based RL in classification tasks because the stochastic policies of the policy network are capable of preventing the agent from getting stuck in an intermediate state. We leverage policy-based optimization for RL training. The objective is to maximize the expected reward:

$$\begin{aligned} \mathcal{J}_{PG}(\theta) &= \mathbb{E}_{\pi_\theta}[R(\theta)] \quad (5) \\ &\approx \sum_{\mathbf{x}_i \in batch} p(\mathbf{x}_i) \sum_{\hat{\mathbf{y}}_i \sim \pi_\theta(\hat{\mathbf{y}}_i|\mathbf{x}_i)} \pi_\theta(\hat{\mathbf{y}}_i|\mathbf{x}_i) R(\hat{\mathbf{y}}_i.\mathbf{x}_i), \end{aligned}$$

The policy network π_θ can be optimized w.r.t. the policy gradient REINFORCE algorithm (Williams, 1992), where the gradient is computed by

$$\begin{aligned} \nabla_\theta \mathcal{J}_{PG}(\theta) &= \sum_{\mathbf{x}_i \in batch} p(\mathbf{x}_i) \quad (6) \\ &\sum_{\hat{\mathbf{y}}_i \sim \pi_\theta(\hat{\mathbf{y}}_i|\mathbf{x}_i)} \nabla_\theta \ln(\pi_\theta(\hat{\mathbf{y}}_i|\mathbf{x}_i)) R(\hat{\mathbf{y}}_i, \mathbf{x}_i), \end{aligned}$$

where $p(\mathbf{x}_i)$ is a prior distribution of input data. Specific to uniform distribution, $p(\mathbf{x}) = \frac{1}{|\mathbf{x}_{batch}|}$.

Overall Training Procedure The overall training process is demonstrated in Algorithm 1, where $\hat{y}_i^c = V_\lambda(\mathbf{x}_i)_c = 1$ refers to the prediction of class c that critic network outputs for the sample $\mathbf{x}_i$ being TRUE ($p_{V_\lambda}^c(\mathbf{x}_i) > 0.5$). There are several strategies that need to be clarified in the overall training procedure. We empirically prove the effectiveness of these strategies.

▷ The Computation of local rewards is based on the annotated positive classes and a randomly selected subset of unknown classes rather than considering the entire class set of an instance, which is intended to emphasize the impact of positive classes within the computation of local rewards.

▷ The enhanced labels are determined by both the policy network and critic network to guarantee high confidence of enhanced positive labels.

▷ We fix the critic network once it converges to enhance training efficiency.

4 Experiments

The Multi-Label Positive Unlabelled Learning (MLPUL) problem is common and essential in the NLP field. There are many tasks in NLP that face incomplete annotation problems, such as fine-grained entity typing, multi-label text classification, and document-level relation extraction. To verify the effectiveness of our proposed RL framework, we experiment with the positive-unlabelled document-level relation extraction (DocRE) task as a representative MLPUL task in NLP. We also conduct experiments in multi-label image classification (MLIC) tasks and *binary* PU learning setting

Method	10%			30%			50%			70%			100%		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1	P	R	F1
SSR-PU (fix-prior)	75.5	35.8	48.6	52.4	69.9	59.9	33.6	83.0	47.8	23.3	87.2	36.8	-	-	-
SSR-PU (vary-prior)	79.6	33.1	46.7	82.1	56.1	66.7	83.1	67.0	74.2	83.0	71.6	76.9	-	-	78.9
Negative Mode	89.8	3.8	7.3	92.0	20.8	33.8	91.8	42.1	57.7	89.6	58.8	70.6	86.0	72.4	78.6
Pos Weight	84.9	39.8	54.1	85.5	59.3	70.0	85.0	66.8	74.8	83.4	72.5	77.6	82.9	77.3	80.0
Neg Weight	86.7	30.7	45.4	85.0	59.3	69.8	84.1	68.2	75.3	82.8	72.8	77.5	79.8	78.7	79.3
MLPAC (Ours)	58.5	77.0	66.0	83.5	67.71	74.7	81.4	73.6	77.3	83.3	73.9	78.3	80.9	80.8	80.9
P3M (fix-prior)	81.8	50.6	62.5	74.3	76.4	75.3	67.6	84.0	75.0	62.0	87.6	72.6	-	-	-
P3M (vary-prior)	76.6	59.2	66.8	73.2	77.2	75.1	70.5	82.4	76.0	69.4	84.3	76.2	-	-	80.0

Table 1: Results on Re-DocRED with varying ratios of positive class annotations. **Fix-prior** means that we keep the same "true positives/observed positives" prior (=3) in their methods under different ratios, while **vary-prior** means that we set the actual prior corresponding to the rations. The concurrent method **P3M** is shown for reference.

to verify the generalization and effectiveness of our framework.

Besides comparing with previous state-of-the-art (SOTA) models in each specific task, we construct some simple baseline methods:

- **Negative Mode**: As mentioned in Section 3.1, all unknown labels are treated as negative labels, performing conditional supervised learning.
- **Pos Weight**: Based on negative mode, we up-weight positive labels' loss in the supervised loss.
- **Neg Weight**: Based on negative mode, we perform negative sampling in the supervised loss.

We train the model on 1 NVIDIA A100 GPU. The total number of training epochs is 30. We iteratively train our critic and policy network for the first 10 epochs, and then we only train the policy network for the last 20 epochs. The threshold γ of choosing enhancement labels is 0.95 in most cases, and we find our framework performs robustly to γ varying from 0.5 to 0.95. We tune the hyper-parameter reward weight w in Eq.3 and sampling number T with different experiment settings, and w is dynamically adjusted during training[2]. All the above hyper-parameters are determined according to validation set performance. We leverage the F1 score and MAP score (for MLIC) to evaluate all the models. We conduct experiments on selected datasets with varying ratios of positive class annotations from 10% to 100%. We randomly keep a ratio of annotated relations and treat all the leaving classes as unknown. Part of the results are shown in the experimental tables. Detailed descriptions of these simple baselines, evaluation metrics, data statistics, and full experimental results can be found in Appendix B, C, D, and E respectively.

[2]Intuitively, the w of recall reward should be dropped along with the training epochs because our critic network provides more and more accurate local rewards beneficial by data enhancement before convergence.

4.1 Document-level Relation Extraction

Document-level Relation Extraction (DocRE) is a task that focuses on extracting fine-grained relations between entity pairs within a lengthy context. Align to our formulation, an input $\mathbf{x}_i$ is an entity pair, and $\mathbf{y}_i$ represents relation types between the entity pair. An entity pair may have multiple relations or have no relation in DocRE. Beyond the experiments trained on an incomplete annotated DocRED training set and tested on an almost fully annotated Re-DocRED test set (as shown in Fig. 1C), we choose the Re-DocRED, which is the most complete annotated dataset in DocRE, for experiments on varying ratio of positive annotations. The size of class set $\mathcal{C}$ is 97 (contains `<None>`).

Configuration and Baselines We adopt the fully supervised SOTA, DREEAM (Ma et al., 2023), as our critic and policy network architectures in this experiment. We keep the same training method with an Adaptive Thresholding Loss (ATL) of DREEAM for our critic network. The hyper-parameter w of reward weight in Eq. (3) is set to 10. The sample steps T in RL is set to 10. We compare our method to **SSR-PU** (Wang et al., 2022) and show the results of concurrent work **PM3** (Wang et al., 2024) for reference. Both of these two models perform rank-based PU loss with an assumption of label distribution prior.

Results Experimental results in Re-DocRED are shown in Table 1. Compared to previous work and our simple baselines, our **MLPAC** demonstrates its advantage in all annotation ratios. Besides, our framework achieves more balanced precision-recall scores, suggesting its ability to deal with imbalanced label challenges and predict more positive

Method	30%		50%		70%	
	F1	mAP	F1	mAP	F1	mAP
ERP	-	71.0	-	73.5	-	73.8
ROLE	-	72.4	-	76.6	-	79.5
P-ASL+Negative	52.1	74.6	54.0	76.9	71.9	81.0
P-ASL+Counting	26.4	63.4	53.7	76.1	71.6	80.1
Negative Mode	33.7	64.3	52.9	73.8	72.3	81.2
Pos Weight	73.0	72.7	75.7	76.7	76.0	79.9
Neg Weight	68.7	74.8	75.9	78.0	77.9	79.7
MLPAC (Ours)	77.0	77.5	79.1	80.4	79.0	81.4

Table 2: Experimental Results on COCO datasets with varying ratios of positive classes annotations.

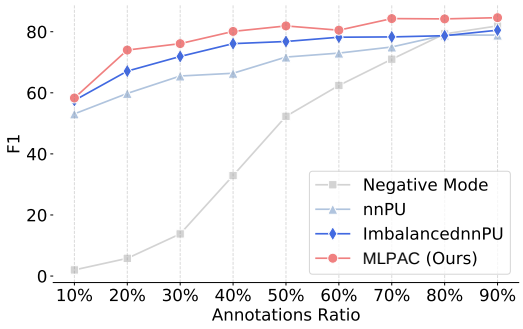

Figure 3: Experimental results of the setting with "truck" category as positives.

Method	Re-DocRED			COCO		
	P	R	F1	P	R	F1
MLPAC (Ours)	64.5	72.8	68.4	80.9	59.2	68.3
w/o. Local reward	12.2	93.3	21.6	61.5	65.5	63.4
w/o. Global reward	84.5	45.9	59.5	89.7	6.9	12.8
w. Prec	85.9	44.0	58.1	96.2	29.8	45.5
w. F1	86.0	43.3	57.6	89.2	46.8	61.4

Table 3: Ablation study on our rewards.

Method	Re-DocRED			COCO		
	P	R	F1	P	R	F1
MLPAC (Ours)	64.5	72.8	68.4	80.9	59.2	68.3
w/o. Iterative training	89.9	34.6	49.9	76.4	33.8	46.9
w/o. Label enhancement	83.6	47.2	60.3	51.7	54.8	53.2
w/o. Action sampling	88.2	36.5	51.7	96.4	20.4	33.7
Supervised self-training	68.0	29.0	40.7	89.7	6.6	12.3

Table 4: Ablation study on our training strategy.

labels. It is worth noting that our framework also achieves improved performance with the full annotated dataset because the full annotations of Re-DocRED still miss some actual relations, as mentioned in Li et al. (2023). Furthermore, our framework does not rely on label prior estimation, while previous rank-based methods are sensitive to a certain extent (fix-prior vs. vary-prior).

4.2 Multi-Label Image Classification

Multi-label image classification is the task of predicting a set of labels corresponding to objects, attributes, or other entities present in an image. Following previous work (Kanehira and Harada, 2016; Ben-Baruch et al., 2022), we utilize MS-COCO dataset (Lin et al., 2014) containing 80 classes.

Configuration and Baselines Our policy and critic networks adopt the same architecture as P-ASL (Ben-Baruch et al., 2022). We rerun previous works (**ERP**, **ROLE**, **P-ASL**) with official code (Cole et al., 2021; Ben-Baruch et al., 2022) for fair comparisons with our methods under the same training data samples. We tune the hyper-parameter w between $\{5, 7, 12\}$ in this task. The sample steps T in RL is set to 50. Detailed descriptions of compared models can be found in Appendix D.

Results Experimental results are shown in Section 4.2. Our **MLPAC** still performs competitively in the MLIC task. Furthermore, we ran our model three times and found very small standard deviations of F1 scores and mAP, which demonstrates the high robustness and stability of our framework. Standard deviations of three runs and more experimental results can be found in Appendix E.2.

4.3 Binary PU Learning Setting

To verify the generalization and wide adaptation of our RL framework, we conduct binary PU learning with the same setting following Su et al. (2021) that concerns positive/negative imbalanced problems in binary image classification. The imbalanced datasets are constructed from CIFAR10[3] by picking only one category as positives and treating all other data as negatives.

Configuration and Baselines Of note, our framework can integrate any supervised model architecture. For a fair comparison, we take the same architecture of Kiryo et al. (2017); Su et al. (2021) as our critic and policy networks. We compared to **nnPU** (Kiryo et al., 2017) and **ImbalancednnPU** (Su et al., 2021). More details are in Appendix D. We tune the hyper-parameter w between $\{10, 20, 50\}$. The action sampling number T is 100.

Results We show F1 scores with varying ratios of annotated positives in Fig. 3. Our **MLPAC** achieves significant improvements over **Negative**

[3]A multi-class dataset containing ten categories. https://web.cs.toronto.edu/

Mode and previous work, It is worth mentioning that our framework still demonstrates its superiority through **ImbalancednnPU**, which is specifically designed for binary PU learning with imbalanced settings.

4.4 Analysis

We conduct ablation studies to analyze our MLPAC framework both on modeling and training strategy.

Rewards Design: To show the effectiveness of combining *exploitation* and *exploration* and the benefit of *local* and *global* rewards, we train our framework in the 10% annotations setting without local rewards and global rewards, respectively. Additionally, we replace the recall scores with precision **w. Prec** or F1 scores **w. F1** as our global rewards to show the effects of different global reward designs. Experimental results are shown in Table 3. It can be observed that it is hard for an RL framework to achieve comparable performance without local rewards to guide exploitation. The reason is that the action space of multi-label classification is too large to find the global optimal directions. Without our global reward, the recall evaluation score drops a lot (72.78 vs. 45.94), which demonstrates the advantage of the global reward in alleviating imbalance distribution. Both the two variants of global reward damage the performance, revealing the advance of taking the exactly accurate evaluation as rewards in the partially annotated setting.

Training Strategy: To verify the effectiveness of our training procedure, we attempt different training strategies shown in Tabel 4. **w/o. Iterative training** means that we fix the critic network after pretraining and only train the policy network in the RL training procedure. **w/o. Data enhancement** means that we still iteratively train our critic and policy network but do not enhance pseudo labels for the critic network. **w/o. Action sampling** means that we leverage the whole action sequence to calculate local rewards without sampling operation illustrated in Section 3.4. **Supervised self-training** means that we conduct self-training of the critic network. It is obvious that our training method makes remarkable achievements. More analysis experiments are in Appendix E.1.

5 Conclusion

In this work, we propose an RL framework to deal with partially annotated multi-label classification tasks. We design local rewards assessed by a critic network and global rewards assessed by recall functions to guide the learning of our policy network, achieving both exploitation and exploration. With an iterative training procedure and a cautious data enhancement, our MLPAC has demonstrated its effectiveness and superiority on different tasks.

Limitation

We have considered the label correlations in our challenge. However, in our setting, label correlations may bring correlation bias to the severe scarcity of positive labels. Therefore, we did not explicitly model the correlations in our current framework. We would like to explore the potential of leveraging the label correlations to enhance our framework in future work.

Acknowledgements

The authors thank the reviewers for their insightful suggestions on improving the manuscript. This work presented herein is supported by the National Natural Science Foundation of China (62376031).

References

Rabab Abdelfattah, Xin Zhang, Mostafa M Fouda, Xiaofeng Wang, and Song Wang. 2022. G2netpl: Generic game-theoretic network for partial-label image classification. *arXiv preprint arXiv:2210.11469*.

Anirudhan Badrinath, Yannis Flet-Berliac, Allen Nie, and Emma Brunskill. 2024. Waypoint transformer: Reinforcement learning via supervised learning with intermediate targets. *Advances in Neural Information Processing Systems*, 36.

Dzmitry Bahdanau, Philemon Brakel, Kelvin Xu, Anirudh Goyal, Ryan Lowe, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2016. An actor-critic algorithm for sequence prediction. In *International Conference on Learning Representations*.

Emanuel Ben-Baruch, Tal Ridnik, Itamar Friedman, Avi Ben-Cohen, Nadav Zamir, Asaf Noy, and Lihi Zelnik-Manor. 2022. Multi-label classification with partial annotations using class-aware selective loss. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4764–4772.

Chong Chen, Tao Wang, Yu Zheng, Ying Liu, Haojia Xie, Jianfeng Deng, and Lianglun Cheng. 2023. Reinforcement learning-based distant supervision relation extraction for fault diagnosis knowledge graph construction under industry 4.0. *Advanced Engineering Informatics*, 55:101900.

Hui Chen, Fangqing Liu, Yin Wang, Liyue Zhao, and Hao Wu. 2020. A variational approach for learning from positive and unlabeled data. *Advances in Neural Information Processing Systems*, 33:14844–14854.

Wei Chu, Lihong Li, Lev Reyzin, and Robert Schapire. 2011. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214. JMLR Workshop and Conference Proceedings.

Elijah Cole, Oisin Mac Aodha, Titouan Lorieul, Pietro Perona, Dan Morris, and Nebojsa Jojic. 2021. Multi-label learning from single positive labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 933–942.

Thibaut Durand, Nazanin Mehrasa, and Greg Mori. 2019. Learning a deep convnet for multi-label classification with partial labels. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 647–657.

Thomas Effland and Michael Collins. 2021. Partially supervised named entity recognition via the expected entity ratio loss. *Transactions of the Association for Computational Linguistics*, 9:1320–1335.

Scott Emmons, Benjamin Eysenbach, Ilya Kostrikov, and Sergey Levine. 2021. Rvs: What is essential for offline rl via supervised learning? In *International Conference on Learning Representations*.

Jun Feng, Minlie Huang, Li Zhao, Yang Yang, and Xiaoyan Zhu. 2018. Reinforcement learning for relation classification from noisy data. In *Proceedings of the aaai conference on artificial intelligence*, volume 32.

Wenpeng Hu, Ran Le, Bing Liu, Feng Ji, Jinwen Ma, Dongyan Zhao, and Rui Yan. 2021. Predictive adversarial learning from positive and unlabeled data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 7806–7814.

Quzhe Huang, Shibo Hao, Yuan Ye, Shengqi Zhu, Yansong Feng, and Dongyan Zhao. 2022. Does recommend-revise produce reliable annotations? an analysis on missing instances in docred. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6241–6252.

Dat Huynh and Ehsan Elhamifar. 2020. Interactive multi-label cnn learning with partial labels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9423–9432.

Qingxia Jiang, Peipei Li, Yuhong Zhang, and Xuegang Hu. 2023. Global and adaptive local label correlation for multi-label learning with missing labels. In *2023 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE.

Warren Jouanneau, Aurélie Bugeau, Marc Palyart, Nicolas Papadakis, and Laurent Vézard. 2023. A patch-based architecture for multi-label classification from single label annotations. In *International Conference on Computer Vision Theory and Applications (VISAPP'23)*.

Atsushi Kanehira and Tatsuya Harada. 2016. Multi-label ranking from positive and unlabeled data. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5138–5146.

Youngwook Kim, Jae Myung Kim, Zeynep Akata, and Jungwoo Lee. 2022. Large loss matters in weakly supervised multi-label classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14156–14165.

Ryuichi Kiryo, Gang Niu, Marthinus C Du Plessis, and Masashi Sugiyama. 2017. Positive-unlabeled learning with non-negative risk estimator. *Advances in neural information processing systems*, 30.

Sanjay Kumar and Reshma Rastogi. 2022. Low rank label subspace transformation for multi-label learning with missing labels. *Information Sciences*, 596:53–72.

Junpeng Li, Zixia Jia, and Zilong Zheng. 2023. Semi-automatic data enhancement for document-level relation extraction with distant supervision from large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5495–5505.

Yangming Li, lemao liu, and Shuming Shi. 2021. Empirical analysis of unlabeled entity problem in named entity recognition. In *International Conference on Learning Representations*.

Yangming Li, Shuming Shi, et al. 2020. Empirical analysis of unlabeled entity problem in named entity recognition. In *International Conference on Learning Representations*.

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

Chuan Luo, Pu Zhao, Chen Chen, Bo Qiao, Chao Du, Hongyu Zhang, Wei Wu, Shaowei Cai, Bing He, Saravanakumar Rajmohan, et al. 2021. Pulns: Positive-unlabeled learning with effective negative sample selector. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8784–8792.

Youmi Ma, An Wang, and Naoaki Okazaki. 2023. Dreeam: Guiding attention with evidence for improving document-level relation extraction. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1963–1975.

Stephen Mayhew, Snigdha Chaturvedi, Chen-Tse Tsai, and Dan Roth. 2019. Named entity recognition with partially annotated training data. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 645–655.

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 2019)*, pages 225–233.

Jongjin Park, Younggyo Seo, Jinwoo Shin, Honglak Lee, Pieter Abbeel, and Kimin Lee. 2021. Surf: Semi-supervised reward learning with data augmentation for feedback-efficient preference-based reinforcement learning. In *Deep RL Workshop NeurIPS 2021*.

Martin L Puterman. 1990. Markov decision processes. *Handbooks in operations research and management science*, 2:331–434.

Pengda Qin, Weiran Xu, and William Yang Wang. 2018. Robust distant supervision relation extraction via deep reinforcement learning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2137–2147.

Reshma Rastogi and Sayed Mortaza. 2021. Multi-label classification with missing labels using label correlation and robust structural learning. *Knowledge-Based Systems*, 229:107336.

Tal Ridnik, Emanuel Ben-Baruch, Nadav Zamir, Asaf Noy, Itamar Friedman, Matan Protter, and Lihi Zelnik-Manor. 2021. Asymmetric loss for multi-label classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 82–91.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. 2016. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489.

Guangxin Su, Weitong Chen, and Miao Xu. 2021. Positive-unlabeled learning from imbalanced data. In *IJCAI*, pages 2995–3001.

Qi Sun, Kun Huang, Xiaocui Yang, Pengfei Hong, Kun Zhang, and Soujanya Poria. 2023. Uncertainty guided label denoising for document-level distant relation extraction. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Qingyu Tan, Lu Xu, Lidong Bing, Hwee Tou Ng, and Sharifah Mahani Aljunied. 2022. Revisiting docred-addressing the false negative problem in relation extraction. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8472–8487.

Paweł Teisseyre. 2021. Classifier chains for positive unlabelled multi-label learning. *Knowledge-Based Systems*, 213:106709.

Ye Wang, Xinxin Liu, Wenxin Hu, and Tao Zhang. 2022. A unified positive-unlabeled learning framework for document-level relation extraction with different levels of labeling. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4123–4135.

Ye Wang, Huazheng Pan, Tao Zhang, Wen Wu, and Wenxin Hu. 2024. A positive-unlabeled metric learning framework for document-level relation extraction with incomplete labeling. *AAAI*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256.

Dong-Dong Wu, Deng-Bao Wang, and Min-Ling Zhang. 2022. Revisiting consistency regularization for deep partial label learning. In *International Conference on Machine Learning*, pages 24212–24225. PMLR.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. DocRED: A large-scale document-level relation extraction dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 764–777, Florence, Italy. Association for Computational Linguistics.

Hai Ye and Zhunchen Luo. 2020. Deep ranking based cost-sensitive multi-label learning for distant supervision relation extraction. *Information Processing & Management*, 57(6):102096.

Yue Yu, Zhengjuan Zhou, Xianju Zheng, Jianping Gou, Weihua Ou, and Fei Yuan. 2024. Enhancing label correlations in multi-label classification through global-local label specific feature learning to fill missing labels. *Computers and Electrical Engineering*, 113:109037.

Zhixiang Yuan, Kaixin Zhang, and Tao Huang. 2023. Positive label is all you need for multi-label classification. *arXiv preprint arXiv:2306.16016*.

Daojian Zeng, Jianling Zhu, Hongting Chen, Jianhua Dai, and Lincheng Jiang. 2024. Document-level denoising relation extraction with false-negative mining and reinforced positive-class knowledge distillation. *Information Processing & Management*, 61(1):103533.

Xiangrong Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Large scaled relation extraction with reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2018. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*.

Xin Zhang, Rabab Abdelfattah, Yuqi Song, and Xiaofeng Wang. 2022. An effective approach for multi-label classification with missing labels. In *2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, pages 1713–1720. IEEE.

Kang Zhou, Yuepei Li, and Qi Li. 2022. Distantly supervised named entity recognition via confidence-based multi-class positive and unlabeled learning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7198–7211, Dublin, Ireland. Association for Computational Linguistics.

A Related Work

Different Settings under Weak Supervision Many previous works, such as partial conditional random field, focus on single-label multi-class tasks with partial supervision (Mayhew et al., 2019; Effland and Collins, 2021; Li et al., 2021; Zhou et al., 2022). Recently, there have been various settings of multi-label classification tasks without fully annotated training sets: *Multi-label positive unlabelled learning*, which is our concern, supposing a multilabel dataset has properties by which (1) assigned labels are definitely positive and (2) some labels are absent but are still considered positive (Kanehira and Harada, 2016; Wang et al., 2022; Yuan et al., 2023); *Partially observed labeling*, which is also termed as *Multi-label classification with missing labels* supposes partial positive *and* negative classes are labeled (Durand et al., 2019; Zhang et al., 2022; Ben-Baruch et al., 2022; Abdelfattah et al., 2022); *Partially positive observed labeling*, which supposes only part of positive classes are observed *with the assumption* that at least one positive label per instance should be observed. *Single positive labeling*, which supposes one and *only* one positive class per instance is observed (Cole et al., 2021; Kim et al., 2022; Jouanneau et al., 2023). *Distantly supervised learning*, which supposes annotated samples contain both false positives and false negatives, devoted to dealing with label noise problems (Ye and Luo, 2020; Sun et al., 2023; Zeng et al., 2024).

B More technology details

B.1 Pos Weight and Neg Weight

In the "Pos Weight" method, we impose a large weight w_p to the positives. We set w_p as the times of spositive targets to unlabeled targets in each training batch. Previous study (Li et al., 2020) had stated that negative sampling can be considered as a type of negative weighting method. And this work experimentally find that negative sampling even work better. In our experiments, we undersampling the unlabeled targets as the "Neg Weight" method. Unlabeled targets 10 times the number of positive targets are retained in each training batch.

B.2 Evaluation Metrics

We compute the F1 scores based on TP (True Positive), FP (False Positive), and FN (False Negative).

$$\begin{aligned} &Recall = TP/(TP+FN), \\ &Precision = TP/(TP+FP), \\ &F1 = \frac{2 * Recall * Precision}{Recall + Precision}. \end{aligned} \tag{7}$$

We choose the widely used evaluation metric mAP on multi-label image classification. N_c is the number of images containing class c, Precision(k, c) is the precision for class c when retrieving k best predictions and rel(k, c) is the relevance indicator function that is 1 if the class c is in the ground-truth of the image at rank k. We also compute the performance across all classes using mean average precision (mAP), where C is the number of classes.

$$AP_c = \frac{1}{N_c} \sum_{k=1}^{N} Precision(k,c) * rel(k,c), \tag{8}$$

$$mAP = \frac{1}{C} \sum_{c} AP_c \tag{9}$$

C Data Statistics

Document-level Relation Extraction Given a document $\mathbf{x}$ containing entities $\mathcal{E}_\mathbf{x} = \{e_i\}_{i=1}^{|\mathcal{E}_\mathbf{x}|}$, DocRE aims to predict all possible relations between every entity pair. Each entity $e \in \mathcal{E}_\mathbf{x}$ is mentioned at least once in $\mathbf{x}$. with all its proper-noun mentions denoted as $\mathcal{M}_e = \{m_i\}_{i=1}^{|\mathcal{M}_e|}$. Each entity pair (e_s, e_o) can hold multiple relations, comprising a set $\mathbf{y} = \mathcal{R}_{s,o} \subset \mathcal{R}$, where $\mathcal{R}$ is a pre-defined relation set. We let the set $\mathcal{R}$ include ϵ, which stands for *no-relation*. To better formulate, we denote the target of DocRE for each document as a set of multi-hot vectors representing labels of relation-existing entity pair $\{\mathbf{y}(e_s, e_o) = \mathbf{y}^{so} = [y_1^{so}, ..., y_i^{so}, ..., y_{|\mathcal{R}|}^{so}], s \in \{1, |\mathcal{E}_D|\}, o \in \{1, |\mathcal{E}_D|\}\}$, where $y_i^{so} \in \{0, 1\}$ and $\sum_{i=1}^{|\mathcal{R}|} y_i = |\mathcal{R}_{s,o}|$.

In this task, we have trained our model based on the training set in the Re-DocRED dataset and validated our model by the Re-DocRED test set and DocGNRE test set. There are 3053 documents (including 59359 entities, and 85932 relations) in the Re-DocRED training set and 500 documents (including 9779 entities, and 17448 relations) in the Re-DocRED test set. DocGNRE test set provides a more accurate and complete test set with the addition of 2078 triples than ReDocRED.

To simulate partial annotation, we randomly kept a ratio of annotated relations. As mentioned in the introduction, Re-DocRED still misses some actual relation annotations. Hence, we also conduct experiments on the full training set to compare our framework with previous fully-supervised work. The size of label set $\mathcal{C}$ is 97 (contains <None>) in this task. Supervised learning on DocRE generally faces two challenges: i) lots of actual relation triples are not labeled by annotators because of the long context and fine-grained relation types; ii) the number of *no-relation* entity pairs are far larger than the number of relation-existing entity pairs, which cause an unbalanced problem.

Multi-label Image Classification Following previous work (Kanehira and Harada, 2016; Ben-Baruch et al., 2022), Ben-Baruch et al. (2022) (**P-ASL**), which deals with partial annotations containing both positive and negative classes, we utilize MS-COCO dataset (Lin et al., 2014) containing 80 classes. We keep the original split with 82081 training samples and 40137 test samples. We simulate the partial annotations following the operations in **P-ASL**. But different from them, we only retain the positive classes in their partial annotations and take all the rest of the classes as UNKNOWN. Specifically, We utilize the MS-COCO dataset to simulate the 'Random per annotation' scheme. We omit each annotation no matter the positive or negative label with probability p. With our setting, the retained positive labels are considered as positives and the rest are all unlabeled. We will not directly exploit unavailable negative annotations.

Binary PU Learning Setting With our formulation, an instance $\mathbf{x}_i$ is an image, and the label of an instance in binary classification settings can be denoted as $\mathbf{y}_i = [y_i^1, y_i^2]$ where y_i^1 is the label of positive and y_i^2 is the label of negative. The prediction for each image is conducted by setting y corresponding to the higher score as 1 and the other as 0. Hence, there are 50,000 training data and 10,000 test data as provided by the original CIFAR10. To make the training data into a partially annotated learning problem, we randomly sample a ratio of positives as annotated data and all the leaving training data as an unknown set.

D Detailed Experimental Configuration

Document-level Relation Extraction We randomly sample three different versions of datasets and report the average results over them. Detailed scores are in Table 5 in Appendix E.1. The training hyper-parameters are the same as DREEAM (Ma et al., 2023). We only train the policy network for the last 20 epochs. It takes about 6 hours.

Multi-Label Image Classification For a fair comparison, our critic and policy networks have the same architecture as **P-ASL**. The training hyper-parameters are the same as that in (Ben-Baruch et al., 2022). Due to the different partially annotated settings, we rerun **P-ASL** utilizing their codebase but with our datasets. **P-ASL+Negative** means training a model taking all UNKNOWN as negative classes to predict label distribution as prior. **P-ASL+Counting** means counting partially labeled positive classes as distribution prior. We also rerun **EPR** and **ROLE** methods from (Cole et al., 2021) with our datasets, utilizing their official code. We tune the hyper-parameter w between $\{5, 7, 12\}$ in this task. Following previous work (Ridnik et al., 2021; Huynh and Elhamifar, 2020), we use both F1 scores and mAP as evaluation metrics in this task. Detailed methodology of the re-weight approach and the detailed formula of metric calculations can be found in Appendix B.

Binary PU Learning We consider a 13-layer CNN with ReLU as the backbone and Adam as the optimizer. Kiryo et al. (2017) designed an algorithm **nnPU** for PU learning with balanced binary classification data, while Su et al. (2021) proposed **ImbalancednnPU** considering imbalanced setting. We take these two previous state-of-the-art models as our compared baselines. We rerun **nnPU** and **ImbalancennPU** with their provided codes and configurations and report the results. The negative reward sampling is 20% for all settings. The threshold γ to choose enhancement labels is 0.8. We keep the values of other hyper-parameters the same as Su et al. (2021). Following previous work, we evaluate all the methods with F1 scores. Unless stated otherwise, the hyper-parameters specified in

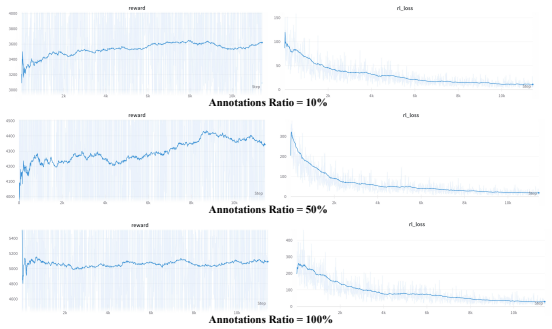

Figure 4: Train Curve.

our framework remain the same in the following experiments.

E More experiments

We indeed considered two different scenarios to evaluate the robustness of our method.

Different dataset versions: To simulate partial annotation, we randomly reserve a ratio of positive classes and treat other classes as unknown. To fully verify the effectiveness and stability of our model, we randomly constructed three versions of data sets (refer to Table 5 for detailed results on DocRE). It can be seen that our method achieves consistent improvement in all the dataset versions. Different runs in the same dataset: For multi-label image classification, we repeat three runs in the same dataset. The standard deviations of F1/MAP scores of Table 10 in the paper are as follows.

E.1 Document-level Relation Extraction

Training curve. In Fig. 4, we display the reward and loss curves of our model in three annotation rations, 10%, 50%, and 100%. Our experimental settings were conducted under partially annotated multi-label tasks, but we also compute metrics on ground Truth during the experiment.

All experiments on different ratios of annotated labels To fully verify the effectiveness and robustness of our model, we randomly constructed three versions of data sets and tested the DREEAM model, Pos Weight, Neg Weight, and our MLPAC model on all data sets respectively. The results are shown in Table 6.

Experiments on selecting action sampling ratios (Take annotations ratio=50% as an example) In order to select the action sampling ratio hyperparameter, we conducted comparative experiments from 0.1 to 0.9, and finally found that the model performed best when the hyperparameter was 0.4. The results are shown in Table 5.

Critic network performance of Our MLPAC We iteratively train our critic network and policy network. After multiple rounds of iterations, the performance of the critic network has been greatly improved. The performance of the critic network of our MLPAC is shown in Table 7.

Case study. In Table 12, we show an example on the prediction of each method. Our MLPAC predicts more true positives.

E.2 Multi-label Image Classification

The experimental results on extra evaluation metrics and annotation ratios. In Table 8 and Table 9, we show Precision and Recall of CIFAR10 and the results of other annotation ratios on Ms-COCO. Table 10 shows the stability of our method MLPAC. The standard deviation was computed from three different runs on the MS-COCO dataset.

The experimental results on synthetic classification. According to the data construction, any category of data in CIFAR10 dataset can be chosen as the positive set. To further make our experiments convincing, we show the results of different data construction in Appendix E.2.

Sampling Ratio	version0			version1			version2			average		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
0.1	71.21	82.27	76.34	77.19	77.3	77.25	81.53	71.14	75.98	76.64	76.9	76.52
0.2	75.22	79.23	77.17	79.72	74.82	77.19	83.26	68.4	75.14	79.4	74.15	76.5
0.3	76.79	77.88	77.33	80.57	73.6	76.93	84.88	66.84	74.79	80.75	72.77	76.35
0.4	78.26	77.18	77.72	81.1	73.58	77.16	83.94	67.4	74.77	81.1	72.72	76.55
0.5	78.35	77.1	77.72	83.26	72.1	77.28	85.23	65.7	74.2	82.28	71.63	76.4
0.6	79.54	76.04	77.75	83.22	71.41	76.86	85.41	65.74	74.29	82.72	71.06	76.3
0.7	80.74	75.35	77.95	83.03	71.12	76.61	86.41	64.4	73.8	83.39	70.29	76.12
0.8	80.43	75.12	77.68	83.92	70.6	76.69	84.65	65.7	73.98	83.0	70.47	76.12
0.9	81.04	74.62	77.7	84.45	69.83	76.45	84.44	65.74	73.92	83.31	70.06	76.02

Table 5: Action Sampling Ratio

Method	Data Ratio	version0			version1			version2			average		
		P	R	F1	P	R	F1	P	R	F1	P	R	F1
DREEAM	**10%**	91.23	4.53	8.64	90.88	3.54	6.82	87.39	3.38	6.5	89.83	3.82	7.32
	20%	90.74	10.0	18.02	92.37	9.65	17.47	90.86	9.8	17.69	91.32	9.82	17.73
	30%	92.45	19.94	32.8	92.48	19.95	32.82	91.19	22.35	35.9	92.04	20.75	33.84
	40%	93.12	28.08	43.15	91.92	34.82	50.51	92.25	36.34	52.14	92.43	33.08	48.6
	50%	92.69	43.63	59.33	91.25	43.28	58.71	91.44	39.29	54.96	91.79	42.07	57.67
	60%	92.16	48.56	63.6	90.49	56.52	69.58	89.52	55.68	68.66	90.72	53.59	67.28
	70%	88.56	60.15	71.64	91.28	55.69	69.17	88.92	60.51	71.01	89.59	58.78	70.61
	80%	87.91	65.18	74.86	89.83	62.75	73.89	86.95	66.49	75.36	88.23	64.81	74.7
	90%	87.49	66.86	75.79	87.61	67.66	76.35	86.4	68.75	76.57	87.17	67.76	76.24
Pos Weight	**10%**	84.43	34.1	48.57	84.61	43.22	57.21	85.8	42.21	56.58	84.95	39.84	54.12
	20%	87.72	47.36	61.51	82.51	57.19	67.56	86.61	51.3	64.44	85.61	51.95	64.5
	30%	83.57	61.65	70.95	87.05	57.04	68.92	85.75	59.23	70.07	85.46	59.31	69.98
	40%	87.51	59.26	70.67	84.29	65.91	73.97	85.65	64.14	73.35	85.82	63.1	72.66
	50%	83.66	68.09	75.08	85.78	66.33	74.81	85.66	65.92	74.5	85.03	66.78	74.8
	60%	84.85	68.57	75.85	85.55	68.09	75.83	84.51	68.87	75.89	84.97	68.51	75.86
	70%	82.77	73.07	77.62	83.0	73.13	77.76	84.37	71.4	77.34	83.38	72.53	77.57
	80%	83.57	73.82	78.39	82.46	75.68	78.93	83.64	73.61	78.31	83.22	74.37	78.54
	90%	83.9	74.54	78.94	82.48	76.44	79.35	82.87	75.8	79.18	83.08	75.59	79.16
Neg Weight	**10%**	88.1	29.67	44.39	86.06	30.1	44.6	86.06	32.37	47.05	86.74	30.71	45.35
	20%	82.94	55.7	66.64	83.72	55.24	66.56	85.49	51.25	64.08	84.05	54.06	65.76
	30%	85.9	58.87	69.86	86.47	55.99	67.97	82.7	63.04	71.55	85.02	59.3	69.79
	40%	86.1	62.08	72.14	85.55	62.72	72.37	85.19	64.47	73.39	85.61	63.09	72.63
	50%	84.25	67.83	75.15	84.27	68.5	75.57	83.64	68.37	75.24	84.05	68.23	75.32
	60%	84.25	69.76	76.32	84.39	69.17	76.02	82.92	71.29	76.67	83.85	70.07	76.34
	70%	81.89	73.52	77.48	82.88	73.03	77.64	83.74	71.72	77.27	82.84	72.76	77.46
	80%	80.99	76.24	78.54	82.51	74.83	78.48	81.58	74.93	78.11	81.69	75.33	78.38
	90%	80.85	77.08	78.92	80.7	76.93	78.77	80.92	77.22	79.03	80.82	77.08	78.91
Our MLPAC	**10%**	64.47	72.78	68.37	62.39	74.98	68.11	48.65	83.15	61.39	58.5	76.97	65.96
	20%	82.25	66.75	73.69	86.2	58.91	69.99	81.94	67.33	73.92	83.46	64.33	72.53
	30%	83.71	67.98	75.03	86.03	63.58	73.12	80.87	71.56	75.93	83.54	67.71	74.69
	40%	84.56	68.92	75.94	83.21	70.08	76.08	83.78	69.2	75.8	83.85	69.4	75.94
	50%	81.4	72.86	76.89	80.32	74.34	77.21	82.55	73.62	77.83	81.42	73.61	77.31
	60%	82.3	73.97	77.92	80.4	75.35	77.79	80.61	74.7	77.54	81.1	74.67	77.75
	70%	83.34	73.57	78.15	83.27	73.87	78.29	83.25	74.36	78.55	83.29	73.93	78.33
	80%	81.92	75.65	78.66	81.68	76.02	78.75	62.77	80.57	70.56	75.46	77.41	75.99
	90%	80.83	77.58	79.18	80.41	78.01	79.19	80.97	77.48	79.19	80.74	77.69	79.2

Table 6: Results of DREEAM, Pos Weight, Neg Weight, MLPAC on different ratios of annotated labels

Data Ratio	version0			version1			version2			average		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
10%	60.69	74.91	67.06	57.47	77.17	65.88	45.89	84.41	59.46	54.68	78.83	64.13
20%	81.16	67.2	73.52	86.34	58.12	69.47	83.3	63.88	72.3	83.6	63.07	71.76
30%	83.1	66.99	74.18	83.65	64.82	73.04	80.95	69.74	74.93	82.57	67.18	74.05
40%	85.19	66.35	74.6	83.25	69.41	75.7	83.12	69.13	75.48	83.85	68.3	75.26
50%	80.44	72.78	76.42	78.51	75.1	76.76	83.24	72.89	77.28	80.73	73.59	76.82
60%	83.53	71.8	77.22	79.9	74.35	77.02	80.59	73.6	76.93	81.34	73.25	77.06
70%	86.14	69.18	76.74	87.65	66.63	75.71	85.57	69.9	76.94	86.45	68.57	76.46
80%	85.77	70.19	77.2	83.64	72.85	77.87	79.93	71.16	75.29	83.11	71.4	76.79
90%	83.86	74.78	79.06	82.85	74.42	78.41	83.77	74.42	78.82	83.49	74.54	78.76

Table 7: Critic network performance of Our MLPAC. We construct the training set three times with different random seeds, corresponding to the three versions.

Data Ratio	nnPU			ImbnnPU			Negative Mode			Our MLPAC		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
10%	52.0	39.3	44.8	41.3	59.2	48.6	40.4	3.8	7.0	47.7	53.0	50.2
20%	54.6	42.6	47.9	43.9	66.8	53.0	76.8	9.6	17.1	61.4	68.1	64.6
30%	58.4	42.3	49.1	43.9	66.8	53.0	71.1	18.0	28.7	59.6	75.6	66.6
40%	57.1	45.1	50.4	54.8	73.7	62.8	76.9	24.9	37.6	62.1	74.6	67.8
50%	56.9	49.2	52.8	61.5	69.4	65.2	75.0	45.8	56.9	63.8	76.9	69.8
60%	59.4	49.7	54.1	61.5	68.1	64.6	69.1	46.5	55.6	65.0	77.7	70.8
70%	61.7	51.5	56.1	62.6	67.4	64.9	82.2	52.7	64.2	72.6	77.7	75.1
80%	63.0	52.7	57.4	63.2	72.8	67.7	79.5	64.4	71.2	78.9	73.0	75.8
90%	70.4	47.5	56.7	62.7	77.2	69.2	82.5	69.1	75.2	75.2	78.7	76.9

Table 8: The results of CIFAR10 dataset. We consider the original class 'airplane' as the positive targets.

Data Ratio	Pos Weight			Neg Weight			Negative Mode			Our MLPAC		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
20%	72.8	69.5	71.1	89.7	39.5	54.9	87.9	9.9	17.9	79.4	67.2	72.8
40%	74.3	75.0	74.7	82.5	65.6	73.1	90.3	28.0	42.8	83.0	74.4	78.5
60%	74.6	79.0	76.7	80.1	74.0	76.9	96.0	47.0	63.1	79.4	76.5	77.9
80%	72.5	83.1	77.4	83.1	75.7	79.2	92.8	65.3	76.6	82.4	77.5	79.9

Table 9: The results of other annotation ratios on MS-COCO dataset.

Standard Deviation										
10%		30%		50%		70%		90%	
F1	mAP	F1	mAP	F1	mAP	F1	mAP	F1	mAP
68.3(0.12)	66.6(0.33)	77.0(0.30)	77.5(0.25)	79.1(0.15)	80.4(0.13)	79.0(0.10)	81.4(0.15)	80.5(0.05)	83.4(0.05)

Table 10: The standard deviations $(\cdot)$ of F1 and mAP were computed from three different runs on the MS-COCO dataset.

Method	10%	20%	30%	40%	50%	60%	70%	80%	90%
nnPU	44.8	47.9	49.1	50.4	52.8	54.1	56.1	57.4	56.7
ImbalancednnPU	48.6	53.0	59.1	62.8	65.2	64.6	64.9	67.7	69.2
Negative Mode	7.0	17.1	28.7	37.6	56.9	55.6	64.2	71.2	75.2
MLPAC (Ours)	50.2	64.6	66.6	67.8	69.8	70.8	75.1	75.8	76.9

Table 11: F1 scores with varying ratios of positive annotations. We take images of the “Airplane" category as positives in this table.

Item	Content or Triples
Title	Guido Bonatti
Document	Guido Bonatti (died between 1296 and 1300) was an Italian mathematician, astronomer and astrologer, who was the most celebrated astrologer of the 13th century. Bonatti was advisor of Frederick II, Holy Roman Emperor, Ezzelino da Romano III, Guido Novello da Polenta and Guido I da Montefeltro. He also served the communal governments of Florence, Siena and Forlì. His employers were all Ghibellines (supporters of the Holy Roman Emperor), who were in conflict with the Guelphs (supporters of the Pope), and all were excommunicated at some time or another. Bonatti ’s astrological reputation was also criticised in Dante’s Divine Comedy, where he is depicted as residing in hell as punishment for his astrology. His most famous work was his Liber Astronomiae or ’Book of Astronomy’, written around 1277. This remained a classic astrology textbook for two centuries.
DREEAM	⟨Dante, notable work, Divine Comedy⟩
Pos Weight	⟨Dante, notable work, Divine Comedy⟩ ⟨Divine Comedy, creator, Dante⟩ ⟨Divine Comedy, author, Dante⟩
Neg Weight	⟨Guido Bonatti, notable work, Liber Astronomiae⟩ ⟨Guido Bonatti, notable work, Book of Astronomy⟩ ⟨Dante, notable work, Divine Comedy⟩ ⟨Divine Comedy, author, Dante⟩
Our MLPAC	⟨Guido Bonatti, notable work, Liber Astronomiae⟩ ⟨Guido Bonatti, notable work, Book of Astronomy⟩ ⟨Dante, notable work, Divine Comedy⟩ ⟨Divine Comedy, creator, Dante⟩ ⟨Divine Comedy, author, Dante⟩ ⟨Liber Astronomiae, author, Guido Bonatti⟩
Ground Truth	⟨Guido Bonatti, date of death, 1296⟩ ⟨Guido Bonatti, date of death, 1300⟩ ⟨Divine Comedy, characters, Guido Bonatti⟩ ⟨Divine Comedy, creator, Dante⟩ ⟨Divine Comedy, author, Dante⟩ ⟨Book of Astronomy, author, Guido Bonatti⟩ ⟨Liber Astronomiae, author, Guido Bonatti⟩ ⟨Guido Bonatti, country of citizenship, Italian⟩ ⟨Guido Bonatti, notable work, Liber Astronomiae⟩ ⟨Dante, notable work, Divine Comedy⟩ ⟨Guido Bonatti, present in work, Divine Comedy⟩ ⟨Guido Bonatti, notable work, Book of Astronomy⟩

Table 12: An Example from Re-DocRED