Article:

Recurrent Interaction Network for Jointly Extracting Entities and Classifying Relations

Kai Sun
BDBC and SKLSDE
Beihang University,China
sunkai@buaa.edu.cn

Richong Zhang[*]
BDBC and SKLSDE
Beihang University, China
zhangrc@act.buaa.edu.cn

Samuel Mensah
BDBC and SKLSDE
Beihang University, China
samensah@buaa.edu.cn

Yongyi Mao
School of EECS
University of Ottawa,Canada
ymao@uottawa.ca

Xudong Liu
BDBC and SKLSDE
Beihang University, Beijing, China
liuxd@act.buaa.edu.cn

Abstract

The idea of using multi-task learning approaches to address the joint extraction of entity and relation is motivated by the relatedness between the entity recognition task and the relation classification task. Existing methods using multi-task learning techniques to address the problem learn interactions among the two tasks through a shared network, where the shared information is passed into the task-specific networks for prediction. However, such an approach hinders the model from learning explicit interactions between the two tasks to improve the performance on the individual tasks. As a solution, we design a multi-task learning model which we refer to as recurrent interaction network which allows the learning of interactions dynamically, to effectively model task-specific features for classification. Empirical studies on two real-world datasets confirm the superiority of the proposed model.

1 Introduction

The extraction of entities and relations from textual data comprises of two sub-tasks: entity recognition (ER) and relation classification (RC). The ER task aims at extracting all entities in a given text. The RC task aims at classifying the relation between any pair of entities in the text. In practice, both tasks are required to be solved jointly, and have been observed to contribute significantly in extracting structured knowledge from unstructured text for several applications, including knowledge base construction (Komninos and Manandhar, 2017; Deng et al., 2019; Nathani et al., 2019). For instance, consider the sentence *John was born in Sheffield, a city in the north of England*. The goal of a joint entity and relation extraction task is to identify all the factual relational triples (or relational facts) (Sheffield,

birth_place_of, John) and (England, contains, Sheffield).

(a) Flat Structure   (b) Graph Structure

Figure 1: Two topological structures for multi-task learning. Here, A and B are related tasks, and S is the shared information of the two tasks. The directed edges define the information flow.

The simplest approach to solve this joint task is to utilize a pipeline-based approach by firstly extracting all entities in the sentence and then classifying the relation between all entity pairs (Zelenko et al., 2003; Zhou et al., 2005; Chan and Roth, 2011). However, pipeline-based approaches disregard the correlation between ER and RC tasks, leading to error propagation in these methods.

Recently, researchers have exploited multi-task learning (MTL) (Collobert and Weston, 2008) techniques to capture the correlation between the ER and RC tasks, and have successfully improved the performance of the individual tasks (Miwa and Bansal, 2016; Fu et al., 2019; Zeng et al., 2019a). These methods have a flat structure (Liu et al., 2019a). Figure 1(a) shows a flat structure for multi-task learning. Methods using a flat structure learn interactions between tasks through a shared network, and extract a shared common representation which is exploited by task-specific networks independently. We refer to MTL methods utilizing a flat structure as conventional MTL methods. A conventional MTL method is effective to an extent because they help to improve generalization performance on all the tasks. However, it is based on the strong assumption that the shared network is sufficient to capture the correlations between the tasks.

---
[*]Corresponding author

Even so, identifying the relational facts in sentences is a difficult problem. Reason being that several relational facts may overlap in a sentence (Zhang et al., 2018). Although a conventional MTL method may learn task-specific features and has been successfully applied in a wide variety of scenarios (Zhang and Wang, 2016; Wu et al., 2016; Goo et al., 2018; Han et al., 2019; Li et al., 2019; Nishino et al., 2019; Liu et al., 2019b; Hu et al., 2019), its flat structure restricts the model to effectively learn the correlations between tasks. For example in Figure 1(a), the model cannot explicitly learn correlations between the two tasks. Without modeling explicit interactions, as shown in a sequence learning task (Liu et al., 2019a), the existing MTL-based methods (Miwa and Bansal, 2016; Fu et al., 2019; Zeng et al., 2019a) cannot effectively capture the correlation between the ER and the RC tasks.

In this paper, we overcome the aforementioned limitation of previous MTL-based methods by proposing a recurrent interaction network (RIN) to effectively capture the correlations between the ER and RC tasks. RIN has a multi-task learning architecture which allows interactions between the ER and RC tasks to be learned explicitly, with the aim to improve the performance on the individual tasks. More specifically, RIN has a recurrent structure comprising of multiple interaction layers, allowing the model to progressively learn complex interactions while refining predictions for ER and RC. The RIN structure is an example of a multi-task learning network with a graph structure (Liu et al., 2019a). We show the graph structure in Figure 1(b). As shown by our experiment, the proposed model progressively provides discriminating features which is an essential requirement for the individual task for classification. Empirical studies on NYT and WebNLG datasets achieve new state-of-the-art performances and confirm the effectiveness of the proposed RIN model.

2 Related Work

Previous neural methods proposed for jointly extracting entities and relations can generally be categorized into three classes. The first class models the joint extraction task as a sequence labeling problem (Zheng et al., 2017; Dai et al., 2019; Takanobu et al., 2019; Yu et al., 2019). Among the proposed works, (Zheng et al., 2017) was the first to introduce a tagging strategy to address the problem,

transferring the joint extraction task to a sequence labelling problem. However, this method has the fundamental weakness of addressing the overlapping problem of relational facts in the text. To meet it, (Dai et al., 2019) proposed a position-attentive tagging scheme to solve the overlapping problem. Meanwhile, (Takanobu et al., 2019; Yu et al., 2019) approach the problem by decomposing the joint extraction task into two sequence labeling sub-tasks, to address the joint entity and relation extraction problem.

The second class of works use a sequence-to-sequence (seq2seq) approach to address the problem (Zeng et al., 2018, 2019b). (Zeng et al., 2018) employs a seq2seq model to directly extract relational facts from the sentence by decoding the first entity, second entity, and relation in that order. But, their approach is limited to extracting a predefined number of relational facts from the text. In extracting relational triples, the order of extraction is key to identify the relational facts. As such, (Zeng et al., 2019b) proposed a seq2seq approach which utilizes a reinforcement learning model to learn the order of extracting the relational triples. Although effective, the proposed seq2seq models (Zeng et al., 2018, 2019b) only decode a single word for an entity.

The third class design a multi-task learning model to extract relational facts. Only few works using this approach have been proposed (Miwa and Bansal, 2016; Fu et al., 2019; Zeng et al., 2019a). (Miwa and Bansal, 2016) is one of the first works to extract relational facts using an MTL framework. (Zeng et al., 2019a) proposed an MTL model which comprises of an ER model to extract entities with multi-tokens, and a seq2seq model to extract relational facts. Their approach solves the entity extraction problem faced by models which are solely seq2seq based. (Fu et al., 2019) exploited a bidirectional recurrent neural network and graph convolutional network to extract common features of the ER and RC tasks, which are further fed into two independent classifiers for ER and RC predictions. Despite the substantial efforts and great successes in the design of these MTL-based methods, these methods follow the conventional MTL approach (Collobert and Weston, 2008). Thus, they only capture implicit interactions by means of the shared network of the ER and RC tasks.

Modelling explicit interactions between multiple tasks in an MTL architecture has been explored to improve predictions in several domains (He et al.,

2019; Zhao et al., 2019; Dankers et al., 2019; Lan et al., 2017; Liu et al., 2019a, 2016). As mentioned in Section 1, it is difficult to effectively learn the correlations between the ER and the RC tasks. To this end, we follow some of the ideas from other domains to dynamically learn the interactions between the two tasks, refining the classifiers of the tasks. To the best of our knowledge, this is the first work to model explicit interactions in a multi-task learning architecture for the joint extraction of entities and relations in text.

3 Problem Statement

In this section, we formally describe the joint entity and relation extraction problem. For a set $T = \{t_1, \cdots, t_l\}$ of pre-defined l relation types, and a given sentence $s = \{w_1, w_2, \cdots, w_n\}$ of n words, the problem is to extract all relational facts for the given sentence. In this paper, a single relational triple is of the form $\langle w_i, t, w_j \rangle$, where $w_i, w_j \in s$ are entity words or heads of multi-token entities, and $w_i \neq w_j$, and the relation $t \in T$. The goal is to predict the probability $y_{(i,j)}^t$ that the relational triple $\langle w_i, t, w_j \rangle$ is factual given the word pair (w_i, w_j). Besides, the entity recognition task can identify the head and tail words of multi-token entities for the extracted relational triple.

4 Model

In this section, we describe the recurrent interaction network (RIN) for extracting relational facts in text. The RIN model is composed of an entity recognition (ER) module and a relation classification (RC) module. We start by presenting an overview of the RIN model, showing the interaction between the ER and RC tasks. Next, we elaborate the ER and RC modules and define the training objective. The framework of RIN is shown in Figure 2.

4.1 Recurrent Interaction Network

The RIN model we propose uses a bidirectional LSTM network to learn correlations between the ER and the RC tasks, and derives shared features for the two tasks. We denote H as the output of shared features, where $H = \{h_1, h_2, \ldots, h_n\}$ corresponds to the representations of words in sentence s. A straightforward strategy for the joint ER and RC task is to pass H into independent ER or RC modules for predictions. Denote C_e as the ER module to identify and extract entities in the text, and C_r as the RC module to classify relational

triples in the text. Formally, Y_e and Y_r, the set of predictions of the entities and relational triples are formulated as:

$$Y_r = C_r(H)$$
$$Y_e = C_e(H) \tag{1}$$

where $Y_e =: \{y_i | h_i \in H\}$, $Y_r =: \{y_{(i,j)} | h_i, h_j \in H\}$, y_i is a probability distribution over BIOES labels (Fu et al., 2019), and $y_{(i,j)}$ is a probability distribution over the relation types $t \in T$. This structure is basically a conventional MTL method, where interactions are learned implicitly, impeding dynamic learning of intrinsic correlations between the two tasks.

To enhance the interaction between the two tasks, we dynamically learn the explicit interactions between the ER and RC tasks. Each layer of the RIN model is an interaction layer comprising of two separate gated recurrent units (GRUs), accounting for the ER task and the RC task. The GRU networks are designed to model task-specific features at the k-th layer, taking into account the previous shared features $H^{(k-1)}$ and the previous predictions Y_e^{k-1} and Y_r^{k-1}. Meanwhile, the shared features $H^{(k)}$ generated at the k-th layer is a sum of the previous task-specific features and the previous shared features $H^{(k-1)}$. Such a mechanism ensures that we retain the learned correlations as learning progresses along the network.

Let GRU_r and GRU_e denote the GRU networks for the relation classification and entity recognition modules in the interaction layer. Denote H_r^k and H_e^k the task-specific features modeled by the respective GRU_r and GRU_e networks at the k-th layer. Formally, the outputs H_r^k and H_e^k and shared features H^k at the k-th interaction layer is computed as follows:

$$H_r^k = \text{GRU}_r\left(Y_r^{k-1}, H^{k-1} | \theta_{\text{GRU}_r}\right)$$
$$H_e^k = \text{GRU}_e\left(Y_e^{k-1}, H^{k-1} | \theta_{\text{GRU}_e}\right) \tag{2}$$
$$H^k = H_r^k + H_e^k + H^{k-1}$$

where θ_{GRU_r} and θ_{GRU_e} are parameters for the GRU_r and GRU_e networks respectively. To take advantage of the previous learned explicit interactions in this network, we allow the network to have a minimum of two layers, i.e, $k = 2, 3 \ldots K$. Hence, for the ER task and RC tasks, the outputs at the k-th layer is formulated as:

$$Y_r^k = C_r(H_r^k)$$
$$Y_e^k = C_e(H_e^k) \tag{3}$$

(a) Framework of RIN.   (b) The ER module (C_e).   (c) The RC module (C_r).
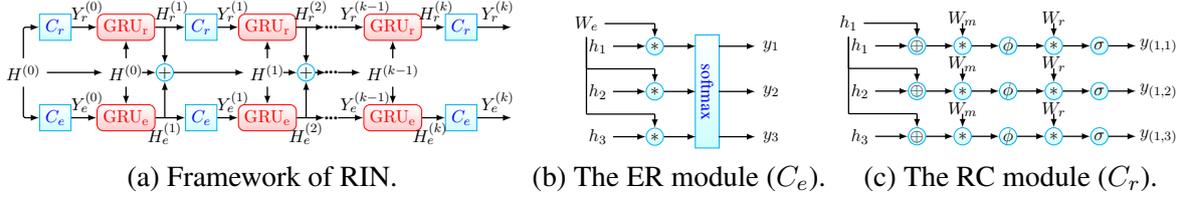
Figure 2: (a) The framework of RIN. (b) The entitiy recognition module. (c) The relation classification module. In (b) and (c), we use a toy example of shared features $H = \{h_1, h_2, h_3\}$ to demonstrate the entity prediction for word w_i and relation prediction for all pairs $(w_1, w_1), (w_1, w_2), (w_1, w_3)$. $+$, $\oplus$, $*$, ϕ, and σ denote a summation operator, a concatenation operator, a matrix multiplication, relu activation function and sigmoid function respectively.

4.1.1 The GRU network

In the RIN model, we proposed the GRU_r and GRU_e networks for the relation classification and entity recognition modules. Formally, for a single word w, the GRU_e network takes the output $y \in Y_e$ and the shared word representation $h \in H$ as inputs and computes the ER task feature vector $h_e \in H_e$. Formally, this can be formulated as:

$$
\begin{aligned}
z &= \sigma\left(W_{\mathrm{z}}(h \oplus y)\right) \\
u &= \sigma\left(W_{\mathrm{u}}(h \oplus y)\right) \\
\check{h} &= \tanh\left(W_{\mathrm{o}}((u * h) \oplus y)\right) \\
h_e &= (1 - z) * h + z * \check{h}
\end{aligned}
\tag{4}
$$

where $\oplus$ is a concatenation operator, W_z, W_u, W_o are learnable parameters of the GRU network. GRU_r follows the same architecture as GRU_e to compute the RC task feature vector $h_r \in H_r$ for word w. However, for a given word w_i, it considers $h_i \in H$ and the vector y_i, where y_i is modeled from the set of relation predictions for all word pairs containing w_i. We can define this set as $Y_r(w_i) =: \{y_{(i,j)} \in Y_r | w_j \in s\}$.

$$
y_i = \mathrm{maxpool}\left(Y_r(w_i)\right), \tag{5}
$$

where the function $\mathrm{maxpool}(\cdot)$ is a maxpool operation along the dimension.

4.2 Entity Recognition (ER)

The ER module C_e attempts to recognize all entities in the text based on the features H_e. As an entity may consist of multiple words, we formalize the ER task as tagging each word with an entity label, taking values from *(Begin, Inside, End, Single, Out)* using the BIOES tagging scheme (Fu et al., 2019). Specifically, the ER module classifies each word to one of the five label clusters. The probability distribution y of word w over these five clusters

is calculated based on the ER task feature vector h_e as follows:

$$
y = \mathrm{softmax}(W_{\mathrm{e}} h_e + b_{\mathrm{e}}), \tag{6}
$$

where $\theta_{\mathrm{ER}} = \{W_{\mathrm{e}}, b_{\mathrm{e}}\}$ are learnable model parameters.

4.3 Relation Classification (RC)

The RC module C_r makes an attempt to identify and extract relational facts from the sentence. Following (Fu et al., 2019), we classify all relations between pairs of words in the sentence based on the features H_r. Thus, the relation classification task is interpreted as a binary classification problem, where we identify the truth value of a relational triple $\langle w_i, t, w_j \rangle$ by classifying the word pair (w_i, w_j). The task can be regarded as learning the probability distribution $y_{(i,j)}$ for each word pair (w_i, w_j). The value $y_{(i,j)}$ is a probability distribution over the relation types $t \in T$. Thus, $y_{(i,j)}$ is a vector with size l, where each dimension is a probability $y_{(i,j)}^t$ of the relational triple $\langle w_i, t, w_j \rangle$ to be factual. We compute $y_{(i,j)}$ for each word pair (w_i, w_j) by performing the following steps:

$$
\begin{aligned}
m &= \phi\left(W_{\mathrm{m}}(h_i \oplus h_j)\right) \\
y_{(i,j)} &= \sigma\left(W_{\mathrm{r}} m + b_r\right)
\end{aligned}
\tag{7}
$$

where $h_i, h_j \in H_r$ are the RC task feature vectors for $w_i, w_j \in s$, $\oplus$ is a concatenation operation, $\phi(\cdot)$ is the ReLU activation function, $\sigma(\cdot)$ is the sigmoid activation function. $\theta_{\mathrm{RC}} = \{W_{\mathrm{m}}, W_{\mathrm{r}}, b_r\}$ are learnable model parameters. Instead of using a softmax function for classification, as used in (Fu et al., 2019), we find that the sigmoid function offers a natural way of identifying multiple relations that may exist between word pairs, solving the overlapping problem more efficiently.

4.4 Training Objective

The RIN model ultimately outputs task-specific representations, which are fed into their corresponding ER module and the RC module for predictions. As such, the training objective of RIN is comprised of two parts: the loss function for RC L_r and the loss function for ER L_e. The losses L_e and L_r are defined as

$$L_e(w) = \text{CrossEntropy}(\bar{y}, y)$$
$$L_r(\langle w_i, t, w_j \rangle) = \text{CrossEntropy}\left(\bar{y}^t_{(i,j)}, y^t_{(i,j)}\right)$$
(8)

where $\bar{y}$ and $\bar{y}^t_{(i,j)}$ are the respective ground truth values of word w and relational triple $\langle w_i, t, w_j \rangle$, and y and $y^t_{(i,j)}$ are the predictions from the ER module (C_e) and the RC module (C_r) at the K-th layer (i.e. the last layer) of RIN.

The total loss L over all words and relational triples for all sentences is then calculated as follows.

$$L = \sum_s \left(\sum_{w \in s} L_e(w) + \sum_{w_i, w_j \in s, t \in T} L_r(\langle w_i, t, w_j \rangle) \right) \quad (9)$$

With gradient based algorithm, we seek to minimize the total loss L over all model parameters $\Theta = \{\theta_{\text{GRU}_r}, \theta_{\text{GRU}_e}, \theta_{\text{RC}}, \theta_{\text{ER}}, \theta_{\text{H}}\}$ (θ_{H} is the parameters for the BiLSTM network) to achieve good performance for both the ER and RC tasks.

5 Experiment

We conduct experiments to evaluate RIN on two public datasets NYT (Riedel et al., 2010) and WebNLG (Gardent et al., 2017). The NYT dataset consists of 1.18M sentences with 24 predefined relation types. The WebNLG dataset was created by Natural Language Generation (NLG) tasks, and adapted by (Zeng et al., 2018) for a relational triple extraction task. We directly use the preprocessed datasets released by (Zeng et al., 2018) [1]. It is worth mentioning that only the tail word of an entity is marked in the preprocessed dataset released by (Zeng et al., 2018). To properly distinguish entities, we take a further step of tagging entities with the conventional BIOES tagging scheme as the one used in (Fu et al., 2019). We report Precision (Prec), Recall (Rec) and micro-F1 (F1) scores on our model and other recent models (Zeng et al., 2018, 2019b; Zheng et al., 2017; Fu et al., 2019;

Zeng et al., 2019a) for the *Partial Match* task and the *Exact Match* task. For our proposed method, we report the mean results over five runs using different random seeds, along with its standard deviation to show the stability of our results. The statistics of datasets are summarized in Table 2. Additional experiments on older datasets NYT10 (Riedel et al., 2010) and NYT11 (Hoffmann et al., 2011) are also performed, and the results are available in the supplementary file. Our results on these datasets show satisfactory performance, generally outperforming previous models on the NYT10 and NYT11.

5.1 Partial Match and Exact Match

Both NYT and WebLG datasets support evaluation for the *Partial Match* task and the *Exact Match* task. The Partial Match task only requires the relation and the heads of both subject and object entities of the extracted relational triple to be correct. For the Exact Match as recently adopted by (Zheng et al., 2017; Fu et al., 2019; Zeng et al., 2019a), the extracted relational triple is considered to be correct if the relation and the heads and tails of the subject and object entities are all correct. Thus, the extracted relational triple completely matches the gold relational triple.

5.2 Implementation Details

For a fair comparison with previous recent works (Zeng et al., 2018), we use the 100-dimensional Glove embedding (Pennington et al., 2014) to represent the word embeddings [2]. Part-of-speech (POS) tags are assigned to words using Stanford POS tagger [3]. We map each POS tag to a randomly initialized 10-dimensional POS embedding. We concatenate both word and POS embeddings as the input embeddings. For any given sentence, the input embeddings for words are fed to a BiLSTM network to learn a 100-dimensional embedding for each word. We improve learning by using dropout regularization in the input embeddings. The BiLSTM embeddings represent the shared features H in the RIN model. Our model is trained using an Adam optimizer (Kingma and Ba, 2014). The hyper-parameters are set empirically and manually tuned on the development set to select the best model. We implement our model using PyTorch on a Linux machine with a GPU device NVIDIA V100 NVLINK 32GB. Table 3

---

[1] https://github.com/xiangrongzeng/copy_re

[2] https://nlp.stanford.edu/projects/glove/
[3] https://stanfordnlp.github.io/CoreNLP/

Evaluation	Model	NYT			WebNLG		
		Prec	Rec	F1	Prec	Rec	F1
Partial Match	OneDecoder	59.4	53.1	56.0	32.2	28.9	30.5
	MultiDecoder	61.0	56.6	58.7	37.7	36.4	37.1
	OrderRL	77.9	67.2	72.1	63.3	59.9	61.6
	$RIN_{w/o\ interaction}$	83.9±0.6	83.1±0.6	83.5±0.2	84.9±0.6	86.3±0.8	85.6±0.3
	RIN	**87.2±0.2**	**87.3±0.3**	**87.3±0.1**	**87.6±0.1**	**87.0±0.9**	**87.3±0.4**
Exact Match	NovelTagging	62.4	31.7	42.0	52.5	19.3	28.3
	$GraphRel_{1p}$	62.9	57.3	60.0	42.3	39.2	40.7
	$GraphRel_{2p}$	63.9	60.0	61.9	44.7	41.1	42.9
	CopyMLT-One	72.7	69.2	70.9	57.8	60.1	58.9
	CopyMLT-Mul	75.7	68.7	72.0	58.0	54.9	56.4
	$RIN_{w/o\ interaction}$	77.4±1.1	76.4±0.7	76.9±0.3	75.0±1.1	73.3±0.7	74.2±0.3
	RIN	**83.9±0.5**	**85.5±0.5**	**84.7±0.4**	**77.3±0.7**	**76.8±1.0**	**77.0±0.2**

Table 1: Precision, Recall and F1 performance of different models on the datasets. Results for the compared models are retrieved from their original papers. We report the mean results over five runs and the standard deviation. The best performance is bold-typed.

Dataset	Train	Dev	Test
NYT	56195	5000	5000
WebNLG	5019	500	703

Table 2: Statistics of NYT and WebNLG

lists the hyper-parameters of RIN on the datasets. For the relation classification task, we threshold the probabilities of the prediction and return only the relations with probability values ≥ 0.5. The code for our model will be made available upon acceptance.

	Hyper-parameter	NYT	WebNLG
Partial Match	K	4	2
	d	0.1	0.1
	η	$1e^{-3}$	$5e^{-4}$
	bs	50	50
	epochs	100	150
Exact Match	K	7	3
	d	0.1	0.1
	η	$1e^{-3}$	$5e^{-4}$
	bs	50	50
	epochs	100	150

Table 3: Hyper-parameter settings of RIN on the datasets (K: number of interaction layers, d: dropout rate for input embeddings, η: learning rate, bs: batch size.)

5.3 Performance Comparison

We compare our model with several recent models based on the Partial Match and the Exact Match evaluation tasks. We also include a baseline model $RIN_{w/o\ interaction}$ which excludes the interaction network used in RIN. In $RIN_{w/o\ interaction}$, the shared features H modeled by BiLSTM network is directly passed into C_e and C_r for task-specific predictions. We also compare with several recent models, including the NovelTagging (Zheng

et al., 2017), sequence-to-sequence (seq2seq) models such as OneDecoder (Zeng et al., 2018), Multi-Decoder (Zeng et al., 2018), and OrderRL (Zeng et al., 2019b), and MTL-based methods Copy-MLT (Zeng et al., 2019a), and GraphRel (Fu et al., 2019).
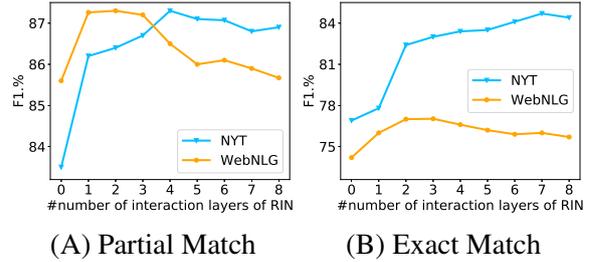

(A) Partial Match (B) Exact Match

Figure 3: Curves of F1 performance on different number of interaction layers K.

Partial Match Table 1 shows the performance of different models on the datasets. For the Partial Match evaluation task, it can first be noted that the small standard deviation for our model RIN and its ablated model $RIN_{w/o\ interaction}$ shows that our results are stable to an extent on the datasets. Even with the simple structure of $RIN_{w/o\ interaction}$, its results outperform the compared methods. In extracting relational facts, our model treats the Partial Match task as a relation classification problem. Whereas the compared methods take a seq2seq based approach to directly extract relational facts in the sentence. The results suggest that our approach may be more effective in identifying the relational facts for this task. It is more interesting to see the performance achieved by RIN. First of all, it can be noted that the model shows a level of stability due to its small standard deviation. Moreover, RIN shows a significant performance boost to the

RIN$_{\text{w/o interaction}}$ model, suggesting the importance of dynamically learning the explicit interactions between the ER task and the RC task.

Exact Match For the Exact Match task, we do not consider the methods (Zeng et al., 2018, 2019b), since these methods consider a seq2seq approach in extracting relational triples. Seq2seq methods are able to only decode a single word for an entity. Hence, they will inevitably fail to identify entities with multiple words.

In Table 1, we find that our ablated model RIN$_{\text{w/o interaction}}$ consistently outperforms previous models on the two datasets. In a more detailed analysis, we can note that the variants of the GraphRel model (Fu et al., 2019) consider the Exact Match task as a relation classification problem which classifies all word pairs in the sentence. In its relation classification module, it exploits a softmax function for the final classification. Hence, the model is not able to address cases where multiple relations exist between a pair of entities. We believe this explains why its results underperforms when compared to RIN$_{\text{w/o interaction}}$. Although CopyMLT and its variants (Zeng et al., 2019a) consider a seq2seq based approach to directly extract relational triples, its ER model can identify entities with multiple words and hence can address the Exact Match task. Nonetheless, it fails to outperform our model due to the fact that it uses a seq2seq based approach which we believe to be a more complex method for identifying relational triples. Besides, our main model RIN significantly outperforms RIN$_{\text{w/o interaction}}$ on the two datasets, further proving the importance of the explicit interactions learned between the ER and RC tasks.

5.4 Impact of K on the results

The hyper-parameter K is the number of interaction layers of the RIN model. Thus, K controls the number of times the RIN model attempts to learn explicit interactions between the ER and RC task. We conduct experiments to study the impact of K on the performance of RIN. We expect that the performance of the model increases as we learn more explicit interactions between the ER task and the RC task. Figure 3 shows the F1 curves of RIN on the datasets for increasing values of K. Here, at $K = 0$ the RIN model is reduced to RIN$_{\text{w/o interaction}}$.

We observe that as K increases the performance of RIN increases to an extent up to a point where it overfits. Taking a closer look at the performance on the Partial Match task, we find that RIN$_{\text{w/o interaction}}$ poorly models the interaction between the ER and RC task. By learning explicit interactions using the RIN model, we observe a sharp rise in performance at $K = 1$. On the Exact Match task we observe an interesting behaviour of RIN on the NYT and WebNLG dataset. Note that the 60% of entities on the WebNLG are multi-tokens, while 30% of the entities in the NYT dataset are multi-tokens. This means that the Exact Match task is more difficult on the WebNLG dataset, compared to the NYT dataset. As a consequence, RIN finds it difficult to learn explicit interactions on the WebNLG, while it learns much more easily on the NYT as K rises. We observe a sharp rise in performance from the first layer to the second layer on the NYT dataset. The second layer of RIN takes advantage of the original shared features H and the task-specific features of the first interaction layer. Thus, effective learning of the interaction between the two tasks takes place from the second layer. This explains the sharp rise in performance on the NYT dataset for the Exact Match task.

The results suggests that, to an extent, the proposed RIN model can dynamically filter additional interaction information between the two tasks along the multiple interaction layers.

5.5 Ablation Experiment

To examine the contributions of our main model components, we conduct ablation experiments on the NYT and WebNLG datasets. We use the default hyper-parameter settings for the ablated models (see Table 3). The ablated models are: (1) RIN$_{\text{w/o ER}}$: A RIN model which excludes the task-specific features H_e in the update of the shared features H, restricting the RC module from learning from the ER module. (2) RIN$_{\text{w/o RC}}$: A RIN model which excludes the task-specific features H_r in the update of the shared features H, restricting the ER module from learning from the RC module. (3) RIN$_{\text{w/o POS}}$: A RIN which only uses the Glove word embeddings as the input embeddings. We also include the ablated model RIN$_{\text{w/o interaction}}$. Table 5 shows the results for the experiment.

We find that the performance of RIN deteriorates as we remove critical components. Among the ablated models designed, RIN$_{\text{w/o interaction}}$ performs very poorly on the two datasets, suggesting the importance of learning explicit interactions dy-

Case1: A cult of victimology arose and was happily exploited by clever radicals among Europes Muslims, especially certain religious leaders like Imam Ahmad Abu Laban in Denmark and Mullah Krekar in Norway.	**Golden**:*Europe, Denmark, Norway* *(Europe, /location/location/contains, Denmark)* *(Europe, /location/location/contains, Norway)* **RIN**$_{\text{w/o interaction}}$: *Europe, Denmark, Norway* *(Europe, /location/location/contains, Denmark)* **RIN**: *Europe, Denmark, Norway* *(Europe, /location/location/contains, Denmark)* ***(Europe, /location/location/contains, Norway)***
Case2: Scott (No rating , 75 minutes) Engulfed by nightmares, blackouts and the anxieties of the age, a Texas woman flees homeland insecurity for a New York vision quest in this acute, resourceful and bracingly ambitious debut film.	**Golden**: *Scott, New York* *(York, /location/location/contains, Scott)* **RIN**$_{\text{w/o interaction}}$: ***Texas**, New York* *(York, /location/location/contains, Scott)* **RIN**: *Scott, New York* *(York, /location/location/contains, Scott)*

Table 4: Case study for RIN and RIN$_{\text{w/o interaction}}$. Entities and relational triples are in blue and orange texts respectively. We mark a wrong prediction with a red text.

Model	NYT	WebNLG
RIN	84.7	77.0
RIN$_{\text{w/o ER}}$	83.9	76.4
RIN$_{\text{w/o RC}}$	77.3	76.0
RIN$_{\text{w/o interaction}}$	76.9	74.2
RIN$_{\text{w/o POS}}$	84.1	76.6

Table 5: F1 performance of different ablation models on the datasets. The Exact Match evaluation is used.

namically between the ER and RC tasks. We also find that RIN$_{\text{w/o ER}}$ marginally underperforms the RIN model, and also showing a better performance when compared to RIN$_{\text{w/o RC}}$. The results suggest that the performance of RIN is heavily dependent on the ER module exploiting information from the RC module. Lastly, the results for RIN$_{\text{w/o POS}}$ suggest that the POS tags does not significantly boost the performance of RIN.

5.6 Case Study

We present two case examples from NYT dataset as illustrations to observe the behaviour of the RIN and RIN$_{\text{w/o interaction}}$ models. Table 4 shows the results of the study. In the first case example, both RIN and RIN$_{\text{w/o interaction}}$ correctly extracts all the gold entities in the sentence. But, RIN$_{\text{w/o interaction}}$ captures only the gold relational triple (Europe, /location/location/contains, Denmark), and misses the gold triple (Europe, /location/location/contains, Norway). Given the fact that (Europe, /location/location/contains, Norway) overlaps a relational fat, it is important to dynamically learn to capture the complex interaction between the ER and RC tasks. The RIN model takes advantage of its interaction network to identify both gold triples.

In the second case, we observe that both RIN and RIN$_{\text{w/o interaction}}$ correctly extract the relational triple (York, /location/location/contains, Scott). However, RIN$_{\text{w/o interaction}}$ identifies Texas as an entity by error while RIN correctly extracts the entity Scott and New York. The results suggest that RIN is able to leverage information from the RC module to correctly identify entities in the ER module. It is worth noting that we can easily complete the entity York in the extracted relational triple by aligning it to the extracted entity New York.

6 Conclusion

In this paper, we tackle the weakness of existing MTL-based methods proposed for the joint extraction of entities and relation in unstructured text. Specifically, these methods assume that a shared network is sufficient to capture the correlations between the entity recognition task and the relation classification task, and that the shared features derived from this network can be passed into models for the task-specific tasks to make predictions independently. Instead, we show that dynamically learning the interactions between the tasks may capture complex correlations which improves the task-specific feature for classification. We proposed multi-task learning model which allows explicit interactions to be dynamically learned among the subtasks. Our experiments on benchmark datasets validates clear advantage over the existing proposed methods. We note that our model can be adapted for other NLP tasks, including aspect level sentiment classification and slot filling. As future work, we intend to explore its application in those fields.

Acknowledgment

References

Yee Seng Chan and Dan Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 551–560.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: deep neural networks with multitask learning. In *Machine Learning, Proceedings of the Twenty-Fifth International Conference (ICML 2008), Helsinki, Finland, June 5-9, 2008*, pages 160–167.

Dai Dai, Xinyan Xiao, Yajuan Lyu, Shan Dou, Qiaoqiao She, and Haifeng Wang. 2019. Joint extraction of entities and overlapping relations using position-attentive sequence labeling. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6300–6308.

Verna Dankers, Marek Rei, Martha Lewis, and Ekaterina Shutova. 2019. Modelling the interplay of metaphor and emotion through multitask learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2218–2229.

Yang Deng, Yuexiang Xie, Yaliang Li, Min Yang, Nan Du, Wei Fan, Kai Lei, and Ying Shen. 2019. Multitask learning with multi-view attention for answer selection and knowledge base question answering. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6318–6325.

Tsu-Jui Fu, Peng-Hsuan Li, and Wei-Yun Ma. 2019. Graphrel: Modeling text as relational graphs for joint entity and relation extraction. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1409–1418.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017. Creating training corpora for NLG micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 179–188.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 753–757.

Rujun Han, Qiang Ning, and Nanyun Peng. 2019. Joint event and temporal relation extraction with shared representations and structured prediction. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 434–444.

Ruidan He, Wee Sun Lee, and Hwee Tou Ng and' Daniel Dahlmeier. 2019. An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 504–515.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke S. Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-based weak supervision for information extraction of overlapping relations. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA*, pages 541–550.

Minghao Hu, Yuxing Peng, Zhen Huang, Dongsheng Li, and Yiwei Lv. 2019. Open-domain targeted sentiment analysis via span-based extraction and classification. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 537–546.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Alexandros Komninos and Suresh Manandhar. 2017. Feature-rich networks for knowledge base completion. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 324–329.

Man Lan, Jianxiang Wang, Yuanbin Wu, Zheng-Yu Niu, and Haifeng Wang. 2017. Multi-task attention-based neural networks for implicit discourse relationship representation and identification. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1299–1308.

Xin Li, Lidong Bing, Wenxuan Zhang, and Wai Lam. 2019. Exploiting BERT for end-to-end aspect-based sentiment analysis. In *Proceedings of the 5th Workshop on Noisy User-generated Text, W-NUT@EMNLP 2019, Hong Kong, China, November 4, 2019*, pages 34–41.

Pengfei Liu, Jie Fu, Yue Dong, Xipeng Qiu, and Jackie Chi Kit Cheung. 2019a. Learning multi-task communication with message passing for sequence learning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 4360–4367.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2873–2879.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019b. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4487–4496.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Deepak Nathani, Jatin Chauhan, Charu Sharma, and Manohar Kaul. 2019. Learning attention-based embeddings for relation prediction in knowledge graphs. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4710–4723.

Toru Nishino, Shotaro Misawa, Ryuji Kano, Tomoki Taniguchi, Yasuhide Miura, and Tomoko Ohkuma. 2019. Keeping consistency of sentence generation and document classification with multi-task learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3193–3203.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

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

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu, and Minlie Huang. 2019. A hierarchical framework for relation extraction with reinforcement learning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7072–7079.

Changxing Wu, Xiaodong Shi, Yidong Chen, Yanzhou Huang, and Jinsong Su. 2016. Bilingually-constrained synthetic data for implicit discourse relation recognition. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2306–2312.

Bowen Yu, Zhenyu Zhang, Jianlin Su, Yubin Wang, Tingwen Liu, Bin Wang, and Sujian Li. 2019. Joint extraction of entities and relations based on a novel decomposition strategy. *CoRR*, abs/1909.04273.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *J. Mach. Learn. Res.*, 3:1083–1106.

Daojian Zeng, Haoran Zhang, and Qianying Liu. 2019a. Copymtl: Copy mechanism for joint extraction of entities and relations with multi-task learning.

Xiangrong Zeng, Shizhu He, Daojian Zeng, Kang Liu, Shengping Liu, and Jun Zhao. 2019b. Learning the extraction order of multiple relational facts in a sentence with reinforcement learning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 367–377, Hong Kong, China. Association for Computational Linguistics.

Xiangrong Zeng, Daojian Zeng, Shizhu He, Kang Liu, and Jun Zhao. 2018. Extracting relational facts by an end-to-end neural model with copy mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 506–514.

Xiaodong Zhang and Houfeng Wang. 2016. A joint model of intent determination and slot filling for spoken language understanding. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, IJCAI 2016, New York, NY, USA, 9-15 July 2016*, pages 2993–2999.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2205–2215.

Sendong Zhao, Ting Liu, Sicheng Zhao, and Fei Wang. 2019. A neural multi-task learning framework to jointly model medical named entity recognition and normalization. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 817–824.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1227–1236.

Guodong Zhou, Jian Su, Jie Zhang, and Min Zhang. 2005. Exploring various knowledge in relation extraction. In *ACL 2005, 43rd Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference, 25-30 June 2005, University of Michigan, USA*, pages 427–434.