What the Role is vs. What Plays the Role: Semi-Supervised Event Argument Extraction via Dual Question Answering

Yang Zhou[1, 2], **Yubo Chen**[1, 2], **Jun Zhao**[1, 2] **Yin Wu**[3], **Jiexin Xu**[3], **Jinlong Li**[3]
[1] National Laboratory of Pattern Recognition, Institute of Automation,
Chinese Academy of Sciences, Beijing, 100190, China
[2] School of Artificial Intelligence, University of Chinese Academy of Sciences,
Beijing, 100049, China
[3] AI Lab, China Merchant Bank, ShenZhen, 518057, China
{yang.zhou2020, yubo.chen, jzhao}@nlpr.ia.ac.cn
{xyionwu, jiexinx, lucida}@cmbchina.com

Abstract

Event argument extraction is an essential task in event extraction, and become particularly challenging in the case of low-resource scenarios. We solve the issues in existing studies under low-resource situations from two sides. From the perspective of the model, the existing methods always suffer from the concern of insufficient parameter sharing and do not consider the semantics of roles, which is not conducive to dealing with sparse data. And from the perspective of the data, most existing methods focus on data generation and data augmentation. However, these methods rely heavily on external resources, which is more laborious to create than obtain unlabeled data. In this paper, we propose DualQA, a novel framework, which models the event argument extraction task as question answering to alleviate the problem of data sparseness and leverage the duality of event argument recognition which is to ask "*What plays the role*", as well as event role recognition which is to ask "*What the role is*", to mutually improve each other. Experimental results on two datasets prove the effectiveness of our approach, especially in extremely low-resource situations.

Introduction

Extracting events (EE) from natural language text has received growing interest these years (Hirschberg and Manning 2015; Liu, Chen, and Liu 2019; Liu et al. 2019; Tong et al. 2020), which is usually modeled as two-stage task, including event detection (ED) and event argument extraction (EAE). As event detection has gained great popularity and reached a fairly high performance (Wang et al. 2019), event argument extraction becomes the key of event extraction. Based on the trigger and event type detected by ED, the goal of EAE is to extract the arguments related to the event and predict their roles according to the event schema, which defines what kind of roles should be contained in specific event type. For example, giving an *Attack* event triggered by "*destroyed*" as well as its event mention "*He claimed Iraqi troops had destroyed five tanks*", EAE needs to recognize "*Iraqi troops*"(Role= *Attacker*), and "*five tanks*"(Role= *Target*).

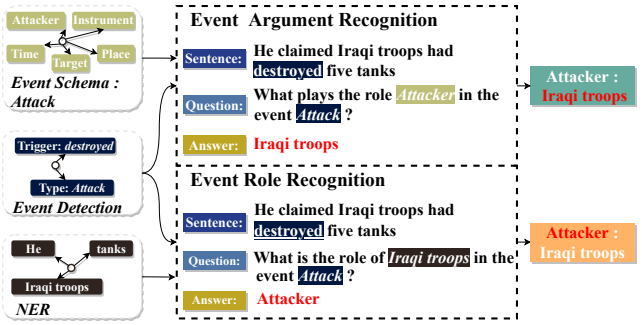

Figure 1: Examples of Event Argument Recognition(EAR) and Event Role Recognition(ERR). EAR depends on the pre-defined event schema "Attack". ERR depends on the Named Entity Recognition(NER) of the sentence. Both of them acted on the premise of event information providing by ED.

So far, many methods have been done under the supervised learning paradigm (Chen et al. 2015; Nguyen, Cho, and Grishman 2016; Liu, Luo, and Huang 2018), and they ordinarily demand quantities of manually annotated data, which is very expensive, and scarce in real situations. According to our statistics, about 60% event types in ACE 2005 English corpus (Doddington et al. 2004) have less than 100 labeled samples and only 1.11% events in ACE 2005 have all roles that the type should contain. Thus, how to address the bottleneck of low-resources EAE has become a challenge. We try to overcome this challenge from both **model** and **data** perspectives.

From the perspective of the model, existing EAE methods usually adopt sequence-labeling paradigm or classification paradigm. (Chen et al. 2015; Nguyen, Cho, and Grishman 2016; Liu, Luo, and Huang 2018) However, they have two major limitations. (1) **Insufficient parameter sharing**: Previous studies always model different roles separately (i.e., different tags or classifiers for various roles), and different roles are trained independently. Such a separated paradigm is natural to restrict the optimization of some roles with lim-

ited resources. (2) **Insufficient utilizing semantics of the roles**: Existing methods hope that the model can learn the patterns of extracting different roles from the raw texts. They treat the roles as labels, without allowing the model to understand the meaning of labels. For example, extract argument corresponding to role "*Attacker*", existing methods always treat the role as a tag or a category or leverage a separated classifiers to extract. They do not tell model the meaning of "*Attacker*". Under this paradigm, extracting argument with few samples is highly difficult. Above issues call for a method that can be fully-parameter sharing and leverage the semantic information of the roles.

From another perspective of the data, to mitigate the impact of data sparseness and cope with low-resource scenarios, there has been a surge of interest in data generation (Chen et al. 2017; Yang et al. 2018) and data augmentation (Liu et al. 2019; Yang et al. 2019). These methods tend to rely heavily on external resources. However, these resources are often incomplete (i.e., events are not always found in external sources) and laborious to build. Aforementioned difficulties motivate us to study semi-supervised event argument extraction method, which seeks to exploit limited event data to annotate substantial unlabeled real sentences automatically. This kind of approach reduces the dependence on external resources. Although traditional semi-supervised learning has many advantages(Rosenberg, Hebert, and Schneiderman 2005; Lee 2013; Miyato et al. 2019), applying this framework to EAE tasks directly has great risks of error propagation, especially in such a complicated task. To overcome this problem, we aim to design an approach, which can benefit from unlabeled data in an effective manner.

To conquer challenges in both model and data aspects, we proposed our framework **DualQA**, which is a question answering based semi-supervised event argument extraction approach. Concretely, from the model aspect, in order to share parameters as much as possible and leverage the role semantics, we formulate the EAE as Machine Reading Comprehension (MRC, we regard MRC as a kind of question answering), where the most advanced method is proven exceed human beings in some specific datasets (Devlin et al. 2019). In this framework, we leverage question to represent the semantics of the role. Besides, we convert EAE task to predict the argument span corresponding to the giving role according to the event schema. This process is consistent for all roles, which achieves the purpose of fully-parameter sharing, as Figure 1 top illustrating (i.e., to extract the argument, giving role: "*Attacker*" $\rightarrow$ "*What plays the role Attacker in the event Attack ?*" $\rightarrow$ argument: "*Iraqi troops*"). Formally, given the event triggers and event types, generating questions containing role information according to the event schema and extracting event arguments from natural language text in turn can be defined as event argument recognition (EAR, **role** $\rightarrow$ **argument**). From the data aspect, we design a dual training process, which can make up for the shortcomings of traditional semi-supervised learning framework. Intuitively, as Figure 1 bottom illustrating (i.e., to recognize the role, giving argument: "*Iraqi troops*" $\rightarrow$ "*What is the role of Iraqi troops in the event Attack ?*" $\rightarrow$ role: "*Attacker*"), also given the event information, generating question burying argument information according to the argument candidate[1] as well as determining it belonging to a specific role in turn can be defined as the dual task of EAR, which we call event role recognition (ERR, **argument** $\rightarrow$ **role**). Specifically, the process of EAR and ERR are of great similarity, and further, their output can be verified by each other (i.e., ERR can generate question from argument extracted by EAR to verify EAR's result, meanwhile EAR can generate question from role identified by ERR to verify ERR's result). However, due to the ERR process is not fully-parameter sharing and depends on the argument candidate which is sometimes unreliable in real situation, we define the EAR as **primal task** and ERR as **dual task**. In the process of semi-supervised training, the two models can mutually collaborate with each other (i.e., two models share the same question understanding module), make up for each other (i.e., both models generate each other's training data), and be enhanced at the same time. Our contributions can be summarized as follows:

- We design a novel semi-supervised framework **DualQA** (dual question answering) to solve the event argument extraction in low-resource scenarios.
- To share parameters as much as possible and leverage the role semantics, we propose EAR and ERR under the question answering paradigm. To reduce the error propagation of traditional semi-supervised methods, we propose a dual training process to utilize the duality of EAR and ERR.
- We conduct extensive experiments on two public event extraction datasets and our method significantly outperforms SOTA methods in low-resource situations.

Methodology

In this section, we will introduce the details of **DualQA** framework to semi-supervised event argument extraction. As we said in the introduction, proposed approach improves the traditional EAE in terms of both model structure and semi-supervised training process. Consequently, we first present the main components of our model, then introduce the semi-supervised training process.

Dual Model Design

The architecture of **DualQA** is illustrated in Figure 2. **DualQA** consists of two models: EAR model $\mathcal{M}_\theta^a$ and ERR model $\mathcal{M}_\phi^r$, where θ and ϕ are their model parameters. Given the event mention (i.e., sentence containing events) x_s, event trigger information x_{tr}, event type x_{ts}, they can be considered as context information C (i.e., $C = \{x_s; x_{tr}; x_{ts}\}$). According to the event schema, EAR generates the question burying the information of roles (r). Afterwards, EAR aims to extract an appropriate argument (a) to the role, which is to approximate $p(a|r, C)$. Also given the same context information C, ERR generates the question burying the information of arguments (a) based on the argument candidates (i.e., utilizing NER in real scene). Then,

[1]Using name entity recognition (NER) in real situations

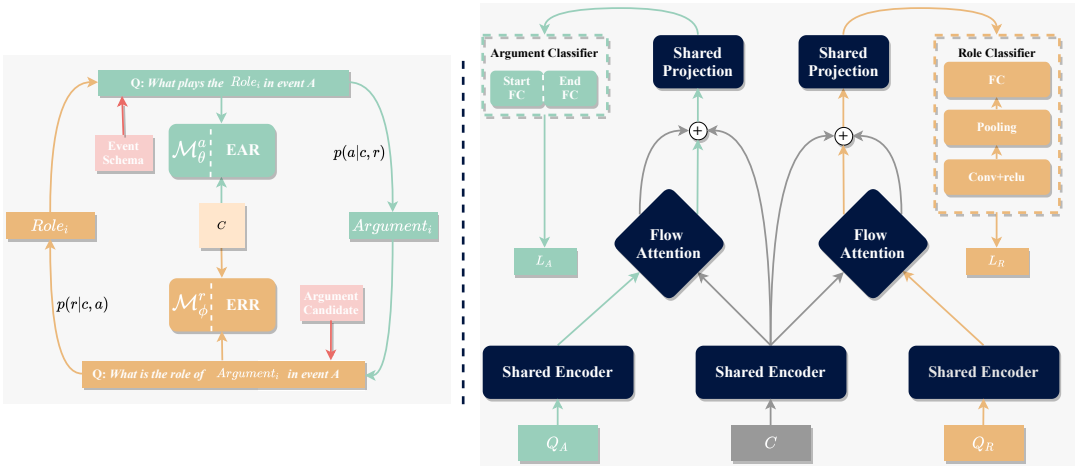

Figure 2: The model design of DualQA. Left: The overall architecture, where C contains the event mention information and the event information provided by event detection. The green parts represent the EAR process ($\mathcal{M}_\theta^a$), while the orange parts represent ERR process ($\mathcal{M}_\phi^r$). Right: The implements structure of DualQA. The green and orange parts are for EAR and ERR respectively. The blue-black parts are shared by the two model.

ERR tries to determine a role (r) to the argument, which is to approximate $p(r|a, C)$.

Next, the rest of this subsection will introduce the model details in the following aspects: question generation, instance encode, flow attention, argument classifier, role classifier.

Question Generation Question generation aims to model event information and role or argument information in question text. In order to help the model understand the semantics of the role, we have incorporated conceptual knowledge into the question. In view of the above, we design simple fixed templates for EAR and ERR. Given the role words x_r, event type words x_{ts}, proposed approach utilize the role words to find the corresponding concept descriptions $\mathbf{X}_d = \{x_d^1, ..., x_d^n\}$ via concept net (Speer and Havasi 2013). Then the question is generated as: "*What plays the role* x_r *in* x_{ts} *?* $(x_d^1, ..., x_d^n)$". Similarly, given the role words x_a, event type words x_{ts}, the question is generated as: "*What is the role of* x_a *in* x_{ts} *?*" (without knowledge).

Instance Encode Our encoding module is a BERT-based(Devlin et al. 2019) contextualized encoder, which is leveraged to encode the context and question generate by EAR and ERR (as shown in Figure 2). We denote the context information as a sequence

$$C = \{[CLS]x_{ts}[SEP]x_1, x_2, ...[SEP]x_{tr}[SEP]...x_n\},$$

where x_{ts} is the words of event type, x_{tr} denotes trigger words, $x_s = x_1, x_2, ..., x_n$ denotes the event mention and $[CLS], [SEP]$ are special tokens of BERT. We encode the event context into hidden representations

$$\mathbf{H}^C = \{\mathbf{h}_i^c\}_{i=1}^{|C|} = \mathbf{BERT}(C) \in \mathbb{R}^{|C| \times d}, \quad (1)$$

where d stands for hidden size. For the purpose of maximizing parameter-sharing, the same encoder will be used for encoding the EAR's question text $Q_A = \{q_a^1, q_a^2, ..., q_a^n\}$ and ERR's question text $Q_R = \{q_r^1, q_r^2, ..., q_r^n\}$, which is

$$\mathbf{U}^A = \{\mathbf{u}_j^a\}_{j=1}^{|Q_A|} = \mathbf{BERT}(Q_A) \in \mathbb{R}^{|Q_A| \times d},$$
$$\mathbf{U}^R = \{\mathbf{u}_j^r\}_{j=1}^{|Q_R|} = \mathbf{BERT}(Q_R) \in \mathbb{R}^{|Q_R| \times d}. \quad (2)$$

Flow Attention The main purpose of flow attention module is to couple the question and context matrix and produce a set of query-aware feature vectors for each word in the context, which is following Seo et al. (2017). The flow attention is not used to summarize the question or context into single feature vectors, which reduce the loss of information caused by early summarisation. The attention will be computed from two directions: from context to question (C2Q) as well as from question to context (Q2C). First calculate the similarity matrix $\mathbf{S}^A \in \mathbb{R}^{|C| \times |Q_A|}$ between EAR question ($\mathbf{U}^\mathbf{R}$) and context ($\mathbf{H}^C$), $\mathbf{S}^R \in \mathbb{R}^{|C| \times |Q_R|}$ between ERR question ($\mathbf{U}^A$) and context ($\mathbf{H}^C$), where s_{ij} indicates the similarity between i-th context word and j-th question word (EAR question or ERR question). The similarity matrix is formulated as

$$s_{ij}^A = \mathbf{E}(\mathbf{h}_i^c, \mathbf{u}_j^a) \in \mathbb{R}, s_{ij}^R = \mathbf{E}(\mathbf{h}_i^c, \mathbf{u}_j^r) \in \mathbb{R}, \quad (3)$$

where $\mathbf{E}(\cdot)$ is a trainable scalar function modeling the similarity between two input vectors. we compute it as

$$\mathbf{E}(\mathbf{h}, \mathbf{u}) = \mathbf{mlp}([\mathbf{h}; \mathbf{u}; \mathbf{h} \circ \mathbf{u}]), \quad (4)$$

where $\mathbf{mlp}(\cdot)$ is a multilayer perceptron (MLP), $[;]$ denotes the vector concatenation operation and $\circ$ is elementwise multiplication. After that, utilize $\mathbf{S}$ to obtain the attentions from both direction for EAR and ERR. From context to question (C2Q) attention signifies the most relevant question words to context, which can be formulated as

$$a_{ij} = \frac{exp(s_{ij})}{\sum_{j=1}^{|Q|} exp(s_{ij})} \in \mathbb{R},$$
$$\tilde{\mathbf{u}}_i = \sum_j a_{ij} \mathbf{u}_j \in \mathbb{R}^d, \quad (5)$$

where a_{ij} represents the context to question attention weight, $\mathbf{U}$ can be $\mathbf{U}^A$ or $\mathbf{U}^R$ and $\tilde{\mathbf{u}}_i$ forms the context-aware question matrix $\tilde{\mathbf{U}}$. From another direction question to context (Q2C) attention signifies the closet similarity to one of the question words which are hence critical for answering the question. We calculate the Q2C attention as

$$b_i = \frac{exp(max(\mathbf{S}_{i:}))}{\sum_{i=1}^{|C|} exp(max(\mathbf{S}_{i:}))} \in \mathbb{R},$$
$$\tilde{\mathbf{h}}_i = \sum_i b_i \mathbf{h_i} \in \mathbb{R}^d, \tag{6}$$

where b_i represents the question to context attention weight, $\mathbf{S}_{i:}$ denotes the i-th row's elements of similarity matrix $\mathbf{S}$ and $\tilde{\mathbf{h}}_i$ forms the context matrix $\tilde{\mathbf{H}}$ indicates the most important words in the context. Finally, as Figure 2 right illustrating, both EAR and ERR share same projection to combine features above to query-aware representations, which can be formulated as

$$\mathbf{g}_i = \mathbf{F}([\mathbf{h}_i^c, \tilde{\mathbf{u}}_i, \tilde{\mathbf{h}}_i]) \in \mathbb{R}^d,$$
$$\mathbf{E}(\mathbf{h}, \tilde{\mathbf{u}}, \tilde{\mathbf{h}}) = \mathbf{mlp}([\mathbf{h}; \tilde{\mathbf{u}}; \mathbf{h} \circ \tilde{\mathbf{u}}; \mathbf{h} \circ \tilde{\mathbf{h}}]), \tag{7}$$

where $\mathbf{mlp}(\cdot)$ is a multilayer perceptron (MLP), $[;]$ denotes the vector concatenation operation and $\circ$ is element-wise multiplication. At last, we get the question-aware context representation for EAR ($\mathbf{G}^A = \{\mathbf{g}_i^a\}_{i=1}^{|C|}$) and ERR ($\mathbf{G}^R = \{\mathbf{g}_i^r\}_{i=1}^{|C|}$).

Argument Classifier After getting the context features, we feed it into the token classifiers to predict the probability of if the token is the start of an argument or end:

$$p(a_s|C, r, \theta) = \mathbf{o}^s = \mathbf{Softmax}(\mathbf{W}_s \cdot \mathbf{G}^a),$$
$$p(a_e|C, r, \theta) = \mathbf{o}^e = \mathbf{Softmax}(\mathbf{W}_e \cdot \mathbf{G}^a), \tag{8}$$

in which $\mathbf{W}_s$ and $\mathbf{W}_e$ are learnable parameters, $\mathbf{o}^s$ and $\mathbf{o}^e$ are the start and end probability distributions predicted by model, θ is all parameters of EAR.

Role Classifier As shown in Figure 2 right, we use a simple CNN (Kim 2014; Krizhevsky, Sutskever, and Hinton 2012) to classify roles:

$$p(r|C, a, \phi) = \mathbf{o}^r = \mathbf{Softmax}(\mathbf{W}_r \cdot \mathbf{CNN}(\mathbf{G}^r)), \tag{9}$$

in which $\mathbf{W}_r$ is learnable parameter, $\mathbf{o}^r$ are the role probability distributions predicted by model, ϕ is all parameters of EAR.

Semi-supervised Dual Training Strategy

This subsection introduces the semi-supervised dual training strategy. The training process is similar to the self-training process (Rosenberg, Hebert, and Schneiderman 2005), in each iteration we need to train the model jointly first, and then use the model with a certain ability to annotate the unlabeled data. After that, add them to the labeled data to rebuild the training set. Each round of training can be divided into two processes: joint train and annotate the unlabeled data. The iterative process stops when the unlabeled data is exhausted or our model converges.

Joint Train In the joint train phase, both EAR model ($\mathcal{M}_\theta^a$) and ERR model ($\mathcal{M}_\phi^r$) are optimized alternative at the same time like generative adversarial nets (Goodfellow et al. 2014) on the training set (S_A for EAR and S_R for ERR), the objective function is given below:

$$\begin{aligned}\mathbf{O}(\theta, \phi) &= \mathbf{O}(\theta) + \mathbf{O}(\phi) \\ &= max(\mathbb{E}_{(c,r,a)\in S_A}[log(p(a|c, r, \theta))] \\ &+ \mathbb{E}_{(c,r,a)\in S_R}[log(p(r|c, a, \phi))]),\end{aligned} \tag{10}$$

where θ and ϕ indicate the parameters of EAR model and ERR model. Given an event context c and one of the argument (a) as well as the role (r), EAR model seeks to maximize the probability of a, in contrast, ERR model tries to maximize the probability of r. Formula 10 can be decomposed into optimized EAR and ERR. The objective function of EAR model is shown below:

$$\begin{aligned}\mathbf{O}(\theta) = -min \sum_{k=1}^{|S_A|} (&log(p(a_s^k|c^k, r^k, \theta)) \\ &+ log(p(a_e^k|c^k, r^k, \theta))),\end{aligned} \tag{11}$$

in which (a^k, c^k, r^k) is the k-th sample in training set S_A, (a_s^k, a_e^k) indicates the start and end position of the argument a^k. Similarly, the objective function of ERR model is:

$$\mathbf{O}(\phi) = -min \sum_{k=1}^{|S_R|} log(p(r^k|c^k, a^k, \phi)), \tag{12}$$

where (a^k, c^k, r^k) is the k-th sample in training set S_R.

Label Data After joint train process, obtain EAR model and ERR model with certain capabilities mutually annotate the unlabeled data (S_U) which is more reliable than only sampling from one distribution. Given an event mention without argument, its event type, and the event schema, we build (context (c), role (r)) pair for each role in event schema, and then estimate the argument via EAR model, which is $\hat{a} = \mathcal{M}_\theta^a(c, r)$. After that, feed $(c, \hat{a})$ pair into ERR model for verification, which is $\hat{r} = \mathcal{M}_\phi^r(c, \hat{a})$. When $\hat{a}$ and $\hat{r}$ are not negative predictions, and $\hat{r} = r$, pair $(c, \hat{a}, r)$ is considered a credible annotation. The annotations of ERR is also checked in the same way. Given the context information, we build (context (c), argument (a)) pair for each argument in argument candidate (NER result (Shaalan 2014; Lample et al. 2016) in unlabeled set). Similarly, we get the estimated role $\hat{r}$ by calculating $\hat{r} = \mathcal{M}_\phi^r(c, a)$, and then verify which by computing $\hat{a} = \mathcal{M}_\theta^a(c, \hat{r})$, when $\hat{a}$ and $\hat{r}$ are not negative predictions, and $\hat{a} = a$, pair $(c, a, \hat{r})$ is considered a reliable annotation. Then these credible data will be added to the labeled data to build a new training set. We summarize the training process for DualQA in algorithm 1. The whole process works in an iterative manner. In each iteration, proposed method rebuild a new training set utilized by optimizing both EAR model and ERR model. The iterative process stops when our model converges or the unlabeled set is exhausted.

Algorithm 1 DualQA Learning Algorithm

Input: Labeled data $S_A = \{(c_i, a_i, r_i)\}_{i=1}^{|S_A|}$ and $S_R = \{(c_i, a_i, r_i)\}_{i=1}^{|S_R|}$, unlabeled data $S_U = \{(c_j)\}_{j=1}^{|S_U|}$

while $S_U \neq \emptyset$ **and** *not converge* **do**
  $\mathcal{M}_\theta^a, \mathcal{M}_\phi^r \leftarrow$ Initialize
  $\mathcal{M}_\theta^a, \mathcal{M}_\phi^r \leftarrow$ Joint train using S_A and S_R (Eq. 10)
  for all c_j in S_U **do**
    for all r in event schema of c_j **do**
      $\hat{a} \leftarrow \mathcal{M}_\theta^a(c_j, r)$
      $\hat{r} \leftarrow \mathcal{M}_\phi^r(c_j, \hat{a})$
      if $\hat{a}$ not *neg* and $\hat{r}$ not *neg* and $\hat{r} = r$ **then**
        Append $(c_j, \hat{a}, r)$ to S_A and S_R
      end if
    end for
    for all a in argument candidate of c_j **do**
      $\hat{r} \leftarrow \mathcal{M}_\phi^r(c_j, a)$
      $\hat{a} \leftarrow \mathcal{M}_\theta^a(c_j, \hat{r})$
      if $\hat{a}$ not *neg* and $\hat{r}$ not *neg* and $\hat{a} = a$ **then**
        Append $(c_j, a, \hat{r})$ to S_A and S_R
      end if
    end for
    if all role of c_j and all argument related to c_j has credible answer **then**
      Remove (c_j) from S_U
    end if
  end for
end while

Output: Enhanced $\mathcal{M}_\theta^a$

Experiments

In this section we conduct experiments to evaluate proposed method. We first introduce the basis settings, including dataset and evaluation, baselines, and experimental settings. Then we illustrate performance comparison results with baseline methods. Finally we introduce the effectiveness of various components of our approach.

Datasets and Evaluation

We choose two public event extraction datasets from completely different fields to validate the effectiveness and annotation ability of our method. (1) **ACE 2005 English corpus**(Doddington et al. 2004): ACE 2005 corpus is a standard benchmark dataset which is widely adopted for evaluating event extraction systems. ACE 2005 corpus is collected from daily data, such as weblogs, news, broadcast conversation and so on. We adopt the configuration as (Liu, Luo, and Huang 2018), in which 529/30/40 documents are use as train/dev/test sets and the time-related tags have merged as one tag "*Time*". As shown in Table 1, ACE 2005 English corpus is an English dataset with extremely sparse in events. (2) **FewFC** [2]: FewFC is a public Chinese dataset for few-shot event extraction in financial field. As shown in Table 1, FewFC corpus is a field-specific Chinese dataset, in which the average role contained in each event type is relatively large. Since the amount of data we obtained at the time of writing this paper is very limited, we only used 7 event types. In addition, we divide the data into 8:1:1 as train/dev/test sets. And in order to make the label of the role more semantic, we semanticize the label according to the type of event. Table 1 shows the overall statistics of two datasets, which are of great difference in regardless of languages, domain or sparseness. Furthermore, for EAR task, there are naturally a large amount of event argument missing in events, which can be the negative samples for EAR. In addition, for the training process of ERR, we constructed negative samples 1:1 relative to the positive samples, which come from NER or argument reduction (e.g., *"Iraqi troops"* to *"Iraqi"*) or expansion (e.g., *"Iraqi troops"* to *"Iraqi troops had"*).

Dataset	#sentences	#roles	#event types	%Neg.
ACE 2005	3887	28	33	68.50
FewFC	8982	19	10	49.58

Table 1: Statistics for ACE 2005 and FewFC dataset. In particular, %Neg. implies the percentage of "no argument" according to the event schema.

The evaluation metrics adopts (Chen et al. 2015; Liu, Luo, and Huang 2018; Nguyen, Cho, and Grishman 2016) argument role classification evaluation strategy: an argument prediction is correct only if its span and role it plays match with golden label.

Baselines

We compare our approach with BERT-based state-of-the-art baselines and their enhanced versions: (1) **BERT-EE**(Devlin et al. 2019): BERT-based sequence labeled model. (2) **PLMEE**(Yang et al. 2019): Current event extraction state-of-the-art, which base on BERT, and utilizes different classifiers for different roles. (3) **self-training**(Rosenberg, Hebert, and Schneiderman 2005): Self-training is a semi-supervised learning method that uses a single model's predictions on unlabeled data to retrain iteratively. All methods with $*$ in the experiment exploit self-training means (i.e., annotating once and training twice).

Additionally, we also conduct ablation study to further analyze our approach: (1) **EAR**: Only leverage the EAR process for training or self-training. EAR model is similar to Du and Cardie (2020), but entire model is implemented by ourselves and adopts our own question generation strategy. (2) **Joint-EAR-ERR**: The result of joint training EAR and ERR only on the given labeled data, without annotating the unlabeled data. (3) **DualQA**: Joint train EAR and ERR and annotate the unlabeled data to boost each other's performance.

Experimental Settings

Hyper-parameter Settings All methods above base on the same pretrained BERT (Devlin et al. 2019) (BERT-BASE-UNCASED for English dataset and BERT-BASE-CHINESE for Chinese dataset) [3]. We trained all model with initial learning rate 1e-5, and AdamW optimizer (Loshchilov and Hutter

[2] https://github.com/TimeBurningFish/FewFC

[3] https://github.com/google-research/bert

Method/Dataset	ACE			FewFC		
	P	R	F1	P	R	F1
BERT-EE(Devlin et al.)	26.7	38.2	31.4	18.9	35.9	24.8
BERT-EE*	28.3	41.9	33.8	19.4	37.6	25.6
PLMEE(Yang et al.)	36.3	46.8	40.9	52.0	30.9	38.8
PLMEE*	37.6	46.6	41.6	54.1	31.9	40.2
DualQA	49.1	42.3	**45.4**	57.4	34.4	**43.1**

Table 2: Overall performance comparison in ACE 2005 English corpus and FewFC.

Method/Dataset	ACE			FewFC		
	P	R	F1	P	R	F1
BERT-EE(Devlin et al.)	26.7	38.2	31.4	18.9	35.9	24.8
BERT-EE*	28.3	41.9	33.8	19.4	37.6	25.6
EAR	33.6	42.6	37.5	34.8	28.4	32.0
EAR*	44.2	35.4	39.3	40.0	30.2	34.4
DualQA	49.1	42.3	**45.4**	57.4	34.4	**43.1**

Table 3: MRC performance comparison in ACE 2005 English corpus and FewFC.

2017), the convolution neural network used in ERR has three convolutional layer with 256 hidden nodes, and filter size are $3\times3/4\times4/5\times5$, other settings of hyper-parameters are following the configuration of BERT. For ERR model, we add an extra type as negative category, and for EAR, when both start and end classifier consider the "[CLS]" token to be the position, this sample is negative.

Data Settings In order to study the performance of semi-supervised event argument extraction method with diverse labeled data sizes and evaluate the performance under low-resource condition, for all datasets, we first retained 60% training data as unlabeled set (the human annotation is not visible during training), then sample a percentage of data from remaining training set as labeled set. There are no overlap between labeled set and unlabeled set. The two sets are the same for all methods as well. During the whole process, we only rebuild the training set, and there is not any sampling operation in test or validation set. Furthermore, all approaches are provided the same context information introduced in Methodology (event mention, event type, event trigger).

Comparisons with SOTA Methods

To build a low-resource environment, following the sample strategy mentioned before, in ACE 2005 English corpus, we sample 10% training data as labeled set and 60% training data as unlabeled set. Besides, in FewFC, we sample 1% training data as labeled set and 60% training data as unlabeled set. Table 2 summarizes the performance comparison between aforementioned SOTA models and our approach in the same test set of two datasets. Under low-resource settings, DualQA can outperform other methods (3.8% on F1 score for ACE and 2.9% on F1 for FewFC), which justifies our approach can learn patterns from few samples and leverage the unlabeled data effectively regardless of Chinese or English, general or financial. And the improvement of DualQA on precision score is quite significant (11.5% in ACE and 3.3% in FewFC). This is probably due to the mutual verification between them in annotating unlabeled data. Moreover, semi-supervised methods are better than fully-supervised methods in almost all points, which further illustrates utilizing unlabeled data can help the model to have generalization capability under low-resource scenarios. This is why we want to study semi-supervised event argument extraction.

Method/Dataset	ACE			FewFC		
	P	R	F1	P	R	F1
PLMEE(Yang et al.)	36.3	46.8	40.9	52.0	30.9	38.8
PLMEE*	37.6	46.6	41.6	54.1	31.9	40.2
EAR	33.6	42.6	37.5	34.8	28.4	32.0
Joint-EAR-ERR	40.5	42.2	41.4	40.0	43.0	41.5
DualQA	49.1	42.3	**45.4**	57.4	34.4	**43.1**

Table 4: Dual learning performance comparison in ACE 2005 English corpus and FewFC.

Ablation Study

The effectiveness of MRC framework. We study the effects of applying MRC framework. The data settings are same as "Comparisons with SOTA methods". Table 3 illustrates the performance comparison between MRC-based method (EAR) and sequence labeling model (BERT-EE) on two datasets. MRC-based methods make significant improvements compare with the sequence labeling model. We analyze this is due to the sequence labeling model suffers from insufficient parameter sharing which is catastrophic especially with few samples. And it also can not solve the roles overlap problem (Yang et al. 2019) (i.e., one argument plays different roles). By contrast, methods only based on question answering can not only leverage the semantics of roles to enhance robustness in low-resource situations, but also solve the issue of roles overlap. However, methods based on MRC framework (EAR) is still much worse than our method (DualQA).

The effectiveness of dual learning. We study the effects of proposed dual learning framework. The data settings are same as "Comparisons with SOTA methods". As can be seen in Table 4, the two tasks can mutually correct each other, and benefit from unlabeled data. Furthermore, comparing EAR model with Joint-EAR-ERR, although MRC framework has above advantages, a single question answering perspective is still not enough, and the dual task can bring a significant improvement. In addition, comparing Joint-EAR-ERR as well as DualQA and PLMEE as well as PLMEE*, our approach is more efficient in benefiting from unlabeled data than self-training (4% (ours) compare with 0.7% (PLMEE) in ACE and 1.6% (ours) compare with 1.4% (PLMEE) in FewFC).

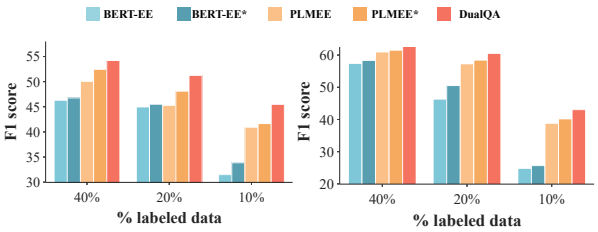

Figure 3: Performance comparison with different amounts of labeled training data and 60% unlabeled training data. Left: ACE 2005 English corpus. Right: FewFC.

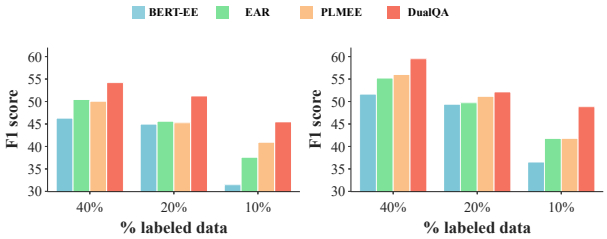

Figure 4: Annotation quality comparison with different amounts of labeled training data and 60% unlabeled training data in ACE 2005. Left: Performance in test set. Right: Annotation quality of unlabeled set.

The effectiveness under different amounts of labeled data. In real scenarios, manually labeled data is of great laborious. We aim to leverage massive unlabeled data to improve our model with limited labeled data. Therefore, we wonder how is our method performances under different amounts of labeled data. As the sample strategy mentioned before, we sample 60% training data as unlabeled set for both ACE and FewFC, and then sample different amounts of data (10%/20%/40% for ACE and 1%/5%10% for FewFC) from remaining training data as labeled set. As Figure 3 presents, our approach (DualQA) outperform other methods under all data conditions we tried, but it is more robust than the baseline under extremely low resource situations (10% in ACE and 1% in FewFC).

The quality of annotations. The unlabeled data we used in the experiment are all derived from the datasets itself. In that case, we can automatically evaluate the quality of annotations from various methods on the "unlabeled set". We sample 60% training data as unlabeled set from ACE, and then sample different amounts of training data (10%/20%/40%) as labeled set. We evaluate the annotation quality of different models on the unlabeled set and test set, as shown in Figure 4. The annotations quality of our method outperforms other methods, which proves the effectiveness of our method in annotating unlabeled data in low-resource situations.

Related Work

Event argument extraction. Event argument extraction is a key step of event extraction, where various methods have been proposed. Traditional approaches (Chen et al. 2015; Liu, Luo, and Huang 2018; Nguyen, Cho, and Grishman 2016; Yang et al. 2019) take advantage of neural network to automatically fit the distribution of training samples, which is often limited by the amount of data. Several works (Mintz et al. 2009; Chen et al. 2017; Yang et al. 2018, 2019) try to leverage the external resources to generate event for making up the shortcoming. The general idea of these works is to leverage either NLP systems (e.g., translation model) or external knowledge bases (e.g., Freebase) under the distant supervision (Mintz et al. 2009) to build an external corpus. However, the external resources are relatively more difficult to obtain than unlabeled data. Motivated by this, our method hopes to benefit from unlabeled data.

Machine reading comprehension Explorations on converting traditional NLP tasks into question answering task has drawn widespread concern, recently. Levy et al. (2017) leverage the MRC framework to settle the zero-shot relation extraction. Li et al. (2019) convert entity-relation extraction as multi-turn question answering. Du and Cardie (2020) and Zhang et al. (2020) model the EAE task as question answering, but their methods are fully-supervised and only model the EAR process as well. Above works illustrate that the MRC paradigm has certain advancements in NLP field. In view of this, we propose two EAE tasks (i.e., EAR and ERR) based on question answering framework.

Dual learning. Dual learning, proposed by He et al. (2016), aims to make use of the duality between the primal task and the dual task to leave the two model mutually benefit from each other and boost each other's performance at the same time. There are various way of cooperation, such as target-source translation and source-target translation (He et al. 2016), query-response conversation and response-query conversation (Shen and Feng 2020), question answering and question generation (Li et al. 2018), and so on. Distinct from them, we took the advantage of the duality between the two question answering tasks under the semi-supervised settings to settle the event argument extraction. We are also the first to perform dual learning framework in this task.

Conclusion and Future Work

In this paper, we proposed a new framework dual question answering (DualQA) for event argument extraction in a semi-supervised learning manner. We define EAR and ERR under the question answering paradigm to share parameter as much as possible, and utilize the semantics of the role. Besides, we propose a dual training process, which encourages the two model mutually enhance each other and verify each other's annotation to reduce the impact of error propagation. We conduct extensive experiments on two public datasets. And the experimental results prove the effectiveness of our approach. In the future, we will further explore the scalability of our method to other tasks.

Acknowledgements

This work is supported by the National Key Research and Development Program of China (2020AAA0106400), the National Natural Science Foundation of China (No.61533018, No.61976211,No.61806201). This work is also supported by the CCF-Tencent Open Research Fund and the independent research project of National Laboratory of Pattern Recognition.

References

Chen, Y.; Liu, S.; Zhang, X.; Liu, K.; and Zhao, J. 2017. Automatically Labeled Data Generation for Large Scale Event Extraction. In *Proceedings of the 55th ACL 2017*, 409–419. Association for Computational Linguistics.

Chen, Y.; Xu, L.; Liu, K.; Zeng, D.; and Zhao, J. 2015. Event Extraction via Dynamic Multi-Pooling Convolutional Neural Networks. In *Proceedings of the 53rd ACL and the 7th IJCNLP*, 167–176. The Association for Computer Linguistics.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the NAACL-HLT 2019*, 4171–4186. Association for Computational Linguistics.

Doddington, G. R.; Mitchell, A.; Przybocki, M. A.; Ramshaw, L. A.; Strassel, S. M.; and Weischedel, R. M. 2004. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Lrec*, volume 2, 837–840. Lisbon.

Du, X.; and Cardie, C. 2020. Event Extraction by Answering (Almost) Natural Questions. *CoRR* abs/2004.13625.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A. C.; and Bengio, Y. 2014. Generative Adversarial Nets. In *NIPS 2014*, 2672–2680.

He, D.; Xia, Y.; Qin, T.; Wang, L.; Yu, N.; Liu, T.; and Ma, W. 2016. Dual Learning for Machine Translation. In *NIPS 2016*, 820–828.

Hirschberg, J.; and Manning, C. D. 2015. Advances in natural language processing. *Science* 349(6245): 261–266.

Kim, Y. 2014. Convolutional Neural Networks for Sentence Classification. In *Proceedings of the EMNLP 2014*, 1746–1751. ACL.

Krizhevsky, A.; Sutskever, I.; and Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks. In *NIPS 2012*, 1106–1114.

Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; and Dyer, C. 2016. Neural Architectures for Named Entity Recognition. In *NAACL HLT 2016*, 260–270. The Association for Computational Linguistics.

Lee, D.-H. 2013. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3.

Levy, O.; Seo, M.; Choi, E.; and Zettlemoyer, L. 2017. Zero-Shot Relation Extraction via Reading Comprehension. In *Proceedings of CoNLL 2017*, 333–342. Association for Computational Linguistics.

Li, X.; Yin, F.; Sun, Z.; Li, X.; Yuan, A.; Chai, D.; Zhou, M.; and Li, J. 2019. Entity-Relation Extraction as Multi-Turn Question Answering. In *Proceedings of the 57th ACL 2019*, 1340–1350. Association for Computational Linguistics.

Li, Y.; Duan, N.; Zhou, B.; Chu, X.; Ouyang, W.; Wang, X.; and Zhou, M. 2018. Visual Question Generation as Dual Task of Visual Question Answering. In *CVPR 2018*, 6116–6124. IEEE Computer Society.

Liu, J.; Chen, Y.; and Liu, K. 2019. Exploiting the Ground-Truth: An Adversarial Imitation Based Knowledge Distillation Approach for Event Detection. In *Proceedings of AAAI 2019*, 6754–6761. AAAI Press.

Liu, J.; Chen, Y.; Liu, K.; and Zhao, J. 2019. Neural Cross-Lingual Event Detection with Minimal Parallel Resources. In *EMNLP-IJCNLP 2019*, 738–748. Association for Computational Linguistics.

Liu, X.; Luo, Z.; and Huang, H. 2018. Jointly Multiple Events Extraction via Attention-based Graph Information Aggregation. In *Proceedings of the EMNLP 2018*, 1247–1256. Association for Computational Linguistics.

Loshchilov, I.; and Hutter, F. 2017. Fixing Weight Decay Regularization in Adam. *CoRR* abs/1711.05101.

Mintz, M.; Bills, S.; Snow, R.; and Jurafsky, D. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the 47th ACL and 4th IJCNLP 2009*, 1003–1011. The Association for Computer Linguistics.

Miyato, T.; Maeda, S.; Koyama, M.; and Ishii, S. 2019. Virtual Adversarial Training: A Regularization Method for Supervised and Semi-Supervised Learning. *IEEE Trans. Pattern Anal. Mach. Intell.* 41(8): 1979–1993.

Nguyen, T. H.; Cho, K.; and Grishman, R. 2016. Joint Event Extraction via Recurrent Neural Networks. In *NAACL HLT 2016*, 300–309. The Association for Computational Linguistics.

Rosenberg, C.; Hebert, M.; and Schneiderman, H. 2005. Semi-Supervised Self-Training of Object Detection Models. In *7th IEEE (WACV/MOTION 2005)*, 29–36. IEEE Computer Society.

Seo, M. J.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2017. Bidirectional Attention Flow for Machine Comprehension. In *5th ICLR 2017*. OpenReview.net.

Shaalan, K. 2014. A Survey of Arabic Named Entity Recognition and Classification. *Comput. Linguistics* 40(2): 469–510.

Shen, L.; and Feng, Y. 2020. CDL: Curriculum Dual Learning for Emotion-Controllable Response Generation. In *Proceedings of the 58th ACL 2020*, 556–566. Association for Computational Linguistics.

Speer, R.; and Havasi, C. 2013. ConceptNet 5: A Large Semantic Network for Relational Knowledge. In *The People's*

Web Meets NLP, Collaboratively Constructed Language Resources, Theory and Applications of Natural Language Processing, 161–176. Springer.

Tong, M.; Xu, B.; Wang, S.; Cao, Y.; Hou, L.; Li, J.; and Xie, J. 2020. Improving Event Detection via Open-domain Trigger Knowledge. In *Proceedings of the 58th ACL 2020*, 5887–5897. Online: Association for Computational Linguistics.

Wang, X.; Wang, Z.; Han, X.; Liu, Z.; Li, J.; Li, P.; Sun, M.; Zhou, J.; and Ren, X. 2019. HMEAE: Hierarchical Modular Event Argument Extraction. In *Proceedings of the EMNLP-IJCNLP 2019*, 5776–5782. Association for Computational Linguistics.

Yang, H.; Chen, Y.; Liu, K.; Xiao, Y.; and Zhao, J. 2018. DCFEE: A Document-level Chinese Financial Event Extraction System based on Automatically Labeled Training Data. In *Proceedings of ACL 2018*, 50–55. Association for Computational Linguistics.

Yang, S.; Feng, D.; Qiao, L.; Kan, Z.; and Li, D. 2019. Exploring Pre-trained Language Models for Event Extraction and Generation. In *Proceedings of the 57th ACL 2019*, 5284–5294. Association for Computational Linguistics.

Zhang, Y.; Xu, G.; Wang, Y.; Lin, D.; Li, F.; Wu, C.; Zhang, J.; and Huang, T. 2020. A Question Answering-Based Framework for One-Step Event Argument Extraction. *IEEE Access* 8: 65420–65431.