Entity-Relation Extraction as Multi-turn Question Answering

Xiaoya Li[*♠], Fan Yin[*♠], Zijun Sun[*◇♠], Xiayu Li[♠]
Arianna Yuan[♠,♡], Duo Chai[♠], Mingxin Zhou[♠] and Jiwei Li[♠,♣]

[♣] School of Information, Renmin University of China
[◇]Computer Center, Peking University
[♡] Computer Science Department, Stanford University
[♠] Shannon.AI
{xiaoya_li, fan_yin, zijun_sun, xiayu_li, duo_chai, mingxin_zhou, jiwei_li}@shannonai.com
xfyuan@stanford.edu

Abstract

In this paper, we propose a new paradigm for the task of entity-relation extraction. We cast the task as a multi-turn question answering problem, i.e., the extraction of entities and relations is transformed to the task of identifying answer spans from the context. This multi-turn QA formalization comes with several key advantages: firstly, the question query encodes important information for the entity/relation class we want to identify; secondly, QA provides a natural way of jointly modeling entity and relation; and thirdly, it allows us to exploit the well developed machine reading comprehension (MRC) models.

Experiments on the ACE and the CoNLL04 corpora demonstrate that the proposed paradigm significantly outperforms previous best models. We are able to obtain the state-of-the-art results on all of the ACE04, ACE05 and CoNLL04 datasets, increasing the SOTA results on the three datasets to 49.4 (+1.0), 60.2 (+0.6) and 68.9 (+2.1), respectively.

Additionally, we construct a newly developed dataset RESUME in Chinese, which requires multi-step reasoning to construct entity dependencies, as opposed to the single-step dependency extraction in the triplet exaction in previous datasets. The proposed multi-turn QA model also achieves the best performance on the RESUME dataset. [1]

1 Introduction

Identifying entities and their relations is the prerequisite of extracting structured knowledge from unstructured raw texts, which has recieved growing interest these years. Given a chunk of natural language text, the goal of entity-relation extraction is to transform it to a structural knowledge base. For example, given the following text:

[1]* indicates equal contribution.

Person	Corp	Time	Position
Musk	SpaceX	2002	CEO
Musk	Tesla	2003	CEO& product architect
Musk	SolarCity	2006	chairman
Musk	Neuralink	2016	CEO
Musk	The Boring Company	2016	-

Table 1: An illustration of an extracted structural table.

In 2002, Musk founded SpaceX, an aerospace manufacturer and space transport services Company, of which he is CEO and lead designer. He helped fund Tesla, Inc., an electric vehicle and solar panel manufacturer, in 2003, and became its CEO and product architect. In 2006, he inspired the creation of SolarCity, a solar energy services Company, and operates as its chairman. In 2016, he co-founded Neuralink, a neurotechnology Company focused on developing brain–computer interfaces, and is its CEO. In 2016, Musk founded The Boring Company, an infrastructure and tunnel-construction Company.

We need to extract four different types of entities, i.e., Person, Company, Time and Position, and three types of relations, FOUND, FOUNDING-TIME and SERVING-ROLE. The text is to be transformed into a structural dataset shown in Table 1.

Most existing models approach this task by extracting a list of triples from the text, i.e., $\text{REL}(e_1, e_2)$, which denotes that relation REL holds between entity e_1 and entity e_2. Previous models fall into two major categories: the pipelined approach, which first uses tagging models to identify entities, and then uses relation extraction models to identify the relation between each entity pair; and the joint approach, which combines the entity model and the relation model throught different strategies, such as constraints or parameters sharing.

There are several key issues with current approaches, both in terms of the task formalization

and the algorithm. At the formalization level, the REL(e_1, e_2) triplet structure is not enough to fully express the data structure behind the text. Take the *Musk* case as an example, there is a hierarchical dependency between the tags: the extraction of Time depends on Position since a Person can hold multiple Positions in a Company during different Time periods. The extraction of Position also depends on Company since a Person can work for multiple companies. At the algorithm level, for most existing relation extraction models (Miwa and Bansal, 2016; Wang et al., 2016a; Ye et al., 2016), the input to the model is a raw sentence with two marked mentions, and the output is whether a relation holds between the two mentions. As pointed out in Wang et al. (2016a); Zeng et al. (2018), it is hard for neural models to capture all the lexical, semantic and syntactic cues in this formalization, especially when (1) entities are far away; (2) one entity is involved in multiple triplets; or (3) relation spans have overlaps[2].

In the paper, we propose a new paradigm to handle the task of entity-relation extraction. We formalize the task as a multi-turn question answering task: each entity type and relation type is characterized by a question answering template, and entities and relations are extracted by answering template questions. Answers are text spans, extracted using the now standard machine reading comprehension (MRC) framework: predicting answer spans given context (Seo et al., 2016; Wang and Jiang, 2016; Xiong et al., 2017; Wang et al., 2016b). To extract structural data like Table 1, the model need to answer the following questions sequentially:

- *Q: who is mentioned in the text? A: Musk*;
- *Q: which Company / companies did Musk work for? A: SpaceX, Tesla, SolarCity, Neuralink and The Boring Company*;
- *Q: when did Musk join SpaceX? A: 2002*;
- *Q: what was Musk's Position in SpaceX? A: CEO*.

Treating the entity-relation extraction task as a multi-turn QA task has the following key advantages: (1) the multi-turn QA setting provides an elegant way to capture the hierarchical dependency of tags. As the multi-turn QA proceeds, we progressively obtain the entities we need for the next turn. This is closely akin to the multi-turn slot filling dialogue system (Williams and Young, 2005; Lemon et al., 2006); (2) the question query encodes important prior information for the relation class we want to identify. This informativeness can potentially solve the issues that existing relation extraction models fail to solve, such as distantly-separated entity pairs, relation span overlap, etc; (3) the QA framework provides a natural way to simultaneously extract entities and relations: most MRC models support outputting special NONE tokens, indicating that there is no answer to the question. Throught this, the original two tasks, entity extraction and relation extraction can be merged to a single QA task: a relation holds if the returned answer to the question corresponding to that relation is not NONE, and this returned answer is the entity that we wish to extract.

In this paper, we show that the proposed paradigm, which transforms the entity-relation extraction task to a multi-turn QA task, introduces significant performance boost over existing systems. It achieves state-of-the-art (SOTA) performance on the ACE and the CoNLL04 datasets. The tasks on these datasets are formalized as triplet extraction problems, in which two turns of QA suffice. We thus build a more complicated and more difficult dataset called RESUME which requires to extract biographical information of individuals from raw texts. The construction of structural knowledge base from RESUME requires four or five turns of QA. We also show that this multi-turn QA setting could easilty integrate reinforcement learning (just as in multi-turn dialog systems) to gain additional performance boost.

The rest of this paper is organized as follows: Section 2 details related work. We describe the dataset and setting in Section 3, the proposed model in Section 4, and experimental results in Section 5. We conclude this paper in Section 6.

2 Related Work

2.1 Extracting Entities and Relations

Many earlier entity-relation extraction systems are pipelined (Zelenko et al., 2003; Miwa et al., 2009; Chan and Roth, 2011; Lin et al., 2016): an entity extraction model first identifies entities of interest and a relation extraction model then constructs relations between the extracted entities. Although pipelined systems has the flexibility of integrating different data sources and learning algorithms, they suffer significantly from error propagation.

To tackle this issue, joint learning models have been proposed. Earlier joint learning approaches connect the two models through various dependen-

[2] e.g., in text *A B C D*, (A, C) is a pair and (B, D) is a pair.

cies, including constraints solved by integer linear programming (Yang and Cardie, 2013; Roth and Yih, 2007), card-pyramid parsing (Kate and Mooney, 2010), and global probabilistic graphical models (Yu and Lam, 2010; Singh et al., 2013). In later studies, Li and Ji (2014) extract entity mentions and relations using structured perceptron with efficient beam-search, which is significantly more efficient and less Time-consuming than constraint-based approaches. Miwa and Sasaki (2014); Gupta et al. (2016); Zhang et al. (2017) proposed the table-filling approach, which provides an opportunity to incorporating more sophisticated features and algorithms into the model, such as search orders in decoding and global features. Neural network models have been widely used in the literature as well. Miwa and Bansal (2016) introduced an end-to-end approach that extract entities and their relations using neural network models with shared parameters, i.e., extracting entities using a neural tagging model and extracting relations using a neural multi-class classification model based on tree LSTMs (Tai et al., 2015). Wang et al. (2016a) extract relations using multi-level attention CNNs. Zeng et al. (2018) proposed a new framework that uses sequence-to-sequence models to generate entity-relation triples, naturally combining entity detection and relation detection.

Another way to bind the entity and the relation extraction models is to use reinforcement learning or Minimum Risk Training, in which the training signals are given based on the joint decision by the two models. Sun et al. (2018) optimized a global loss function to jointly train the two models under the framework work of Minimum Risk Training. Takanobu et al. (2018) used hierarchical reinforcement learning to extract entities and relations in a hierarchical manner.

2.2 Machine Reading Comprehension

Main-stream MRC models (Seo et al., 2016; Wang and Jiang, 2016; Xiong et al., 2017; Wang et al., 2016b) extract text spans in passages given queries. Text span extraction can be simplified to two multi-class classification tasks, i.e., predicting the starting and the ending positions of the answer. Similar strategy can be extended to multi-passage MRC (Joshi et al., 2017; Dunn et al., 2017) where the answer needs to be selected from multiple passages. Multi-passage MRC tasks can be easily simplified to single-passage MRC tasks by concatenating passages (Shen et al., 2017; Wang et al., 2017b). Wang et al. (2017a) first rank the passages and then run single-passage MRC on the selected passage. Tan et al. (2017) train the passage ranking model jointly with the reading comprehension model. Pretraining methods like BERT (Devlin et al., 2018) or Elmo (Peters et al., 2018) have proved to be extremely helpful in MRC tasks.

There has been a tendency of casting non-QA NLP tasks as QA tasks (McCann et al., 2018). Our work is highly inspired by Levy et al. (2017). Levy et al. (2017) and McCann et al. (2018) focus on identifying the relation between two pre-defined entities and the authors formalize the task of relation extraction as a single-turn QA task. In the current paper we study a more complicated scenario, where hierarchical tag dependency needs to be modeled and single-turn QA approach no longer suffices. We show that our multi-turn QA method is able to solve this challenge and obtain new state-of-the-art results.

3 Datasets and Tasks

3.1 ACE04, ACE05 and CoNLL04

We use ACE04, ACE05 and CoNLL04 (Roth and Yih, 2004), the widely used entity-relation extraction benchmarks for evaluation. ACE04 defines 7 entity types, including Person (PER), Organization (ORG), Geographical Entities (GPE), Location (loc), Facility (FAC), Weapon (WEA) and Vehicle (VEH). For each pair of entities, it defines 7 relation categories, including Physical (PHYS), Person-Social (PER-SOC), Employment-Organization (EMP-ORG), Agent-Artifact (ART), PER/ORG Affiliation (OTHER-AFF), GPE- Affiliation (GPE-AFF) and Discourse (DISC). ACE05 was built upon ACE04. It kept the PER-SOC, ART and GPE-AFF categories from ACE04 but split PHYS into PHYS and a new relation category PART-WHOLE. It also deleted DISC and merged EMP-ORG and OTHER-AFF into a new category EMP-ORG. As for CoNLL04, it defines four entity types (LOC, ORG, PERand OTHERS) and five relation categories (LOCATED_IN, WORK_FOR, ORGBASED_IN, LIVE_IN]and KILL).

For ACE04 and ACE05, we followed the training/dev/test split in Li and Ji (2014) and Miwa and Bansal (2016)[3]. For the CoNLL04 dataset, we followed Miwa and Sasaki (2014).

[3] https://github.com/tticoin/LSTM-ER/.

3.2 RESUME: A newly constructed dataset

The ACE and the CoNLL-04 datasets are intended for triplet extraction, and two turns of QA is sufficient to extract the triplet (one turn for head-entities and another for joint extraction of tail-entities and relations). These datasets do not involve hierarchical entity relations as in our previous *Musk* example, which are prevalent in real life applications.

Therefore, we construct a new dataset called RESUME. We extracted 841 paragraphs from chapters describing management teams in IPO prospectuses. Each paragraph describes some work history of an executive. We wish to extract the structural data from the resume. The dataset is in Chinese. The following shows an examples:

郑强先生，本公司监事，1973年出生，中国国籍，无境外永久居留权。1995年，毕业于南京大学经济管理专业；1995年至1998年，就职于江苏常州公路运输有限公司，任主办会计；1998年至2000年，就职于越秀会计师事务所，任项目经理；2000年至2010年，就职于国富浩华会计师事务所有限公司广东分所，历任项目经理、部门经理、合伙人及副主任会计师；2010年至2011年，就职于广东中科招商创业投资管理有限责任公司，任副总经理；2011年至今，任广东中广投资管理有限公司董事、总经理；2016年至今，任湛江中广创业投资有限公司董事、总经理；2016年3月至今，担任本公司监事.

Mr. Zheng Qiang, a supervisor of the Company. He was born in 1973. His nationality is Chinese with no permanent residency abroad. He graduated from Nanjing University with a major in economic management in 1995. From 1995 to 1998, he worked for Jiangsu Changzhou Road Transportation Co., Ltd. as an organizer of accounting. From 1998 to 2000, he worked as a project manager in Yuexiu Certified Public Accountants. In 2010, he worked in the Guangdong branch of Guofu Haohua Certified Public Accountants Co., Ltd., and served as a project manager, department manager, partner and deputy chief accountant. From 2010 to 2011, he worked for Guangdong Zhongke Investment Venture Capital Management Co., Ltd. as a deputy general manager; since 2011, he has served as thedirector and general manager of Guangdong Zhongguang Investment Management Co., Ltd.; since 2016, he has served as director and general manager of Zhanjiang Zhongguang Venture Capital Co., Ltd.; since March 2016, he has served as the supervisor of the Company.

We identify four types of entities: Person (the name of the executive), Company (the company that the executive works/worked for), Position (the position that he/she holds/held) and Time (the time period that the executive occupies/occupied that position). It is worth noting that one person can work for different companies during different periods of time and that one person can hold different positions in different periods of time for the same company.

	Total #	Average # per passage
Person	961	1.09
Company	1988	2.13
Position	2687	1.33
Time	1275	1.01

Table 2: Statistics for the RESUME dataset.

We recruited crowdworkers to fill the slots in Table 1. Each passage is labeled by two different crowdworkers. If labels from the two annotators disagree, one or more annotators were asked to label the sentence and a majority vote was taken as the final decision. Since the wording of the text is usually very explicit and formal, the inter-agreement between annotators is very high, achieving a value of 93.5% for all slots. Some statistics of the dataset are shown in Table 2. We randomly split the dataset into training (80%), validation(10%) and test set (10%).

4 Model

4.1 System Overview

The overview of the algorithm is shown in Algorithm 1. The algorithm contains two stages:

(1) The head-entity extraction stage (line 4-9): each episode of multi-turn QA is triggered by an entity. To extract this starting entity, we transform each entity type to a question using EntityQuesTemplates (line 4) and the entity e is extracted by answering the question (line 5). If the system outputs the special NONE token, then it means s does not contain any entity of that type.

(2) The relation and the tail-entity extraction stage (line 10-24): ChainOfRelTemplates defines a chain of relations, the order of which we need to follow to run multi-turn QA. The reason is that the extraction of some entities depends on the extraction of others. For example, in the RESUME dataset, the position held by an executive relies on the company he works for. Also the extraction of the Time entity relies on the extraction of both the Company and the Position. The extraction order is manually pre-defined. ChainOfRelTemplates also

Relation Type	head-e	tail-e	Natural Language Question & Template Question
GEN-AFF	FAC	GPE	find a geo-political entity that connects to XXX XXX; has affiliation; geo-political entity
PART-WHOLE	FAC	FAC	find a facility that geographically relates to XXX XXX; part whole; facility
PART-WHOLE	FAC	GPE	find a geo-political entity that geographically relates to XXX XXX; part whole; geo-political entity
PART-WHOLE	FAC	VEH	find a vehicle that belongs to XXX XXX; part whole; vehicle
PHYS	FAC	FAC	find a facility near XXX? XXX; physical; facility
ART	GPE	FAC	find a facility which is made by XXX XXX; agent artifact; facility
ART	GPE	VEH	find a vehicle which is owned or used by XXX XXX; agent artifact; vehicle
ART	GPE	WEA	find a weapon which is owned or used by XXX XXX; agent artifact; weapon
ORG-AFF	GPE	ORG	find an organization which is invested by XXX XXX; organization affiliation; organization
PART-WHOLE	GPE	GPE	find a geo political entity which is controlled by XXX XXX; part whole; geo-political entity
PART-WHOLE	GPE	LOC	find a location geographically related to XXX XXX; part whole; location

Table 3: Some of the question templates for different relation types in AEC.

Q1 Person:	who is mentioned in the text?	A: e_1
Q2 Company:	which companies did e_1 work for?	A: e_2
Q3 Position:	what was e_1's position in e_2?	A: e_3
Q4 Time:	During which period did e_1 work for e_2 as e_3	A: e_4

Table 4: Question templates for the RESUME dataset.

defines the template for each relation. Each template contains some slots to be filled. To generate a question (line 14), we insert previously extracted entity/entities to the slot/slots in a template. The relation REL and tail-entity e will be jointly extracted by answering the generated question (line 15). A returned NONE token indicates that there is no answer in the given sentence.

It is worth noting that entities extracted from the head-entity extraction stage may not all be head entities. In the subsequent relation and tail-entity extraction stage, extracted entities from the first stage are initially assumed to be head entities, and are fed to the templates to generate questions. If an entity e extracted from the first stage is indeed a head-entity of a relation, then the QA model will extract the tail-entity by answering the corresponding question. Otherwise, the answer will be NONE and thus ignored.

For ACE04, ACE05 and CoNLL04 datasets, only two QA turns are needed. ChainOfRelTemplates thus only contain chains of 1. For RESUME, we need to extract 4 entities, so ChainOfRelTemplates contain chains of 3.

4.2 Generating Questions using Templates

Each entity type is associated with a type-specific question generated by the templates. There are two ways to generate questions based on templates: natural language questions or pseudo-questions. A pseudo-question is not necessarily grammatical. For example, the natural language question for the Facility type could be *Which facility is mentioned in the text*, and the pseudo-question could just be *entity: facility*.

At the relation and the tail-entity joint extraction stage, a question is generated by combing a relation-specific template with the extracted head-entity. The question could be either a natural language question or a pseudo-question. Examples are shown in Table 3 and Table 4.

4.3 Extracting Answer Spans via MRC

Various MRC models have been proposed, such as BiDAF (Seo et al., 2016) and QANet (Yu et al., 2018). In the standard MRC setting, given a question $Q = \{q_1, q_2, ..., q_{N_q}\}$ where N_q denotes the number of words in Q, and context $C = \{c_1, c_2, ..., c_{N_c}\}$, where N_c denotes the num-

Input: sentence s, EntityQuesTemplates, ChainOfRelTemplates
Output: a list of list (table) M = []

$M \leftarrow \emptyset$
HeadEntList$\leftarrow \emptyset$
for entity_question in EntityQuesTemplates **do**
  e_1 = Extract_Answer(entity_question, s)
  if $e_1 \neq$ NONE **do**
    HeadEntList = HeadEntList + $\{e_1\}$
  endif
end for
for head_entity in HeadEntList **do**
  ent_list = [head_entity]
  for [rel, rel_temp] in ChainOfRelTemplates **do**
    for (rel, rel_temp) in List of [rel, rel_temp] **do**
      q = GenQues(rel_temp, rel, ent_list)
      e = Extract_Answer(rel_question, s)
      if $e \neq$ NONE
        ent_list = ent_list + e
      endif
    end for
  end for
  if len(ent_list)=len([rel, rel_temp])
    M = M + ent_list
  endif
end for
return M

Algorithm 1: Transforming the entity-relation extraction task to a multi-turn QA task.

ber of words in C, we need to predict the answer span. For the QA framework, we use BERT (Devlin et al., 2018) as a backbone. BERT performs bidirectional language model pretraining on large-scale datasets using transformers (Vaswani et al., 2017) and achieves SOTA results on MRC datasets like SQUAD (Rajpurkar et al., 2016). To align with the BERT framework, the question Q and the context C are combined by concatenating the list [CLS, Q, SEP, C, SEP], where CLS and SEP are special tokens, Q is the tokenized question and C is the context. The representation of each context token is obtained using multi-layer transformers.

Traditional MRC models (Wang and Jiang, 2016; Xiong et al., 2017) predict the starting and ending indices by applying two softmax layers to the context tokens. This softmax-based span extraction strategy only fits for single-answer extraction tasks, but not for our task, since one sentence/passage in our setting might contain multiple answers. To tackle this issue, we formalize the task as a query-based tagging problem (Lafferty et al., 2001; Huang et al., 2015; Ma and Hovy, 2016). Specially, we predict a BMEO (beginning, inside, ending and outside) label for each token in the context given the query. The representation of each word is fed to a softmax layer to output a BMEO label. One can think that we are transforming two N-class classification tasks of predicting the starting and the ending indices (where N denotes the length of sentence) to N 5-class classification tasks[4].

Training and Test At the training time, we jointly train the objectives for the two stages:

$$\mathcal{L} = (1-\lambda)\mathcal{L}(\text{head-entity}) + \lambda\mathcal{L}(\text{tail-entity, rel}) \quad (1)$$

$\lambda \in [0, 1]$ is the parameter controling the trade-off between the two objectives. Its value is tuned on the validation set. Both the two models are initialized using the standard BERT model and they share parameters during the training. At test time, head-entities and tail-entities are extracted separately based on the two objectives.

4.4 Reinforcement Learning

Note that in our setting, the extracted answer from one turn not only affects its own accuracy, but also determines how a question will be constructed for the downstream turns, which in turn affect later accuracies. We decide to use reinforcement learning to tackle it, which has been proved to be successful in multi-turn dialogue generation (Mrkšić et al., 2015; Li et al., 2016a; Wen et al., 2016), a task that has the same challenge as ours.

Action and Policy In a RL setting, we need to define action and policy. In the multi-turn QA setting, the action is selecting a text span in each turn. The policy defines the probability of selecting a certain span given the question and the context. As the algorithm relies on the BMEO tagging output, the probability of selecting a certain span $\{w_1, w_2, ..., w_n\}$ is the joint probability of w_1 being assigned to B (beginning), $w_2, ..., w_{n-1}$ being assigned to M (inside) and w_n being assigned to E (end), written as follows:

$$\begin{aligned} &p(y(w_1, ..., w_n) = \text{answer}|\text{question}, s) \\ &= p(w_1 = \text{B}) \times p(w_n = \text{E}) \prod_{i \in [2, n-1]} p(w_i = \text{M}) \end{aligned} \quad (2)$$

Reward For a given sentence s, we use the number of correctly retrieved triples as rewards. We use the REINFORCE algorithm (Williams, 1992), a kind of policy gradient method, to find the optimal policy, which maximizes the expected reward

[4] For some of the relations that we are interested in, their corresponding questions have single answers. We tried the strategy of predicting the starting and the ending index and found the results no different from the ones in the multi-answer QA-based tagging setting.

$E_\pi[R(w)]$. The expectation is approximated by sampling from the policy π and the gradient is computed using the likelihood ratio:

$$\nabla E(\theta) \approx [R(w) - b]\nabla \log \pi(y(w)|\text{question s})) \quad (3)$$

where b denotes a baseline value. For each turn in the multi-turn QA setting, getting an answer correct leads to a reward of +1 . The final reward is the accumulative reward of all turns. The baseline value is set to the average of all previous rewards. We do not initialize policy networks from scratch, but use the pre-trained head-entity and tail-entity extraction model described in the previous section. We also use the experience replay strategy (Mnih et al., 2015): for each batch, half of the examples are simulated and the other half is randomly selected from previously generated examples.

For the RESUME dataset, we use the strategy of curriculum learning (Bengio et al., 2009), i.e., we gradually increase the number of turns from 2 to 4 at training.

5 Experimental Results

5.1 Results on RESUME

Answers are extracted according to the order of Person (first-turn), Company (second-turn), Position (third-turn) and Time (forth-turn), and the extraction of each answer depends on those prior to them.

For baselines, we first implement a joint model in which entity extraction and relation extraction are trained together (denoted by *tagging+relation*). As in Zheng et al. (2017), entities are extracted using BERT tagging models, and relations are extracted by applying a CNN to representations output by BERT transformers.

Existing baselines which involve entity and relation identification stages (either pipelined or joint) are well suited for triplet extractions, but not really tailored to our setting because in the third and forth turn, we need more information to decide the relation than just the two entities. For instance, to extract Position, we need both Person and Company, and to extract Time, we need Person, Company and Position. This is akin to a dependency parsing task, but at the tag-level rather than the word-level (Dozat and Manning, 2016; Chen and Manning, 2014). We thus proposed the following baseline, which modifies the previous entity+relation strategy to entity+dependency, denoted by *tagging+dependency*. We use the BERT tagging model to assign tagging labels to each word, and modify the current SOTA dependency parsing model Biaffine (Dozat and Manning, 2016) to construct dependencies between tags. The Biaffine dependency model and the entity-extraction model are jointly trained.

Results are presented in Table 5. As can be seen, the tagging+dependency model outperforms the tagging+relation model. The proposed multi-turn QA model performs the best, with RL adding additional performance boost. Specially, for Person extraction, which only requires single-turn QA, the multi-turn QA+RL model performs the same as the multi-turn QA model. It is also the case in tagging+relation and tagging+dependency.

5.2 Results on ACE04, ACE05 and CoNLL04

For ACE04, ACE05 and CoNLL04, only two turns of QA are required. For evaluation, we report micro-F1 scores, precision and recall on entities and relations (Tables 6, 7 and 8) as in Li and Ji (2014); Miwa and Bansal (2016); Katiyar and Cardie (2017); Zhang et al. (2017). For ACE04, the proposed multi-turn QA model already outperforms previous SOTA by +1.8% for entity extraction and +1.0% for relation extraction. For ACE05, the proposed multi-turn QA model outperforms previous SOTA by +1.2% for entity extraction and +0.6% for relation extraction. The proposed multi-turn QA model leads to a +2.2% improvement on entity F1 and +1.1% on relation F1.

6 Ablation Studies

6.1 Effect of Question Generation Strategy

In this subsection, we compare the effects of natural language questions and pseudo-questions. Results are shown in Table 9.

We can see that natural language questions lead to a strict F1 improvement across all datasets. This is because natural language questions provide more fine-grained semantic information and can help entity/relation extraction. By contrast, the pseudo-questions provide very coarse-grained, ambiguous and implicit hints of entity and relation types, which might even confuse the model.

6.2 Effect of Joint Training

In this paper, we decompose the entity-relation extraction task into two subtasks: a multi-answer task for head-entity extraction and a single-answer task for joint relation and tail-entity extraction. We jointly train two models with parameters shared.

	multi-turn QA			multi-turn QA+RL			tagging+dependency			tagging+relation		
	p	r	f	p	r	f	p	r	f	p	r	f
Person	**98.1**	**99.0**	**98.6**	**98.1**	**99.0**	**98.6**	97.0	97.2	97.1	97.0	97.2	97.1
Company	82.3	87.6	84.9	**83.3**	**87.8**	**85.5**	81.4	87.3	84.2	81.0	86.2	83.5
Position	97.1	98.5	97.8	**97.3**	**98.9**	**98.1**	96.3	98.0	97.0	94.4	97.8	96.0
Time	96.6	98.8	97.7	**97.0**	**98.9**	**97.9**	95.2	96.3	95.7	94.0	95.9	94.9
all	91.0	93.2	92.1	**91.6**	**93.5**	**92.5**	90.0	91.7	90.8	88.2	91.5	89.8

Table 5: Results for different models on the RESUME dataset.

Models	**Entity P**	**Entity R**	**Entity F**	**Relation P**	**Relation R**	**Relation F**
Li and Ji (2014)	**83.5**	76.2	79.7	**60.8**	36.1	49.3
Miwa and Bansal (2016)	80.8	**82.9**	**81.8**	48.7	**48.1**	**48.4**
Katiyar and Cardie (2017)	81.2	78.1	79.6	46.4	45.3	45.7
Bekoulis et al. (2018)	-	-	81.6	-	-	47.5
Multi-turn QA	**84.4**	82.9	**83.6**	50.1	**48.7**	**49.4** (+1.0)

Table 6: Results of different models on the ACE04 test set. Results for pipelined methods are omitted since they consistently underperform joint models (see Li and Ji (2014) for details).

The parameter λ control the tradeoff between the two subtasks:

$$\mathcal{L} = (1-\lambda)\mathcal{L}(\text{head-entity}) + \lambda\mathcal{L}(\text{tail-entity}) \quad (4)$$

Results regarding different values of λ on the ACE05 dataset are given as follows:

λ	**Entity F1**	**Relation F1**
$\lambda = 0$	85.0	55.1
$\lambda = 0.1$	84.8	55.4
$\lambda = 0.2$	**85.2**	56.2
$\lambda = 0.3$	84.8	56.4
$\lambda = 0.4$	84.6	57.9
$\lambda = 0.5$	84.8	58.3
$\lambda = 0.6$	84.6	58.9
$\lambda = 0.7$	84.8	**60.2**
$\lambda = 0.8$	83.9	58.7
$\lambda = 0.9$	82.7	58.3
$\lambda = 1.0$	81.9	57.8

When λ is set to 0, the system is essentially only trained on the head-entity prediction task. It is interesting to see that $\lambda = 0$ does not lead to the best entity-extraction performance. This demonstrates that the second-stage relation extraction actually helps the first-stage entity extraction, which again confirms the necessity of considering these two subtasks together. For the relation extraction task, the best performance is obtained when λ is set to 0.7.

6.3 Case Study

Table 10 compares outputs from the proposed multi-turn QA model with the ones of the previous SOTA MRT model (Sun et al., 2018). In the first example, MRT is not able to identify the relation between *john scottsdale* and *iraq* because the two entities are too far away, but our proposed QA model is able to handle this issue. In the second example, the sentence contains two pairs of the same relation. The MRT model has a hard time identifying handling this situation, not able to locate the *ship* entity and the associative relation, which the multi-turn QA model is able to handle this case.

7 Conclusion

In this paper, we propose a multi-turn question answering paradigm for the task of entity-relation extraction. We achieve new state-of-the-art results on 3 benchmark datasets. We also construct a new entity-relation extraction dataset that requires hierarchical relation reasoning and the proposed model achieves the best performance.

References

Giannis Bekoulis, Johannes Deleu, Thomas Demeester, and Chris Develder. 2018. Adversarial training for multi-context joint entity and relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2830–2836. Association for Computational Linguistics.

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. 2009. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM.

Yee Seng Chan and Dan Roth. 2011. Exploiting syntactico-semantic structures for relation extraction. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 551–560. Association for Computational Linguistics.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 conference on*

Models	**Entity P**	**Entity R**	**Entity F**	**Relation P**	**Relation R**	**Relation F**
Li and Ji (2014)	**85.2**	76.9	80.8	**65.4**	39.8	49.5
Miwa and Bansal (2016)	82.9	**83.9**	83.4	57.2	54.0	55.6
Katiyar and Cardie (2017)	84.0	81.3	82.6	55.5	51.8	53.6
Zhang et al. (2017)	-	-	83.5	-	-	57.5
Sun et al. (2018)	83.9	83.2	**83.6**	64.9	**55.1**	**59.6**
Multi-turn QA	84.7	**84.9**	**84.8**	64.8	**56.2**	**60.2** (+0.6)

Table 7: Results of different models on the ACE05 test set. Results for pipelined methods are omitted since they consistently underperform joint models (see Li and Ji (2014) for details).

Models	**Entity P**	**Entity R**	**Entity F1**	**Relation P**	**Relation R**	**Relation F**
Miwa and Sasaki (2014)	–	–	80.7	–	–	61.0
Zhang et al. (2017)	–	–	85.6	–	–	67.8
Bekoulis et al. (2018)	–	–	83.6	–	–	62.0
Multi-turn QA	**89.0**	**86.6**	**87.8**	**69.2**	**68.2**	**68.9** (+2.1)

Table 8: Comparison of the proposed method with the previous models on the CoNLL04 dataset. Precision and recall values of baseline models were not reported in the previous papers.

RESUME			
Model	Overall P	Overall R	Overall F
Pseudo Q	90.2	92.3	91.2
Natural Q	91.0	93.2	92.1

ACE04						
Model	EP	ER	EF	RP	RR	RF
Pseudo Q	83.7	81.3	82.5	49.4	47.2	48.3
Natural Q	**84.4**	**82.9**	**83.6**	**50.1**	**48.7**	**49.9**

ACE05						
Model	EP	ER	EF	RP	RR	RF
Pseudo Q	83.6	84.7	84.2	60.4	55.9	58.1
Natural Q	**84.7**	**84.9**	**84.8**	**64.8**	**56.2**	**60.2**

CoNLL04						
Model	EP	ER	EF	RP	RR	RF
Pseudo Q	87.4	86.4	86.9	68.2	67.4	67.8
Natural Q	**89.0**	**86.6**	**87.8**	**69.6**	**68.2**	**68.9**

Table 9: Comparing of the effect of natural language questions with pseudo-questions.

EXAMPLE1	[john scottsdale] PER: PHYS-1 is on the front lines in [iraq]GPE: PHYS-1 .
MRT	[john scottsdale] PER is on the front lines in [iraq]GPE .
MULTI-QA	[john scottsdale] PER: PHYS-1 is on the front lines in [iraq]GPE: PHYS-1 .
EXAMPLE2	The [men] PER: ART-1 held on the sinking [vessel] VEH: ART-1 until the [passenger] PER: ART-2 [ship] VEH: ART-2 was able to reach them.
MRT	The [men] PER: ART-1 held on the sinking [vessel] VEH: ART-1 until the [passenger]PER ship was able to reach them.
MULTI-QA	The [men] PER: ART-1 held on the sinking [vessel] VEH: ART-1 until the [passenger] PER: ART-2 [ship] VEH: ART-2 was able to reach them.

Table 10: Comparing the multi-turn QA model with MRT (Sun et al., 2018).

empirical methods in natural language processing (EMNLP), pages 740–750.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Timothy Dozat and Christopher D Manning. 2016. Deep biaffine attention for neural dependency parsing. *arXiv preprint arXiv:1611.01734*.

Matthew Dunn, Levent Sagun, Mike Higgins, V Ugur Guney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

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

Zhiheng Huang, Wei Xu, and Kai Yu. 2015. Bidirectional lstm-crf models for sequence tagging. *arXiv preprint arXiv:1508.01991*.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Rohit J Kate and Raymond J Mooney. 2010. Joint entity and relation extraction using card-pyramid parsing. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 203–212. Association for Computational Linguistics.

Arzoo Katiyar and Claire Cardie. 2017. Going out on a limb: Joint extraction of entity mentions and relations without dependency trees. In *Proceedings*

of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 917–928. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

John Lafferty, Andrew McCallum, and Fernando CN Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data.

Oliver Lemon, Kallirroi Georgila, James Henderson, and Matthew Stuttle. 2006. An isu dialogue system exhibiting reinforcement learning of dialogue policies: generic slot-filling in the talk in-car system. In *Proceedings of the Eleventh Conference of the European Chapter of the Association for Computational Linguistics: Posters & Demonstrations*, pages 119–122. Association for Computational Linguistics.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. *arXiv preprint arXiv:1706.04115*.

Jiwei Li, Alexander H Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. 2016a. Dialogue learning with human-in-the-loop. *arXiv preprint arXiv:1611.09823*.

Jiwei Li, Alexander H Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. 2016b. Learning through dialogue interactions by asking questions. *arXiv preprint arXiv:1612.04936*.

Qi Li and Heng Ji. 2014. Incremental joint extraction of entity mentions and relations. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 402–412.

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

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional lstm-cnns-crf. *arXiv preprint arXiv:1603.01354*.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. *arXiv preprint arXiv:1601.00770*.

Makoto Miwa, Rune Sætre, Yusuke Miyao, and Jun'ichi Tsujii. 2009. A rich feature vector for protein-protein interaction extraction from multiple corpora. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 1-Volume 1*, pages 121–130. Association for Computational Linguistics.

Makoto Miwa and Yutaka Sasaki. 2014. Modeling joint entity and relation extraction with table representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1858–1869.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2015. Multi-domain dialog state tracking using recurrent neural networks. *arXiv preprint arXiv:1506.07190*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Dan Roth and Wen-tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. Technical report, ILLINOIS UNIV AT URBANA-CHAMPAIGN DEPT OF COMPUTER SCIENCE.

Dan Roth and Wen-tau Yih. 2007. Global inference for entity and relation identification via a linear programming formulation. *Introduction to statistical relational learning*, pages 553–580.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. 2017. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1047–1055. ACM.

Sameer Singh, Sebastian Riedel, Brian Martin, Jiaping Zheng, and Andrew McCallum. 2013. Joint inference of entities, relations, and coreference. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 1–6. ACM.

Changzhi Sun, Yuanbin Wu, Man Lan, Shiliang Sun, Wenting Wang, Kuang-Chih Lee, and Kewen Wu. 2018. Extracting entities and relations with joint minimum risk training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2256–2265.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. *arXiv preprint arXiv:1503.00075*.

Ryuichi Takanobu, Tianyang Zhang, Jiexi Liu, and Minlie Huang. 2018. A hierarchical framework for relation extraction with reinforcement learning. *arXiv preprint arXiv:1811.03925*.

Chuanqi Tan, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2017. S-net: From answer extraction to answer generation for machine reading comprehension. *arXiv preprint arXiv:1706.04815*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Linlin Wang, Zhu Cao, Gerard de Melo, and Zhiyuan Liu. 2016a. Relation classification via multi-level attention cnns.

Shuohang Wang and Jing Jiang. 2016. Machine comprehension using match-lstm and answer pointer. *arXiv preprint arXiv:1608.07905*.

Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2017a. Evidence aggregation for answer re-ranking in open-domain question answering. *arXiv preprint arXiv:1711.05116*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017b. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 189–198.

Zhiguo Wang, Haitao Mi, Wael Hamza, and Radu Florian. 2016b. Multi-perspective context matching for machine comprehension. *arXiv preprint arXiv:1612.04211*.

Tsung-Hsien Wen, David Vandyke, Nikola Mrksic, Milica Gasic, Lina M Rojas-Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2016. A network-based end-to-end trainable task-oriented dialogue system. *arXiv preprint arXiv:1604.04562*.

Jason D Williams and Steve Young. 2005. Scaling up pomdps for dialog management: The"summary pomdp"method. In *IEEE Workshop on Automatic Speech Recognition and Understanding, 2005.*, pages 177–182. IEEE.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Caiming Xiong, Victor Zhong, and Richard Socher. 2017. Dcn+: Mixed objective and deep residual coattention for question answering. *arXiv preprint arXiv:1711.00106*.

Bishan Yang and Claire Cardie. 2013. Joint inference for fine-grained opinion extraction. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1640–1649.

Hai Ye, Wenhan Chao, Zhunchen Luo, and Zhoujun Li. 2016. Jointly extracting relations with class ties via effective deep ranking. *arXiv preprint arXiv:1612.07602*.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. *arXiv preprint arXiv:1804.09541*.

Xiaofeng Yu and Wai Lam. 2010. Jointly identifying entities and extracting relations in encyclopedia text via a graphical model approach. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 1399–1407. Association for Computational Linguistics.

Dmitry Zelenko, Chinatsu Aone, and Anthony Richardella. 2003. Kernel methods for relation extraction. *Journal of machine learning research*, 3(Feb):1083–1106.

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 (Volume 1: Long Papers)*, volume 1, pages 506–514.

Meishan Zhang, Yue Zhang, and Guohong Fu. 2017. End-to-end neural relation extraction with global optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1730–1740.

Suncong Zheng, Yuexing Hao, Dongyuan Lu, Hongyun Bao, Jiaming Xu, Hongwei Hao, and Bo Xu. 2017. Joint entity and relation extraction based on a hybrid neural network. *Neurocomputing*, 257:59–66.