# REET Joint Relation Extraction and Entity Typing via Multi-task Learning

ADVISOR: JIA-LING, KOH

SOURCE: NLPCC 2019

SPEAKER: SHAO-WEI, HUANG

DATE: 2020/03/16

# OUTLINE

- Introduction
- Method
- Experiment
- Conclusion

# INTRODUCTION

- **Relation Extraction（RE）**：Extracting semantic relations between two entities from the text corpus.

(Ex): Steve jobs was the co-founder of apple.

Co-Founder

# INTRODUCTION

- **Entity Typing（ET）**：Assign types into the entity mention in a sentence.

(Ex): Steve jobs was the co-founder of apple.

Person

Company

# INTRODUCTION

- Most existing works solve RE and ET separately and regard them as independent tasks.
- In fact, the two tasks have a strong inner relationship.

REET Model

Joint Relation Extraction and Entity Typing.

# INTRODUCTION

## Problem definition

- Given a sentence s = $\{w_1, w_2, \ldots e_1, \ldots, e_2 \ldots\}$ and two target entities $(e_1, e_2)$.
- Subtasks :
  1. Relation extraction for the entity pair.
  2. Entity typing for $e_1$.
  3. Entity typing for $e_2$.

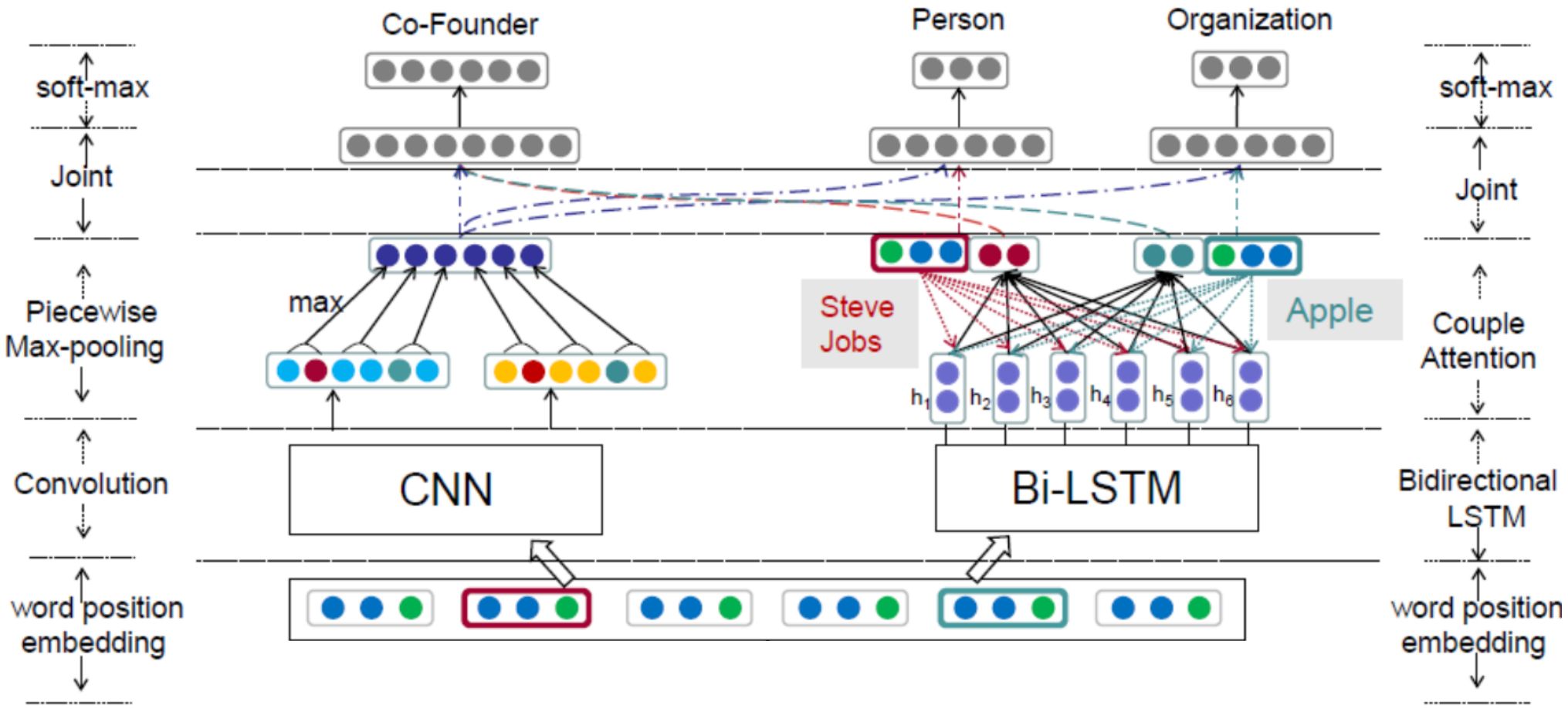
# OUTLINE

- Introduction
- Method
- Experiment
- Conclusion

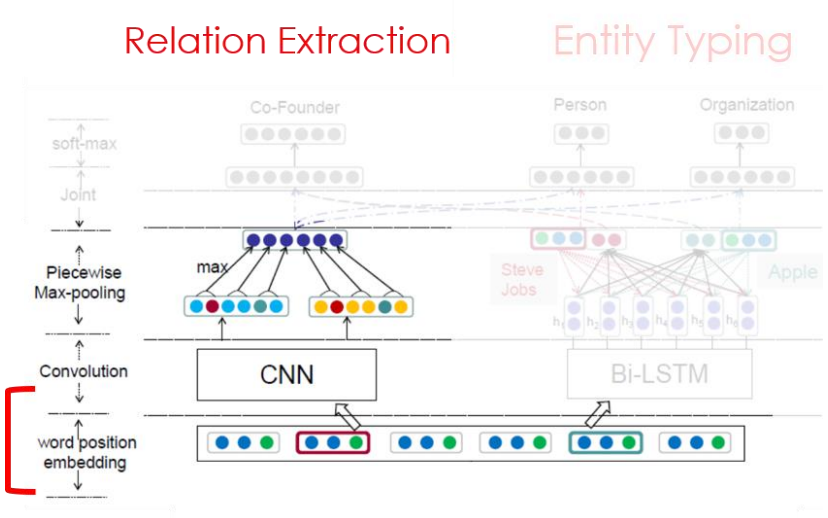
# FRAMEWORK



Relation Extraction Entity Typing



# METHOD **Relation Extraction Module**




- For a sentence s = $\{w_1, w_2, \ldots e_1, \ldots, e_2 \ldots w_n\}$ , transform each word $w_i$ into :
  1. Word embeddings
  2. Position embeddings : Encodes the relative distances between $w_i$ and the two entities.

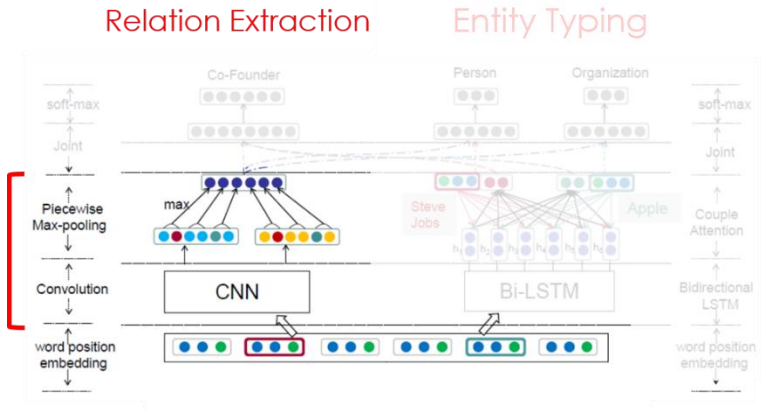
-3 $w_i$ 2

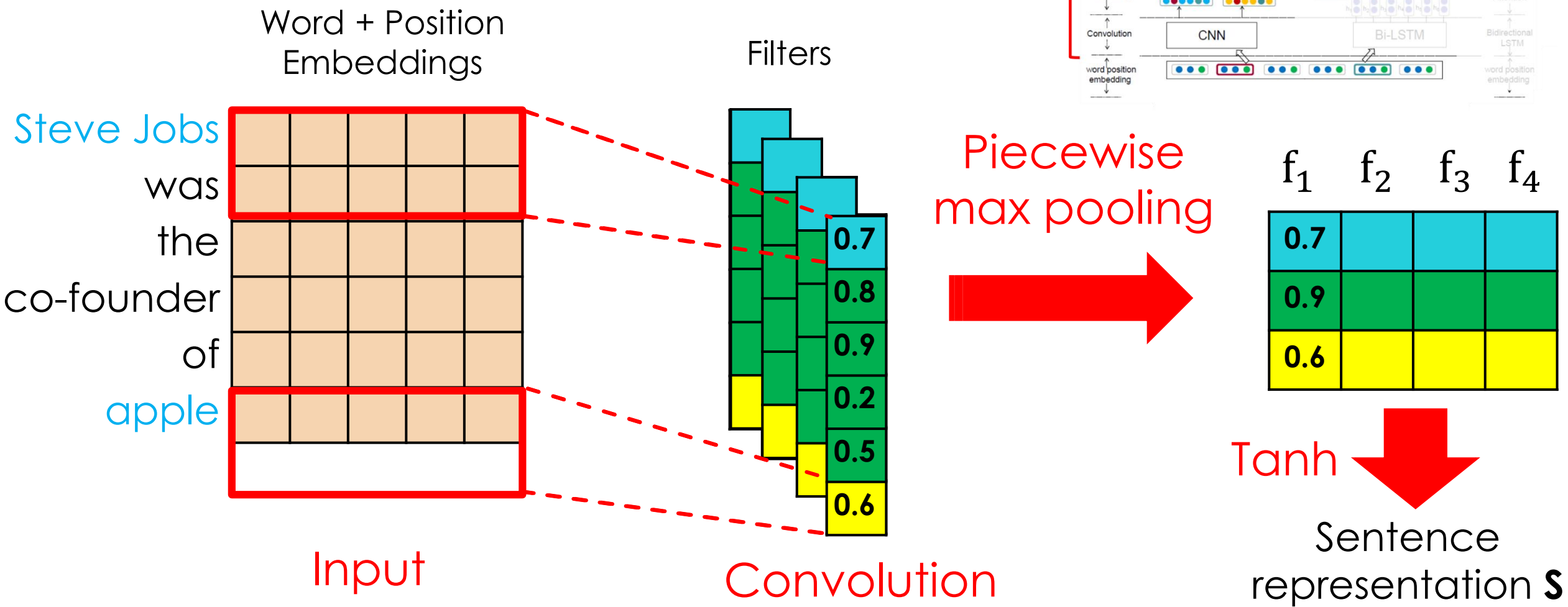
(Ex): Steve jobs was the co-founder of apple.

# METHOD Relation Extraction Module

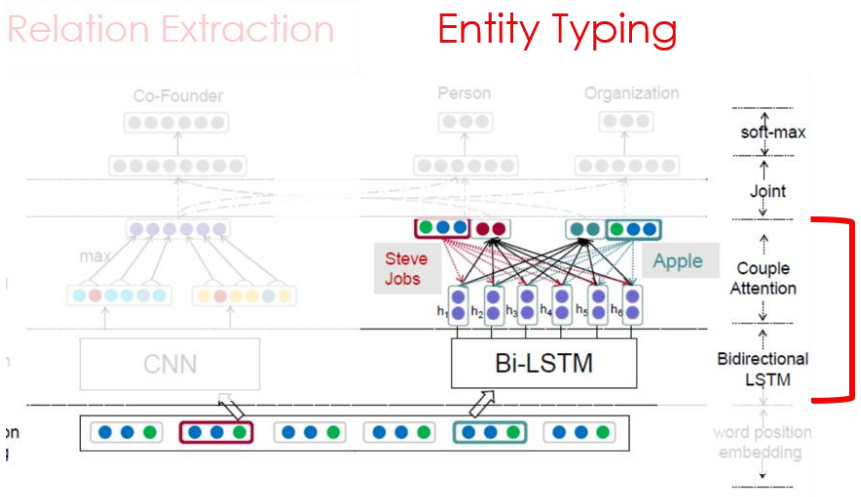


➢**Convolution and Piecewise max pooling** :



# METHOD Entity Typing Module

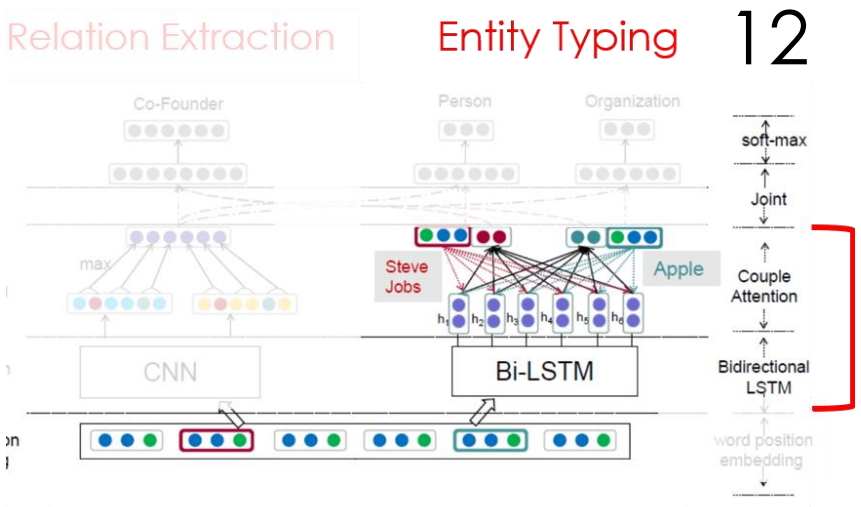



- **Input Layer**：Shared with RE module.
  1. Word embeddings.
  2. Position embeddings.

- **Bi-LSTM layer**：Obtain the hidden state(high-level semantic representation) of each $w_i$.

# METHOD Entity Typing Module






➢ **Couple Attention**：To get entity-related representations for sentences.

- $k_i = \tanh(W_s \mathbf{h_i} + b_s)$

- Treat entities as query, other words as key.

$$\boxed{\alpha_m^i} = \frac{\exp\left(k_i^T \mathbf{e_m}\right)}{\sum_j^n \exp\left(k_j^T \mathbf{e_m}\right)} \ , \ \boxed{m = 1, 2}$$

The weights of the i-th word under the m-th entity

entity1, entity2

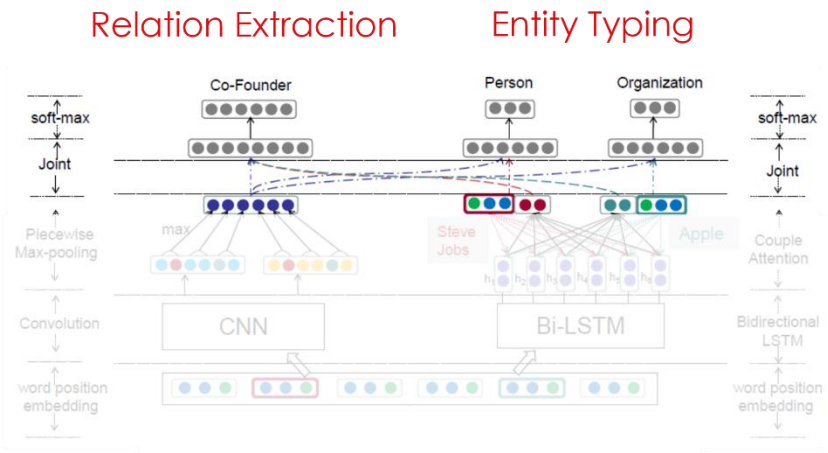
- Weight sum： $\mathbf{v_m} = \sum_i^n \alpha_m^i \mathbf{h_i} \ , \ m = 1, 2$

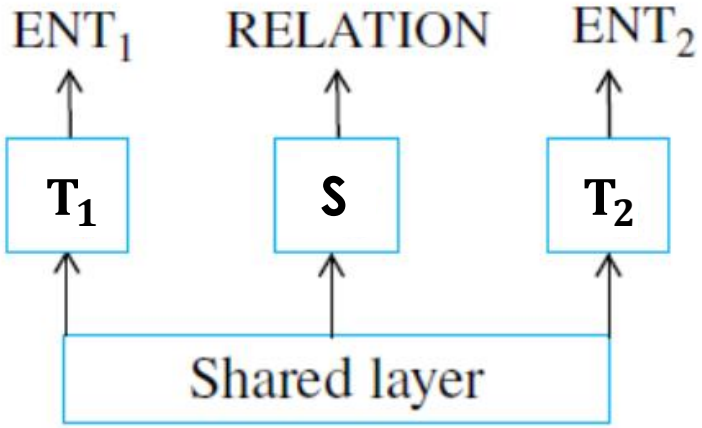
- Final representation of two ET tasks：

$$\mathbf{T_1} = [\mathbf{e_1}, \mathbf{v_1}] \ , \quad \mathbf{T_2} = [\mathbf{e_2}, \mathbf{v_2}]$$

# METHOD Multi-task Learning Framework

➢ **REET1** : Treat RE task and ET task are independent and only share input embedding layers.




- 

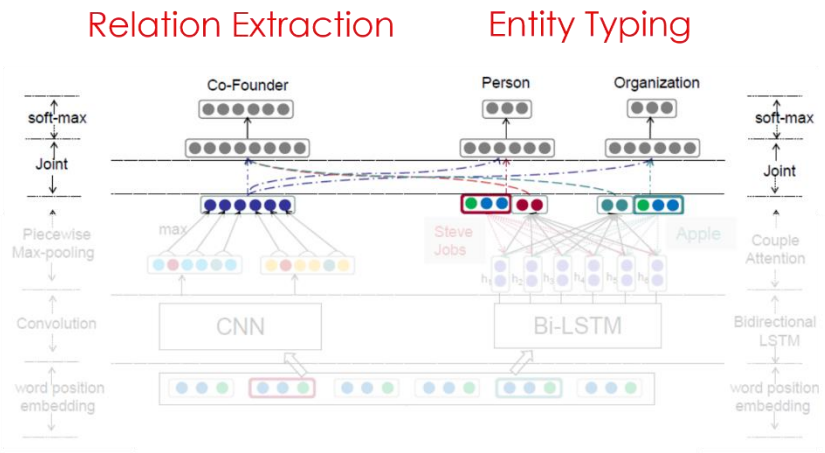
(REET-I)

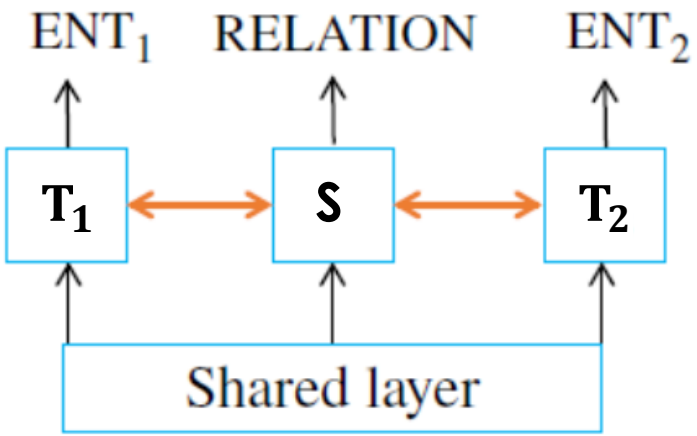
- $$\mathbf{p_r} = \mathrm{softmax}(W_r \mathbf{S} + b_r) \ ,$$
$$\mathbf{p_{t_i}} = \mathrm{softmax}(W_{t_i} \mathbf{T_i} + b_{t_i}) \ , \ i = 1, 2$$

Prediction probabilities for RE and ET respectively.

# METHOD Multi-task Learning Framework

➢ **REET2** : Concatenate representations of RE and ET before the last classication layer.




- 

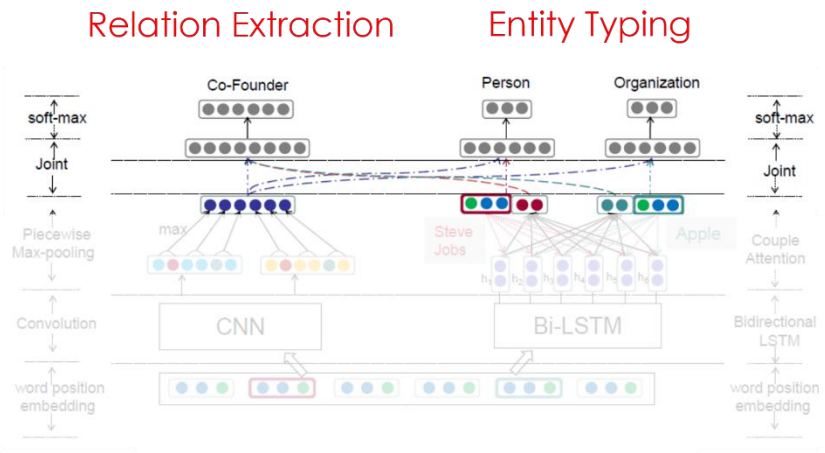
(REET-2)

- $$\mathbf{p_r} = \mathtt{softmax}(W_r[\mathbf{T_1}, \mathbf{S}, \mathbf{T_2}] + b_r) \ ,$$
$$\mathbf{p_{t_i}} = \mathtt{softmax}(W_{t_i}[\mathbf{S}, \mathbf{T_i}] + b_{t_i}) \ , \ i = 1, 2$$

Prediction probabilities for RE and ET respectively.

***RE and ET can share a high-level feature with each other.

# METHOD Multi-task Learning Framework




- **Loss Function**：Cross entropy loss.
  - $$L_r(\theta_0) = -\frac{1}{R}\sum_{k=1}^{R} y_r \log \mathbf{p_r}(k)$$

    $$L_{\mathbf{t_i}}(\theta_i) = -\frac{1}{C}\sum_{k=1}^{C} y_{\mathbf{t_i}} \log \mathbf{p_{t_i}}(k)\ ,\ \ i = 1, 2$$

- **Multi-task Learning**：Add the loss of each task together.
  - $$L(\theta) = \boxed{\lambda} L_{t_1} + \lambda L_{t_2} + (1-\lambda) L_r$$

    Balance weight

# OUTLINE

- Introduction
- Method
- Experiment
- Conclusion

# EXPERIMENT

https://ai.googleblog.com/2013/04/50000-lessons-on-how-to-read-relation.html



## Dataset

- **NYT+Freebase**：Aligning entities and relations in Freebase with the corpus of New York Times.
- **Google Distant Supervision(SGD)**： Extracted from Google Relation Extraction corpus and is a human-judged dataset.

| Dataset | # relations | #entity types | # sentences | # entity-pair |
|---|---|---|---|---|
| **NYT+Freebase Dataset** | | | | |
| Train | 53 | 5 | 455,771 | 233,064 |
| Dev | 53 | 5 | 114,317 | 58,635 |
| Test | 53 | 5 | 172,448 | 96,678 |
| **GDS Dataset** | | | | |
| Train | 5 | 25 | 11,297 | 6,498 |
| Dev | 5 | 25 | 1,864 | 1,082 |
| Test | 5 | 25 | 5,663 | 3,247 |

https://blog.csdn.net/u013249853/article/details/96132766



# EXPERIMENT

## Performance in RE

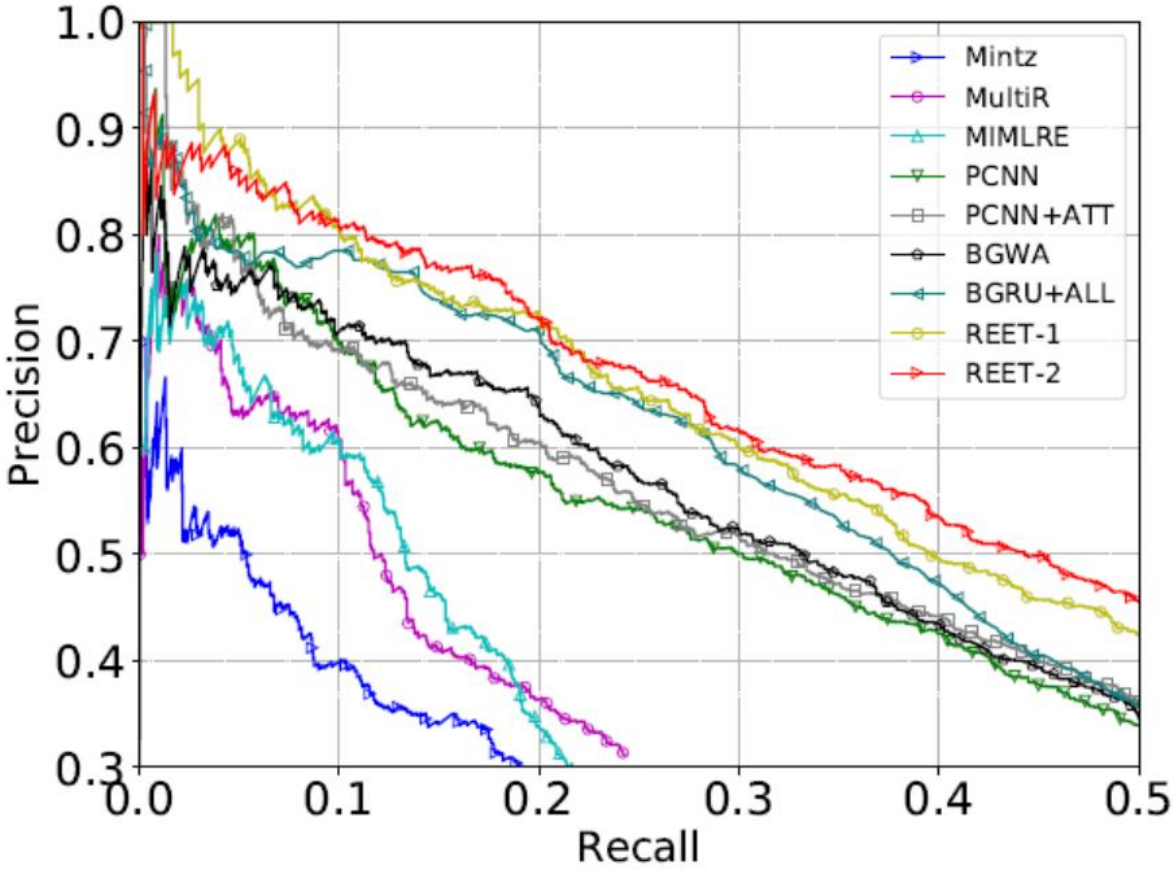



**Fig. 3.** Precision-Recall curves on NYT.

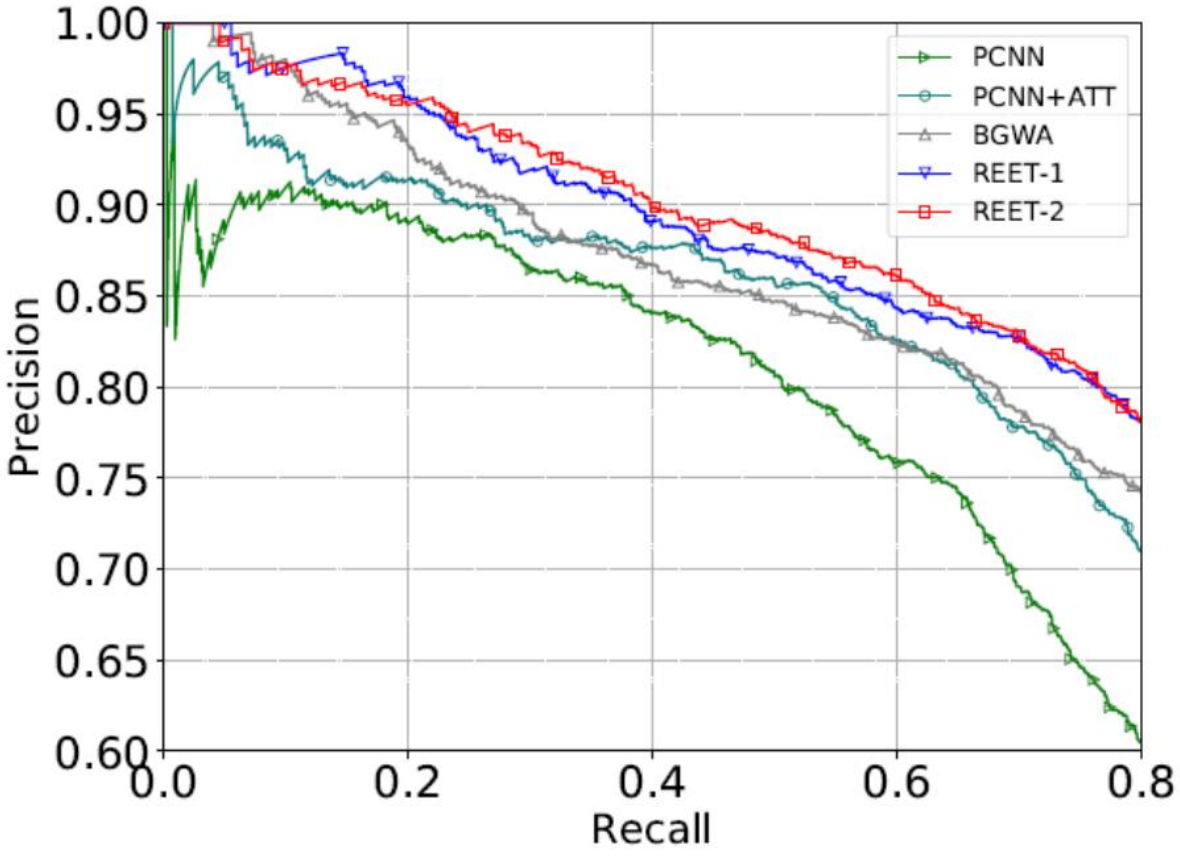



**Fig. 4.** Precision-Recall curves on GDS.

# EXPERIMENT

## Performance in ET

| F1 (%) | BiLSTM | BiLSTM+Co_ATT | REET-1 | REET-2 |
|---|---|---|---|---|
| NYT+Freebase | 94.7 | 95.5 | 96.5 | **96.8** |
| GDS | 70.1 | 72.8 | 74.2 | **76.6** |

**Table 3.** Classification performance of entity typing task.

https://blog.xuite.net/metafun/life/65137005-Information+Retrieval%E7%9A%84%E8%A1%A1%E9%87%8F%E6%8C%87%E6%A8%99-MAP%E5%92%8CMRR



# EXPERIMENT

## Parameter analysis

*** $L(\theta) = \lambda L_{t_1} + \lambda L_{t_2} + (1-\lambda) L_r$

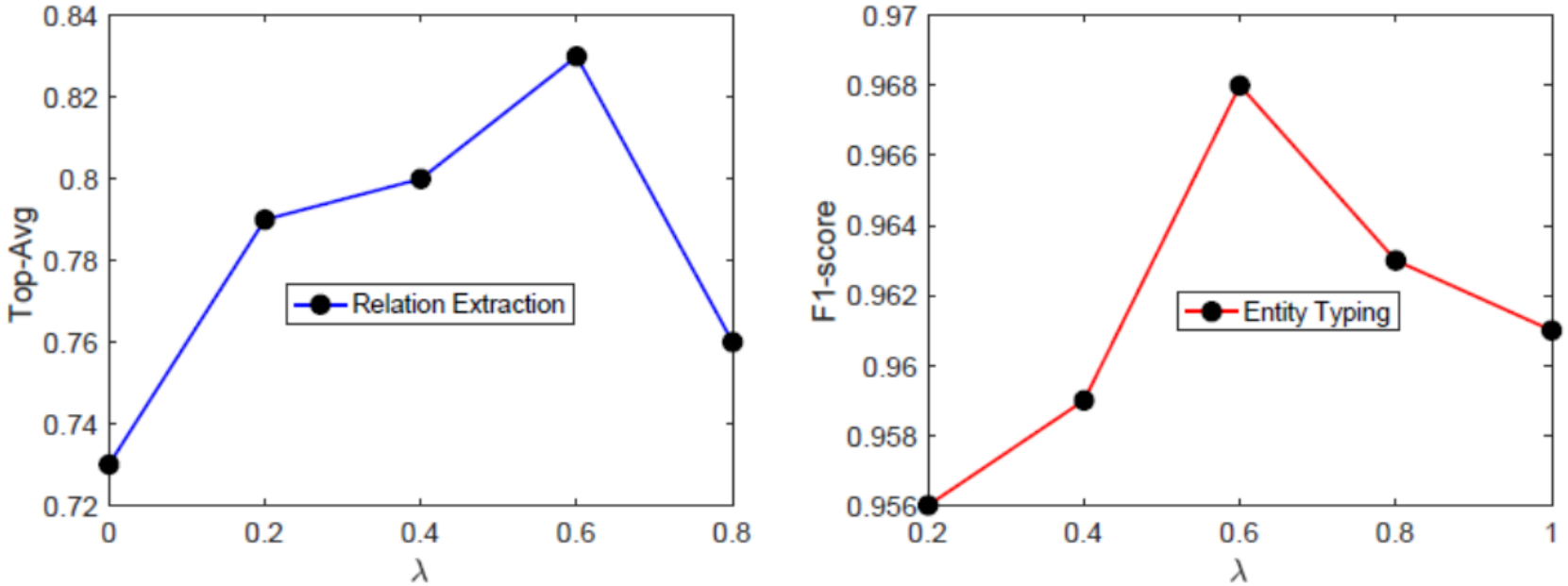



**Fig. 5.** the influence of parameter of $\lambda$ in relation extraction and entity typing.

# OUTLINE

- Introduction
- Method
- Experiment
- Conclusion

# CONCLUSION

- Propose a multi-task learning frame that integrates relation extraction task and entity typing task jointly.
- The two tasks share low-level (i.e., input embedding layer) and high-level information (i.e., task-specic feature).
- Both relation extraction task and entity typing task achieve a signicant improvement and our approach outperforms many baseline methods.