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

ACL2019

Overview

- Real-life RE could involve hierarchical entity relations which are not well-captured by current approaches.
- This paper introduces a new way to do entity-relation extraction as multi-terun question answering.
- Error propagation between turns of answering the questions is relieved by reinforcement learning.

Hierarchical entity relations

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.

- *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.*

MRC-like model

- Question: $Q = \{q_1, q_2, ..., q_{N_q}\}$
- Context: $C = \{c_1, c_2, ..., c_{N_c}\}$
- BERT: [CLS, Q, SEP, C, SEP]
- Softmax over BMEO tags.
- Training objective:
  $\mathcal{L} = (1 - \lambda)\mathcal{L}(\text{head-entity}) + \lambda\mathcal{L}(\text{tail-entity, rel})$

Approach: 2-turn QA

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.

Approach: 4-turn QA

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.

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.

Multi-turn Question Answering

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

M ← ∅
HeadEntList ← ∅
for entity_question in EntityQuesTemplates do
    e1 = Extract_Answer(entity_question, s)
    if e1 ≠ NONE do
        HeadEntList = HeadEntList + {e1}
    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 ≠ 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.

Reinforcement learning

- Action: text spans being selected.
- Policy:

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

- Reward: +1 if the selected answer is correct.
- Initialized with pretrained MRC-like models.
- Use REINFORCE to maximize the expected reward:

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

Results

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).

	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.