[ACL20]CASREL

A Novel Cascade Binary Tagging Framework for Relational Triple Extraction

Zhepei Wei,[1,2] Jianlin Su,[4] Yue Wang,[5] Yuan Tian[1,2], Yi Chang[1,2,3*]*

[1] School of Artificial Intelligence, Jilin University,
[2] Key Laboratory of Symbolic Computation and Knowledge Engineering, Jilin University
[3] International Center of Future Science, Jilin University
[4] Shenzhen Zhuiyi Technology Co., Ltd.
[5] School of Information and Library Science, University of North Carolina at Chapel Hill

Speaker: 杨晰

xyang41@stu.ecnu.edu.cn

Outline

- Background: Overlapping Triple Problem
- Motivation: Triple level
- Framework
- Experiments
- Conclusion

BG: Overlapping Triple Problem

- [1] EPO: Entity Pair Overlap
- [1] SEO: Single Entity Overlap

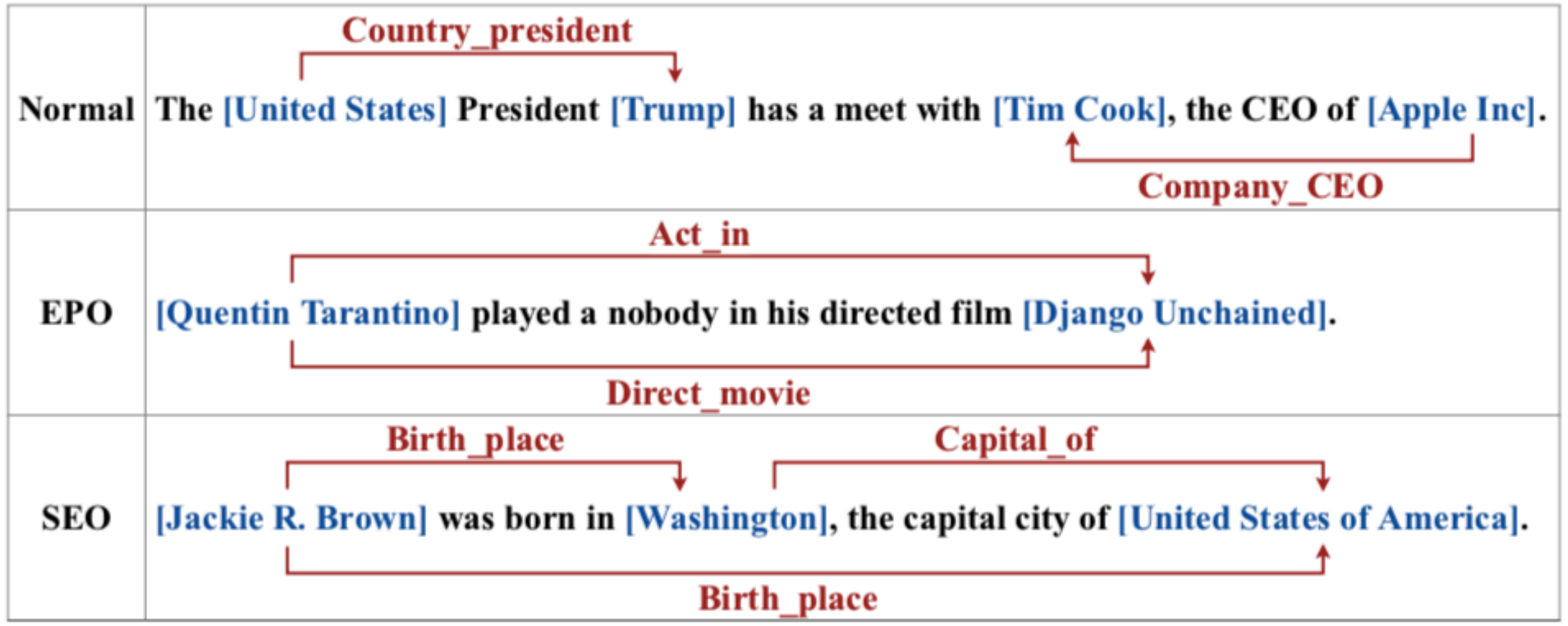


Figure 1: Examples of *Normal*, *EntityPairOverlap (EPO)* and *SingleEntityOverlap (SEO)* overlapping patterns.

[1] 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

MotivationI

- Entity Pairs + Relations vs Triple Level
  - the class distribution is highly imbalance ➡️ many negative examples
  - overlapping triples ➡️ need enough training examples
  - Relations as discrete labels on entity pairs 🆚 Relations as functions that map subjects to object
  - $f(s, o) \rightarrow r$ vs $f_r(s) \rightarrow o$ (Relation Classifiers vs Relation-Specific taggers)

MotivationII

$$\prod_{j=1}^{|D|}\left[\prod_{(s,r,o)\in T_j} p((s,r,o)|x_j)\right] \quad (1)$$

$$= \prod_{j=1}^{|D|}\left[\prod_{s\in T_j} p(s|x_j) \prod_{(r,o)\in T_j|s} p((r,o)|s,x_j)\right] \quad (2)$$

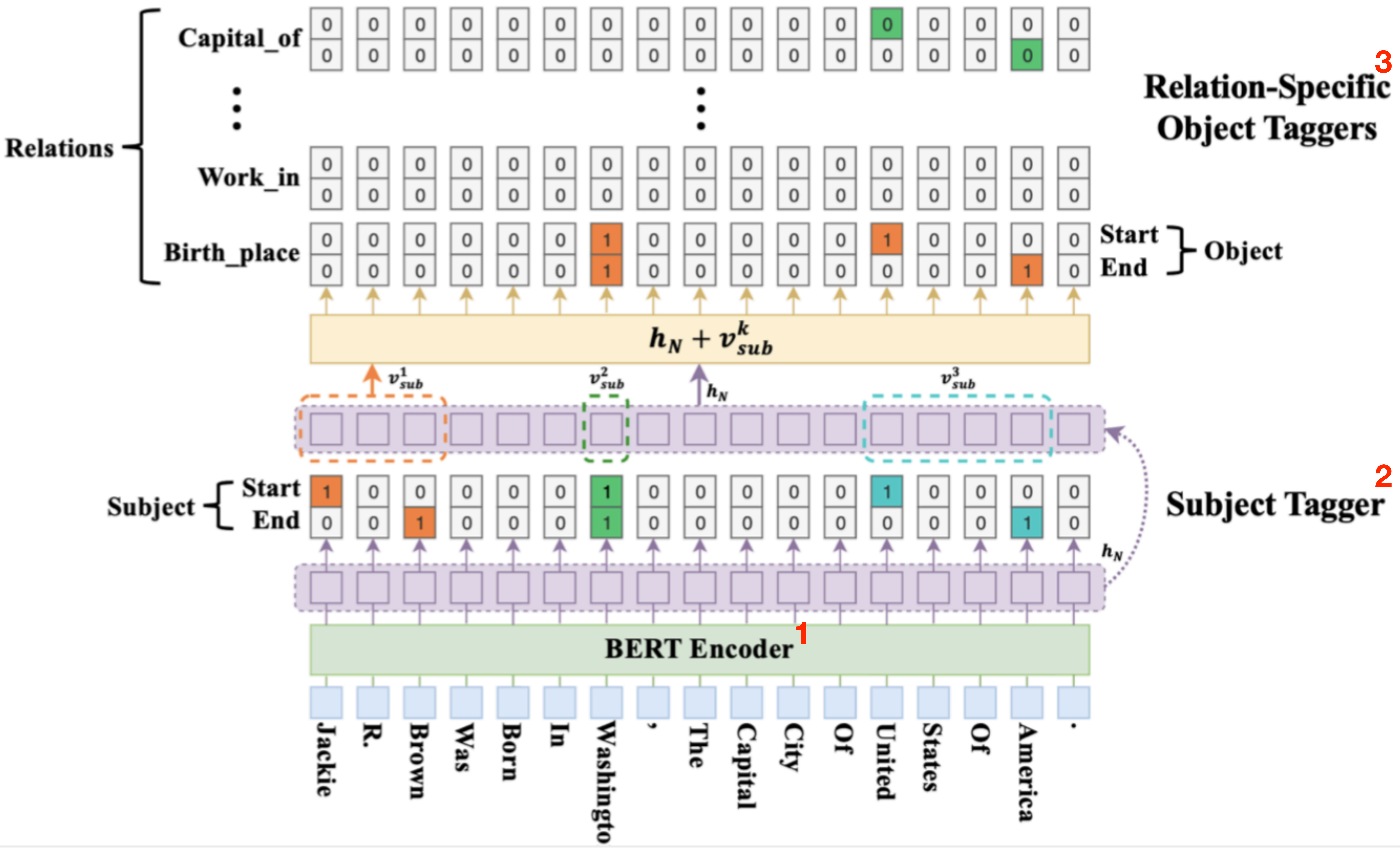
$$= \prod_{j=1}^{|D|}\left[\prod_{s\in T_j} p(s|x_j) \prod_{r\in T_j|s} p_r(o|s,x_j) \prod_{r\in R\backslash T_j|s} p_r(o_{\varnothing}|s,x_j)\right]. \quad (3)$$

- Training Objective at the Triple Level
  - Consistent with the final evaluation criteria
  - Handle the overlapping triple problem by design
  - Inspire a novel tagging scheme for Triple Extraction: 1️⃣Subject tagger 2️⃣Object tagger

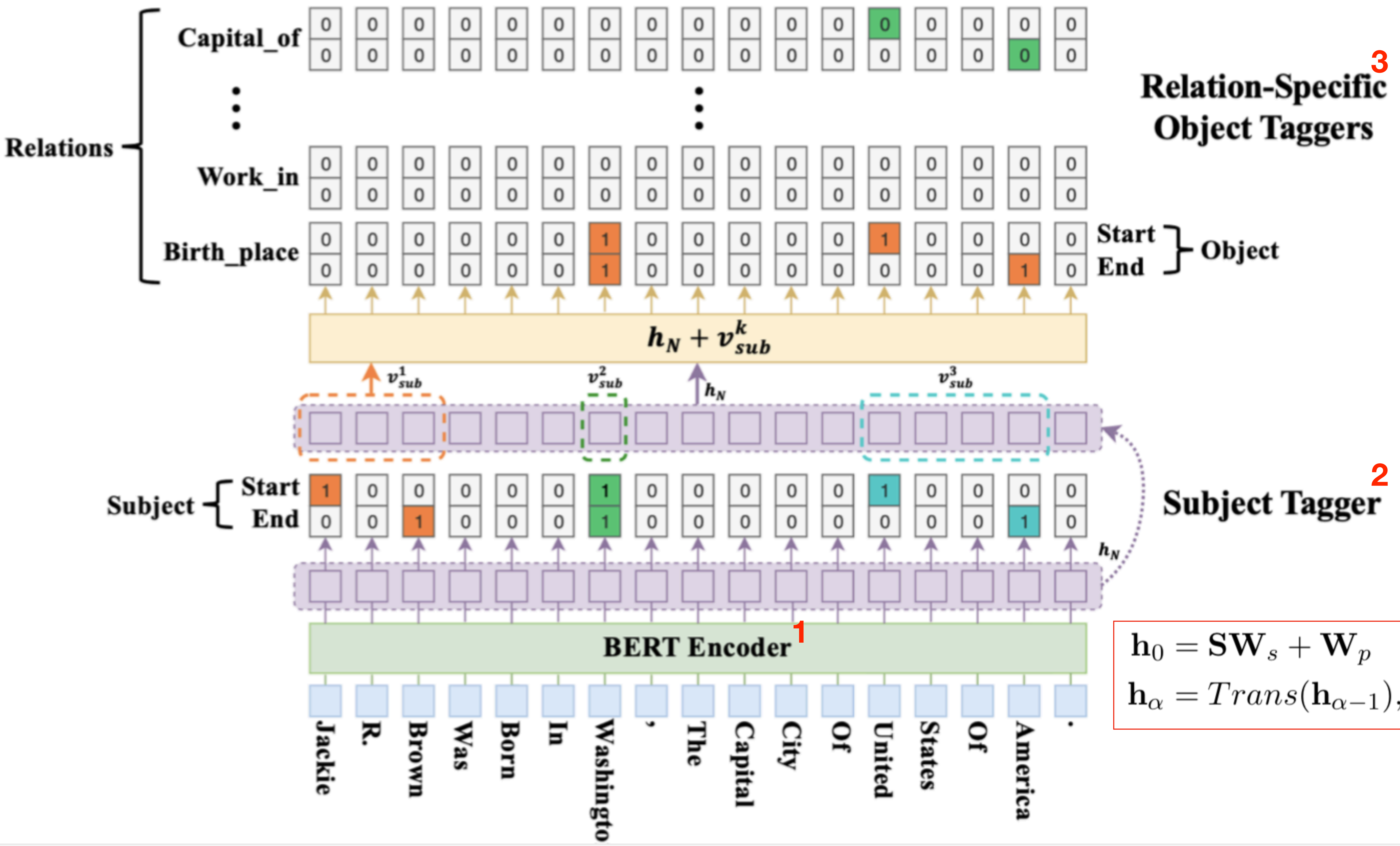
The CASREL Framework

- BERT Encoder
- Cascade Decoder
  - Subject Tagger
  - Relation-specific Object Taggers

The CASREL Framework



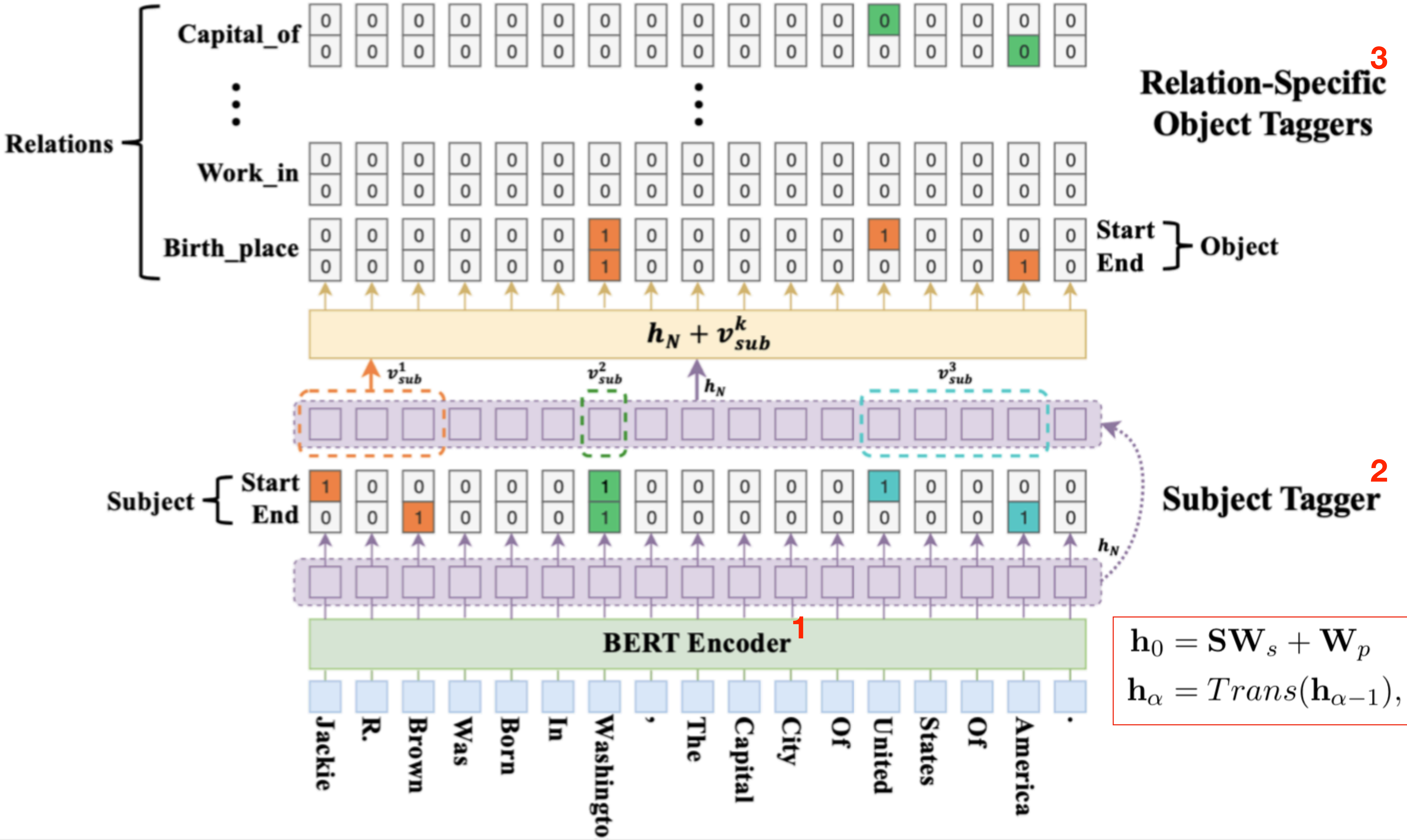
The CASREL Framework



$$\mathbf{h}_0 = \mathbf{S}\mathbf{W}_s + \mathbf{W}_p$$

$$\mathbf{h}_\alpha = Trans(\mathbf{h}_{\alpha-1}), \alpha \in [1, N]$$

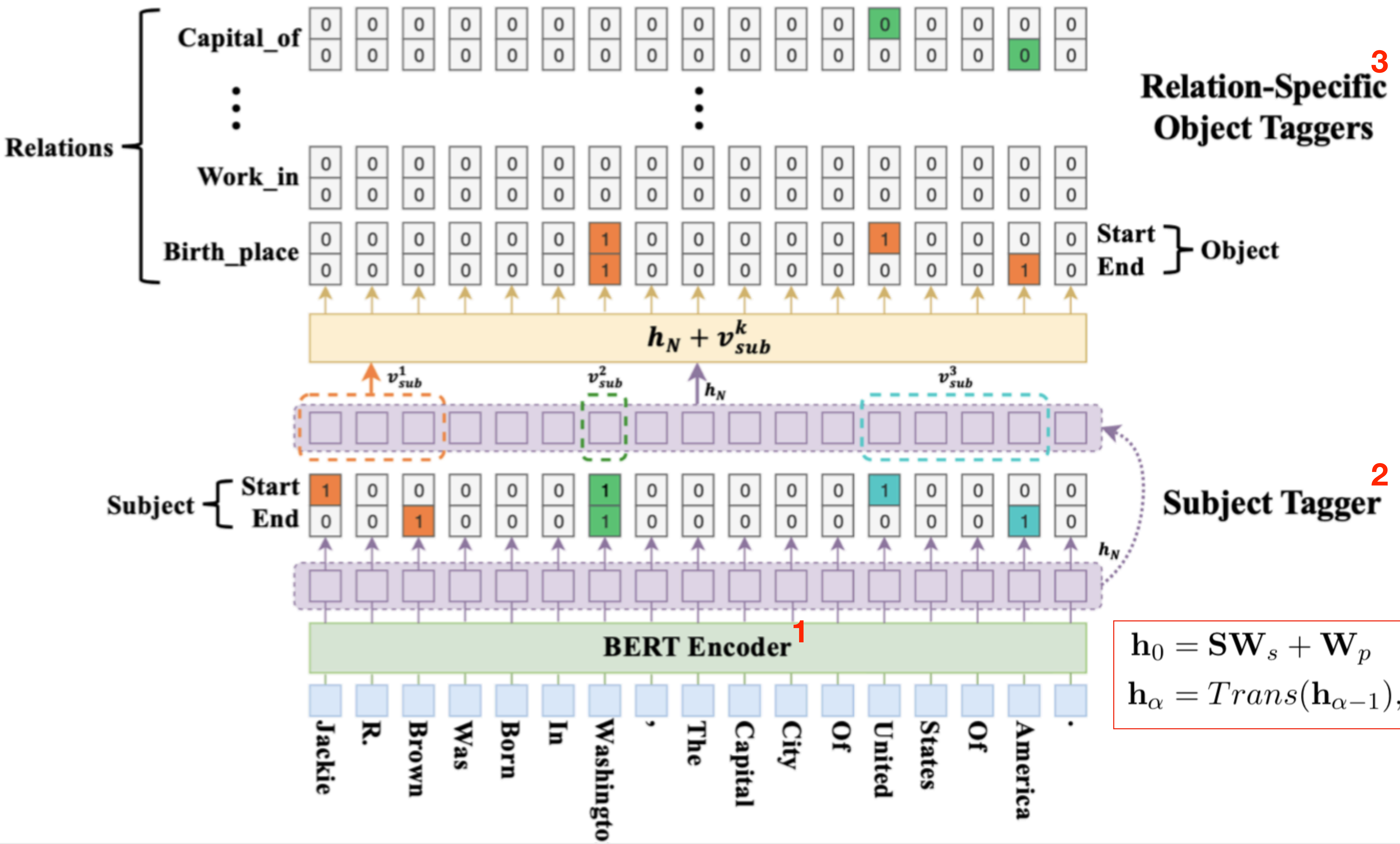
The CASREL Framework



$$p_i^{start\_s} = \sigma(\mathbf{W}_{start}\mathbf{x}_i + \mathbf{b}_{start})$$
$$p_i^{end\_s} = \sigma(\mathbf{W}_{end}\mathbf{x}_i + \mathbf{b}_{end})$$

$$\mathbf{h}_0 = \mathbf{S}\mathbf{W}_s + \mathbf{W}_p$$
$$\mathbf{h}_\alpha = Trans(\mathbf{h}_{\alpha-1}), \alpha \in [1, N]$$

The CASREL Framework



$$p_i^{start\_o} = \sigma(\mathbf{W}^r_{start}(\mathbf{x}_i + \mathbf{v}^k_{sub}) + \mathbf{b}^r_{start})$$
$$p_i^{end\_o} = \sigma(\mathbf{W}^r_{end}(\mathbf{x}_i + \mathbf{v}^k_{sub}) + \mathbf{b}^r_{end})$$

$$p_i^{start\_s} = \sigma(\mathbf{W}_{start}\mathbf{x}_i + \mathbf{b}_{start})$$
$$p_i^{end\_s} = \sigma(\mathbf{W}_{end}\mathbf{x}_i + \mathbf{b}_{end})$$

$$\mathbf{h}_0 = \mathbf{S}\mathbf{W}_s + \mathbf{W}_p$$
$$\mathbf{h}_\alpha = Trans(\mathbf{h}_{\alpha-1}), \alpha \in [1, N]$$

Experiments

- Datasets

Category	NYT		WebNLG	
	Train	Test	Train	Test
Normal	37013	3266	1596	246
EPO	9782	978	227	26
SEO	14735	1297	3406	457
ALL	56195	5000	5019	703

Table 1: Statistics of datasets. Note that a sentence can belong to both *EPO* class and *SEO* class.

Experiments

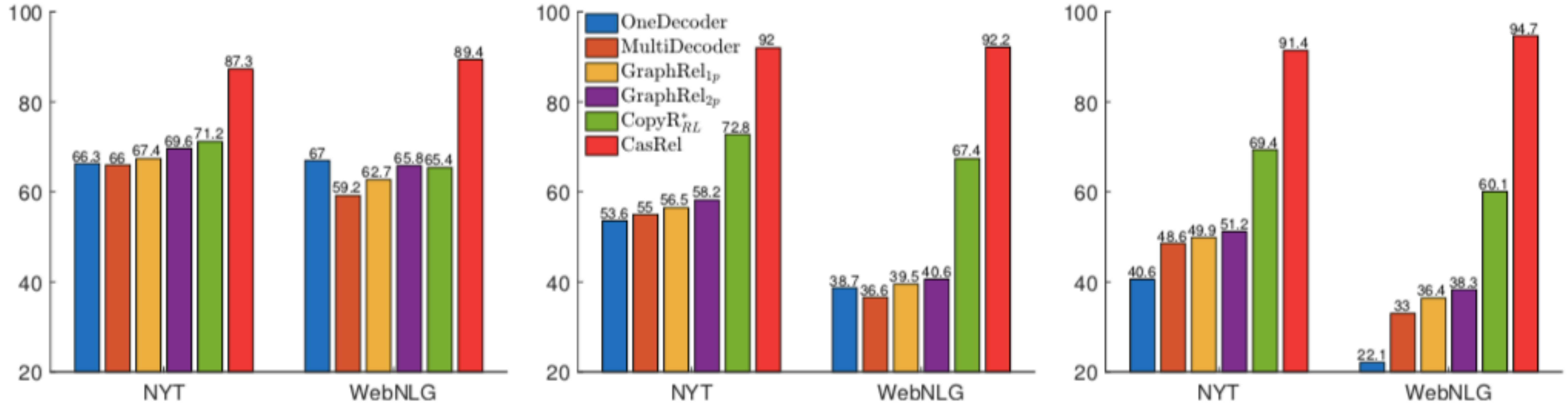
- Performance

Method	NYT			WebNLG		
	Prec.	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*
NovelTagging (Zheng et al., 2017)	62.4	31.7	42.0	52.5	19.3	28.3
CopyR$_{OneDecoder}$ (Zeng et al., 2018)	59.4	53.1	56.0	32.2	28.9	30.5
CopyR$_{MultiDecoder}$ (Zeng et al., 2018)	61.0	56.6	58.7	37.7	36.4	37.1
GraphRel$_{1p}$ (Fu et al., 2019)	62.9	57.3	60.0	42.3	39.2	40.7
GraphRel$_{2p}$ (Fu et al., 2019)	63.9	60.0	61.9	44.7	41.1	42.9
CopyR$_{RL}$ (Zeng et al., 2019)	77.9	67.2	72.1	63.3	59.9	61.6
CopyR$^{*}_{RL}$	72.8	69.4	71.1	60.9	61.1	61.0
CasRel$_{random}$	81.5	75.7	78.5	84.7	79.5	82.0
CasRel$_{LSTM}$	84.2	83.0	83.6	86.9	80.6	83.7
CasRel	**89.7**	**89.5**	**89.6**	**93.4**	**90.1**	**91.8**

Table 2: Results of different methods on NYT and WebNLG datasets. Our re-implementation is marked by *.

Experiments

- Detailed Results



(a) F1 of Normal Class (b) F1 of EntityPairOverlap Class (c) F1 of SingleEntityOverlap Class

Figure 3: F1-score of extracting relational triples from sentences with different overlapping pattern.

Method	NYT					WebNLG				
	N=1	*N=2*	*N=3*	*N=4*	*N≥5*	*N=1*	*N=2*	*N=3*	*N=4*	*N≥5*
$\text{CopyR}_{OneDecoder}$	66.6	52.6	49.7	48.7	20.3	65.2	33.0	22.2	14.2	13.2
$\text{CopyR}_{MultiDecoder}$	67.1	58.6	52.0	53.6	30.0	59.2	42.5	31.7	24.2	30.0
GraphRel_{1p}	69.1	59.5	54.4	53.9	37.5	63.8	46.3	34.7	30.8	29.4
GraphRel_{2p}	71.0	61.5	57.4	55.1	41.1	66.0	48.3	37.0	32.1	32.1
CopyR^{*}_{RL}	71.7	72.6	72.5	77.9	45.9	63.4	62.2	64.4	57.2	55.7
CasRel	**88.2**	**90.3**	**91.9**	**94.2**	**83.7** (+37.8)	**89.3**	**90.8**	**94.2**	**92.4**	**90.9** (+35.2)

Table 3: F1-score of extracting relational triples from sentences with different number (denoted as N) of triples.

Supplemental Experiments

Category	ACE04	NYT10-HRL		NYT11-HRL		Wiki-KBP	
	ALL	Train	Test	Train	Test	Train	Test
Normal	1604	59396	2963	53395	368	57020	265
EPO	8	5376	715	2100	0	3217	4
SEO	561	8772	742	7365	1	21238	20
ALL	2171	70339	4006	62648	369	79934	289

Table 6: Statistics of datasets. Note that a sentence can belong to both *EPO* class and *SEO* class.

Method	*Partial Match*									*Exact Match*		
	ACE04			NYT10-HRL			NYT11-HRL			Wiki-KBP		
	Prec.	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*	*Prec.*	*Rec.*	*F1*
Chan and Roth (2011)	42.9	38.9	40.8	–	–	–	–	–	–	–	–	–
MultiR (Hoffmann et al., 2011)	–	–	–	–	–	–	32.8	30.6	31.7	30.1	53.0	38.0
DS-Joint (Li and Ji, 2014)	**64.7**	38.5	48.3	–	–	–	–	–	–	–	–	–
FCM (Gormley et al., 2015)	–	–	–	–	–	–	43.2	29.4	35.0	–	–	–
SPTree (Miwa and Bansal, 2016)	–	–	–	49.2	55.7	52.2	52.2	54.1	53.1	–	–	–
CoType (Ren et al., 2017)	–	–	–	–	–	–	48.6	38.6	43.0	31.1	**53.7**	38.8
Katiyar and Cardie (2017)	50.2	**48.8**	49.3	–	–	–	–	–	–	–	–	–
NovelTagging (Zheng et al., 2017)	–	–	–	59.3	38.1	46.4	46.9	48.9	47.9	**53.6**	30.3	38.7
ReHession (Liu et al., 2017)	–	–	–	–	–	–	–	–	–	36.7	49.3	42.1
CopyR (Zeng et al., 2018)	–	–	–	56.9	45.2	50.4	34.7	53.4	42.1	–	–	–
HRL (Takanobu et al., 2019)	–	–	–	71.4	58.6	64.4	**53.8**	53.8	53.8	–	–	–
PA-LSTM-CRF (Dai et al., 2019)	–	–	–	–	–	–	–	–	–	51.1	39.3	44.4
CasRel	57.2	47.6	**52.0**	**77.7**	**68.8**	**73.0**	50.1	**58.4**	**53.9**	49.8	42.7	**45.9**

Table 5: Relational triple extraction results of different methods under *Partial Match* and *Exact Match* metrics.

Conclusion

- A Fresh Perspective to RE task
  - Model relations as functions that map subjects to objects
  - Simultaneously extract multiple relational triples from sentences
  - Without the overlapping problem

[WWW19]DualRE
Learning Dual Retrieval Module for Semi-supervised Relation Extraction

Hongtao Lin[1], Jun Yan[2], Meng Qu[3], Xiang Ren[1]

[1] University of Southern California,
[2] Tsinghua University
[3] University of Illinois at Urbana-Champaign

Speaker: 杨晰

xyang41@stu.ecnu.edu.cn

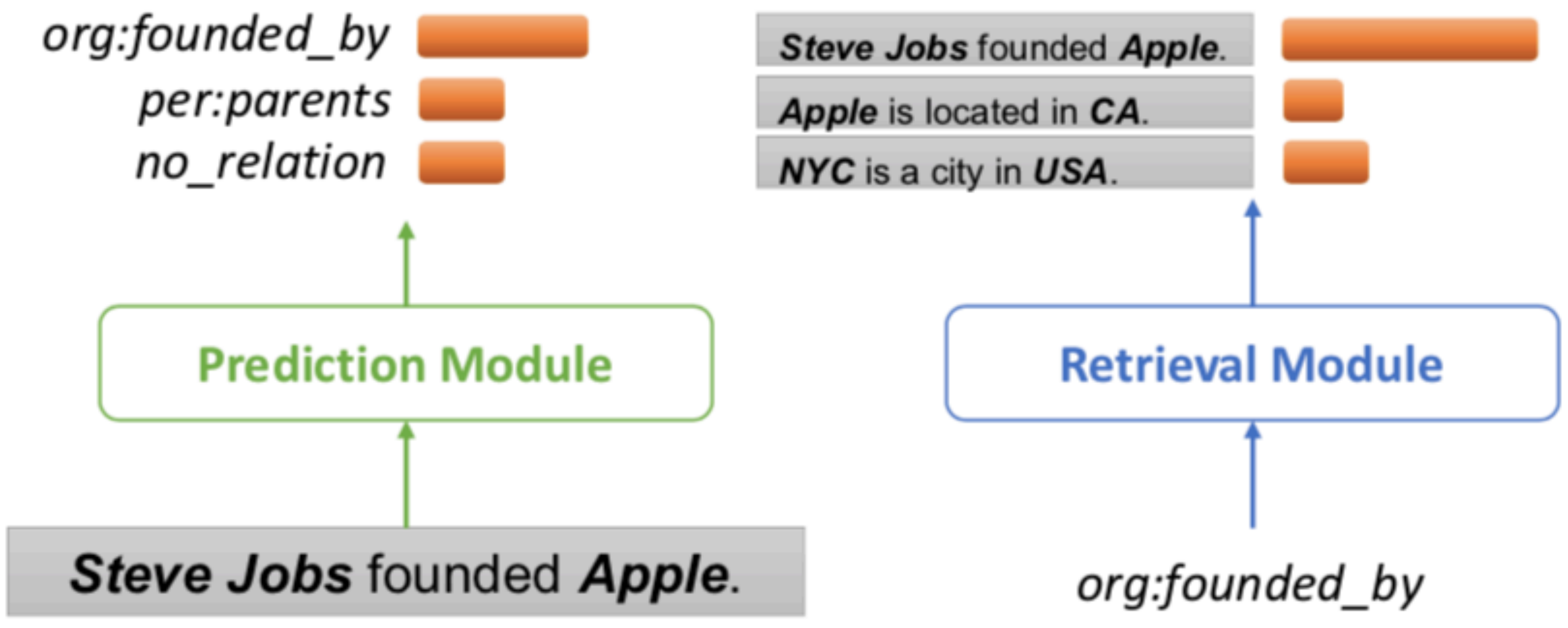
Outline

- Motivation
- Framework
- Experiments
- Conclusion

Motivationl

- Semi-Supervised RE
  - Self-Ensemble Methods: Insufficient Supervision
  - Self-Training Methods: Semantic Drift

MotivationII

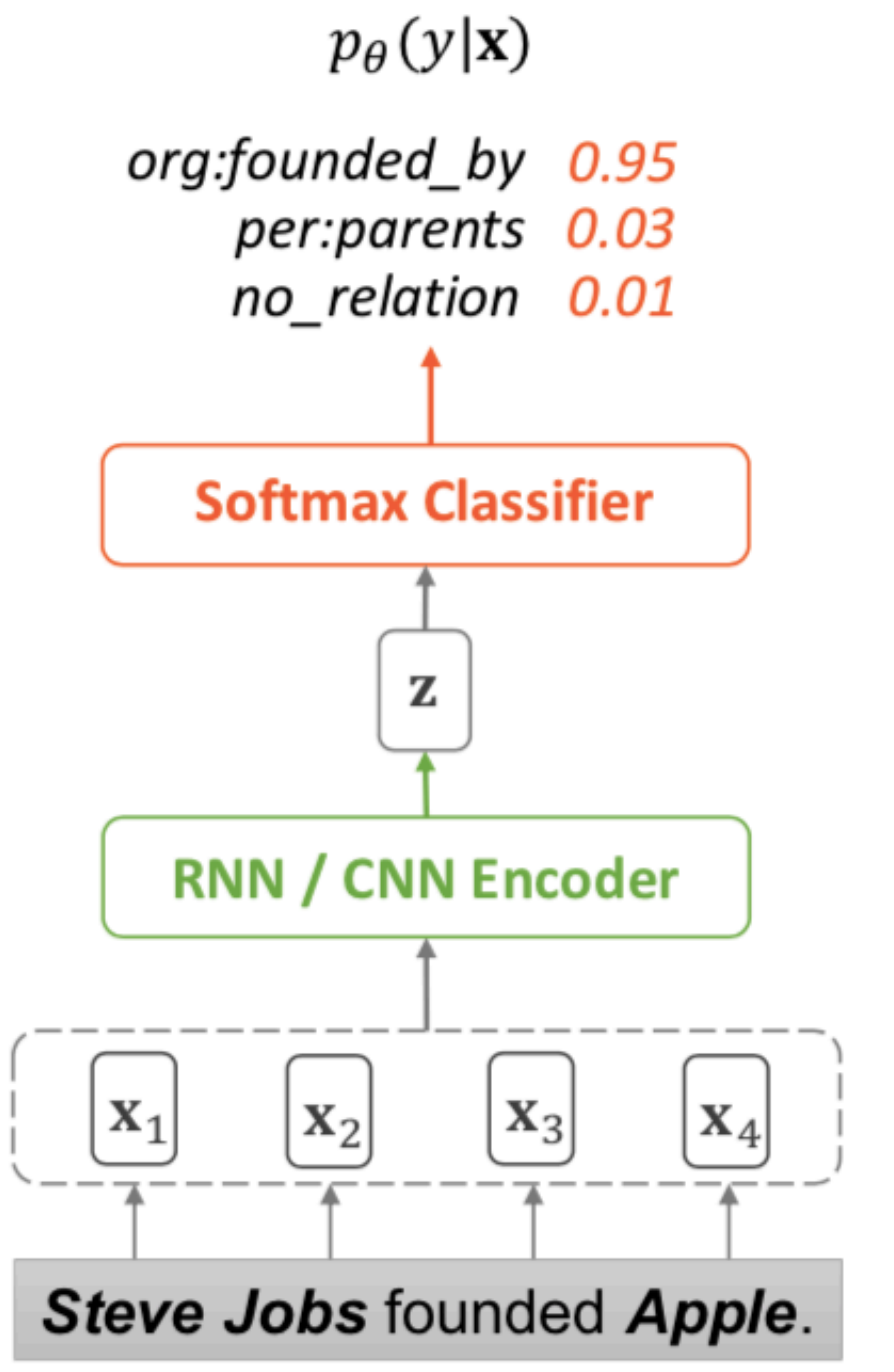


- Dual Task
  - Predicting the relation expressed in a sentence | Retrieving sentences for a given relation
  - Annotate/Retrieve unlabeled sentences ➡️ the insufficient supervision
  - Joint Learning ➡️ Generate high-quality labeled date ➡️ semantic drift

The DualRE Framework

- Relation Prediction Module
- Sentence Retrieval Module
- Interaction Between the Two Modules

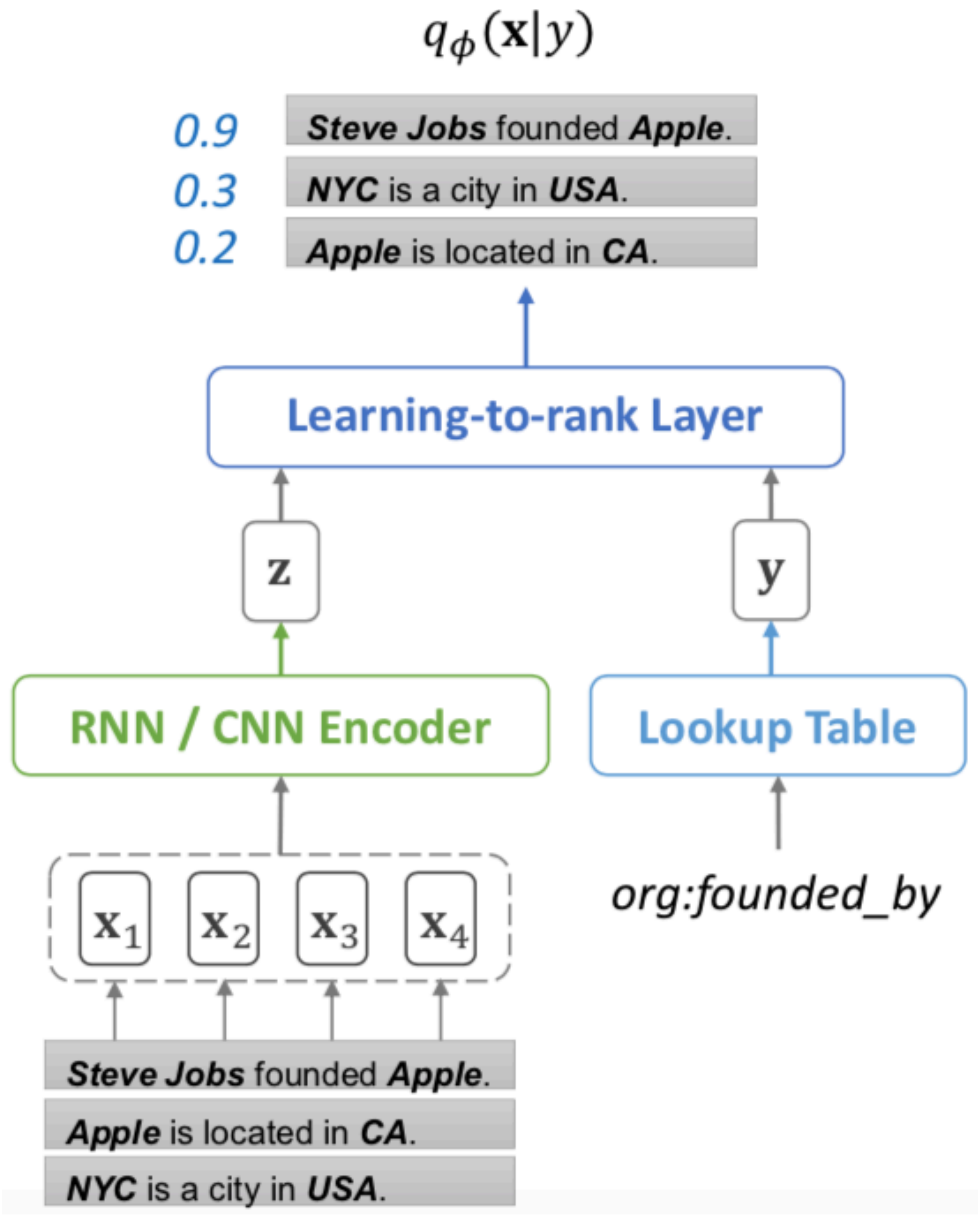
Relation Prediction Module



Object function on labeled data:

$$\mathbf{O}_P = \mathbb{E}_{(\mathbf{x},y)\in L}[\log p_\theta(y|\mathbf{x})]$$

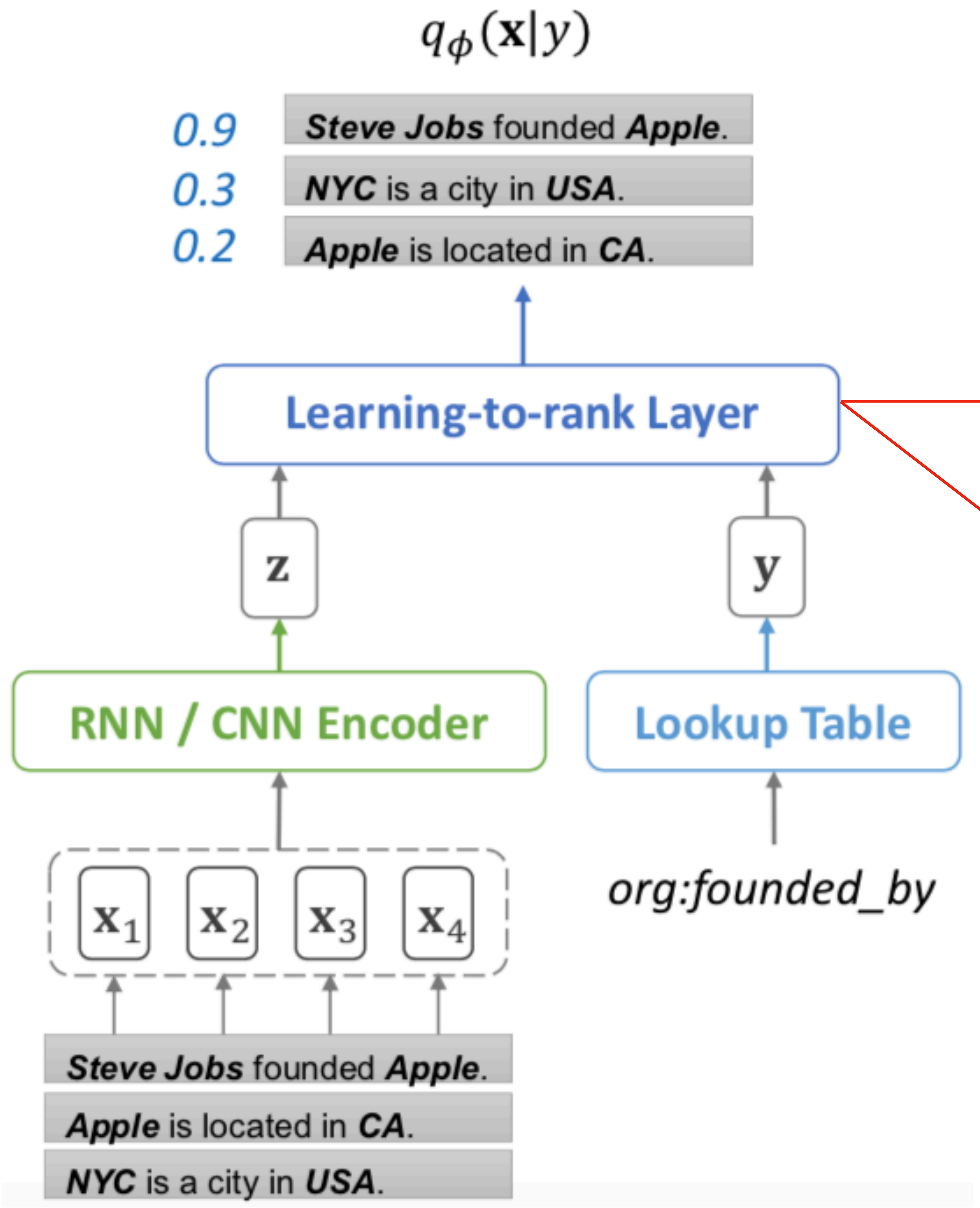
Sentence Retrieval Module



Object function on labeled data:

$$\mathbf{O}_R = \mathbb{E}_{(\mathbf{x}, y)\in L}[\log q_\phi(\mathbf{x}, y)]$$

Sentence Retrieval Module



Object function on labeled data:

$$\mathrm{O}_R = \mathbb{E}_{(\mathbf{x},y)\in L}[\log q_\phi(\mathbf{x},y)]$$

1) **Pointwise Approach**

$$\mathbb{E}_{(\mathbf{x},y)\in L}[\log \sigma(\mathbf{z}^\top \mathbf{y})] + \mathbb{E}_{(\mathbf{x},y')\notin L}[\log(1-\sigma(\mathbf{z}^\top \mathbf{y}'))]$$

2) **Pairwise Approach**

$$\mathbb{E}_{(\mathbf{x},y),(\mathbf{x}',y')\in L\times L}[r(\mathbf{x},\mathbf{x}',y)\log \sigma(\mathbf{z}^T\mathbf{y}-\mathbf{z}'^T\mathbf{y})]$$

Interaction Between the Two Modules

Object function on unlabeled data:

$$\mathbf{O}_U = \mathbb{E}_{\mathbf{x} \in U}[\log p(\mathbf{x})]$$
$$\geq \mathbb{E}_{\mathbf{x} \in U, y \sim p_\theta(y|\mathbf{x})}[\log \frac{q_\phi(\mathbf{x}, y)}{p_\theta(y|\mathbf{x})}]$$

Interaction Between the Two Modules

Object function on unlabeled data:

$$\mathbf{O}_U = \mathbb{E}_{\mathbf{x}\in U}[\log p(\mathbf{x})]$$
$$\geq \mathbb{E}_{\mathbf{x}\in U, y\sim p_\theta(y|\mathbf{x})}[\log \frac{q_\phi(\mathbf{x}, y)}{p_\theta(y|\mathbf{x})}]$$

Jensen 不等式得到下界，优化下界过程中包含了两个模块

Interaction Between the Two Modules

Object function on unlabeled data:

$$\mathbf{O}_U = \mathbb{E}_{\mathbf{x}\in U}[\log p(\mathbf{x})]$$

$$\geq \mathbb{E}_{\mathbf{x}\in U, y\sim p_\theta(y|\mathbf{x})}[\log \frac{q_\phi(\mathbf{x}, y)}{p_\theta(y|\mathbf{x})}]$$

Jensen 不等式得到下界，优化下界过程中包含了两个模块

可看作最小化KL(p(y|x) || q(y|x))

Joint Learning Algorithm: EM-based

Algorithm 1: DualRE Learning Algorithm.

Input: Labeled data $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_L}$, unlabeled data $U = \{\mathbf{x}_j\}_{j=1}^{N_U}$, the amount of data to retrieve each in each iteration k.

Initialize: $L_U \leftarrow \emptyset$.

$P_\theta, Q_\phi \leftarrow$ Pretrain prediction and retrieval module using L.

while $U \neq \emptyset$ ***and*** *not converge* **do**

- $L' \leftarrow$ Retrieve k annotated instances $(\mathbf{x}, y)$ from U (Sec. 4.2).
- Remove instances L' from U and add them to L_U.
- // Update prediction module:
- Optimize P_θ using data from both L and L_U (Eq. 8).
- // Update retrieval module:
- Optimize Q_ϕ using data from both L and L_U (Eq. 9).

end

Joint Learning Algorithm: EM-based

Algorithm 1: DualRE Learning Algorithm.

Input: Labeled data $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_L}$, unlabeled data $U = \{\mathbf{x}_j\}_{j=1}^{N_U}$, the amount of data to retrieve each in each iteration k.

Initialize: $L_U \leftarrow \emptyset$.

$P_\theta, Q_\phi \leftarrow$ Pretrain prediction and retrieval module using L.

while $U \neq \emptyset$ ***and*** *not converge* **do**

    $L' \leftarrow$ Retrieve k annotated instances $(\mathbf{x}, y)$ from U (Sec. 4.2).

    Remove instances L' from U and add them to L_U.

    // Update prediction module:

    Optimize P_θ using data from both L and L_U (Eq. 8).

    // Update retrieval module:

    Optimize Q_ϕ using data from both L and L_U (Eq. 9).

end

两个模块预测结果的**交集**

Joint Learning Algorithm: EM-based

Algorithm 1: DualRE Learning Algorithm.

Input: Labeled data $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_L}$, unlabeled data $U = \{\mathbf{x}_j\}_{j=1}^{N_U}$, the amount of data to retrieve each in each iteration k.

Initialize: $L_U \leftarrow \emptyset$.

$P_\theta, Q_\phi \leftarrow$ Pretrain prediction and retrieval module using L.

while $U \neq \emptyset$ ***and*** *not converge* **do**

  $L' \leftarrow$ Retrieve k annotated instances $(\mathbf{x}, y)$ from U (Sec. 4.2).

  Remove instances L' from U and add them to L_U.

  // Update prediction module:

E Step：

  Optimize P_θ using data from both L and L_U (Eq. 8).

  // Update retrieval module:

  Optimize Q_ϕ using data from both L and L_U (Eq. 9).

end

$$\boxed{\begin{aligned}\nabla_\theta \mathbf{O} &= \mathbb{E}_{(\mathbf{x},y)\in L}[\nabla_\theta \log p_\theta(y|\mathbf{x})] \\ &+ \mathbb{E}_{\mathbf{x}\in U, y\sim(q_\phi(y|\mathbf{x})+p_\theta(y|\mathbf{x}))}[\nabla_\theta \log p_\theta(y|\mathbf{x})]\end{aligned}}$$

Joint Learning Algorithm: EM-based

Algorithm 1: DualRE Learning Algorithm.

Input: Labeled data $L = \{(\mathbf{x}_i, y_i)\}_{i=1}^{N_L}$, unlabeled data $U = \{\mathbf{x}_j\}_{j=1}^{N_U}$, the amount of data to retrieve each in each iteration k.

Initialize: $L_U \leftarrow \emptyset$.

$P_\theta, Q_\phi \leftarrow$ Pretrain prediction and retrieval module using L.

while $U \neq \emptyset$ ***and*** *not converge* **do**

  $L' \leftarrow$ Retrieve k annotated instances $(\mathbf{x}, y)$ from U (Sec. 4.2).

  Remove instances L' from U and add them to L_U.

  // Update prediction module:

  Optimize P_θ using data from both L and L_U (Eq. 8).

  // Update retrieval module:

M Step： Optimize Q_ϕ using data from both L and L_U (Eq. 9).

end

$$\nabla_\phi O = \mathbb{E}_{(\mathbf{x},y)\in L}[\nabla_\phi \log q_\phi(\mathbf{x}, y)] + \mathbb{E}_{(\mathbf{x},y)\sim(p_\theta(\mathbf{x},y)+q_\phi(\mathbf{x},y))}[\nabla_\phi \log q_\phi(\mathbf{x}, y)]$$

Experiments

- Performance on SemEval

Methods / % Labeled Data	5% Precision	5% Recall	5% F_1	10% Precision	10% Recall	10% F_1	30% Precision	30% Recall	30% F_1
LSTM [17]	25.31 ± 2.16	20.72 ± 3.91	22.71 ± 3.31	36.91 ± 6.10	29.85 ± 7.25	32.92 ± 6.71	64.83 ± 0.69	63.03 ± 0.67	63.91 ± 0.66
PCNN [37]	42.95 ± 4.69	40.79 ± 4.59	41.84 ± 4.63	53.78 ± 1.51	49.11 ± 2.22	51.32 ± 1.74	64.54 ± 0.58	62.98 ± 0.48	63.75 ± 0.33
PRNN [39]	56.16 ± 1.32	54.87 ± 1.49	55.49 ± 0.90	61.70 ± 1.16	63.61 ± 2.07	62.63 ± 1.42	69.66 ± 2.19	68.76 ± 2.60	69.14 ± 1.02
Mean-Teacher (PRNN) [33]	53.71 ± 4.43	49.54 ± 3.29	51.51 ± 3.58	62.43 ± 1.28	60.34 ± 0.62	61.36 ± 0.75	68.65 ± 0.64	69.84 ± 0.65	69.24 ± 0.56
Self-Training (PRNN) [29]	56.47 ± 1.11	56.14 ± 1.33	56.30 ± 0.96	64.27 ± 2.37	63.48 ± 2.02	63.79 ± 0.28	68.95 ± 0.68	72.63 ± 0.82	70.74 ± 0.58
RE-Ensemble (PRNN)	58.77 ± 0.58	58.50 ± 0.97	58.63 ± 0.62	65.10 ± 0.84	64.57 ± 0.54	64.83 ± 0.61	70.26 ± 0.92	73.20 ± 1.22	71.69 ± 0.47
DualRE-Pairwise (PRNN)	59.76 ± 0.47	63.36 ± 0.77	**61.51 ± 0.56**	64.39 ± 0.75	67.70 ± 0.80	66.00 ± 0.48	70.05 ± 0.53	74.83 ± 0.88	72.36 ± 0.60
DualRE-Pointwise (PRNN)	58.73 ± 1.50	62.23 ± 1.93	60.43 ± 1.67	64.50 ± 1.14	67.67 ± 1.66	**66.03 ± 1.00**	70.03 ± 0.74	74.87 ± 0.75	**72.36 ± 0.35**
RE-Gold (PRNN w. gold labels)*	72.57 ± 1.47	74.65 ± 1.98	73.56 ± 0.31	71.40 ± 1.42	76.72 ± 0.64	73.95 ± 0.50	72.98 ± 0.96	78.86 ± 0.76	75.80 ± 0.24

Table 2: Performance comparison on SemEval [15] with various amounts of labeled data and 50% unlabeled data. We report the mean and standard deviation of the evaluation metrics by conducting 5 runs of training and testing using different random seeds. DualRE outperforms all the baseline methods.

Experiments

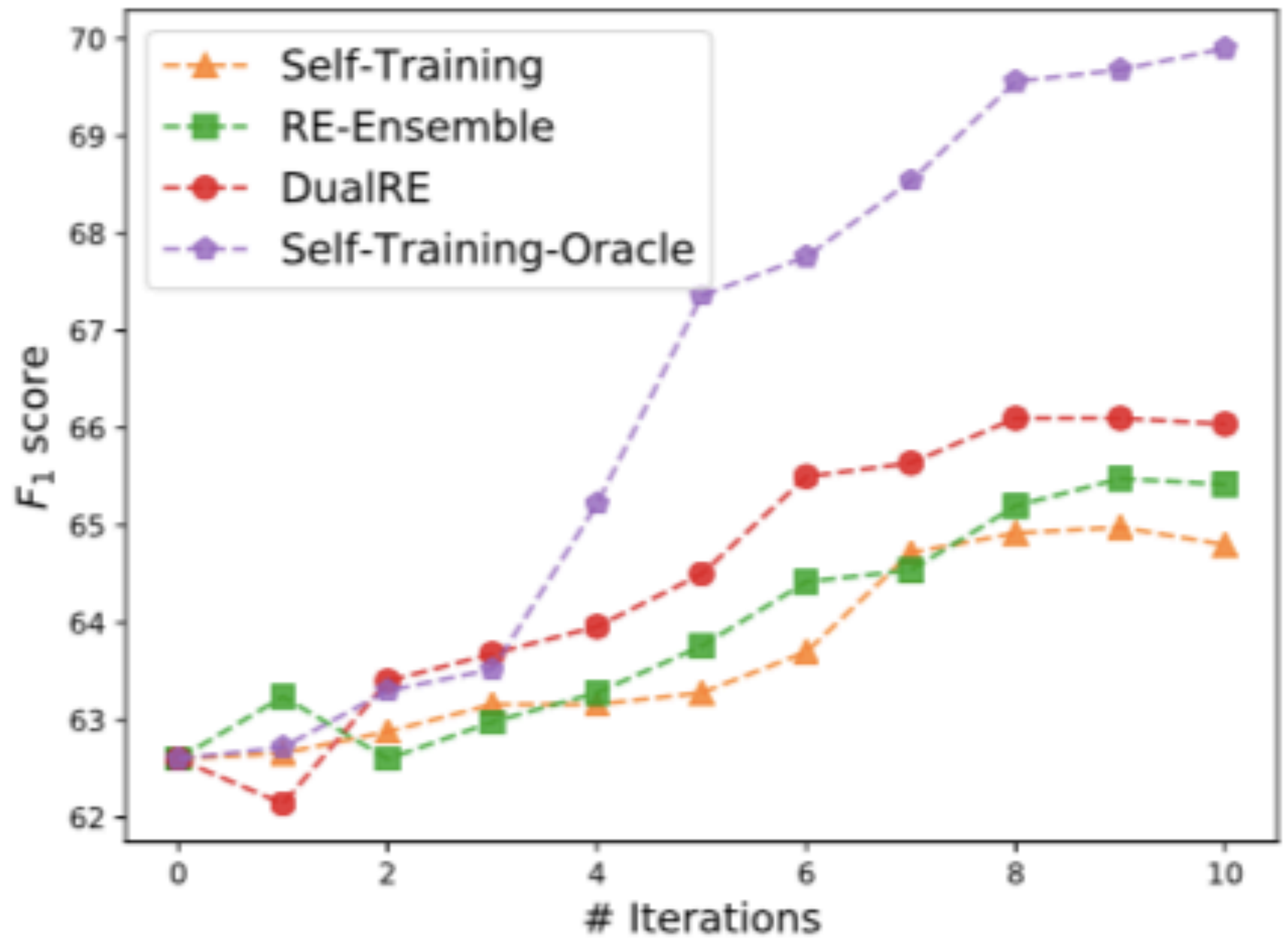
- Performance on TACRED

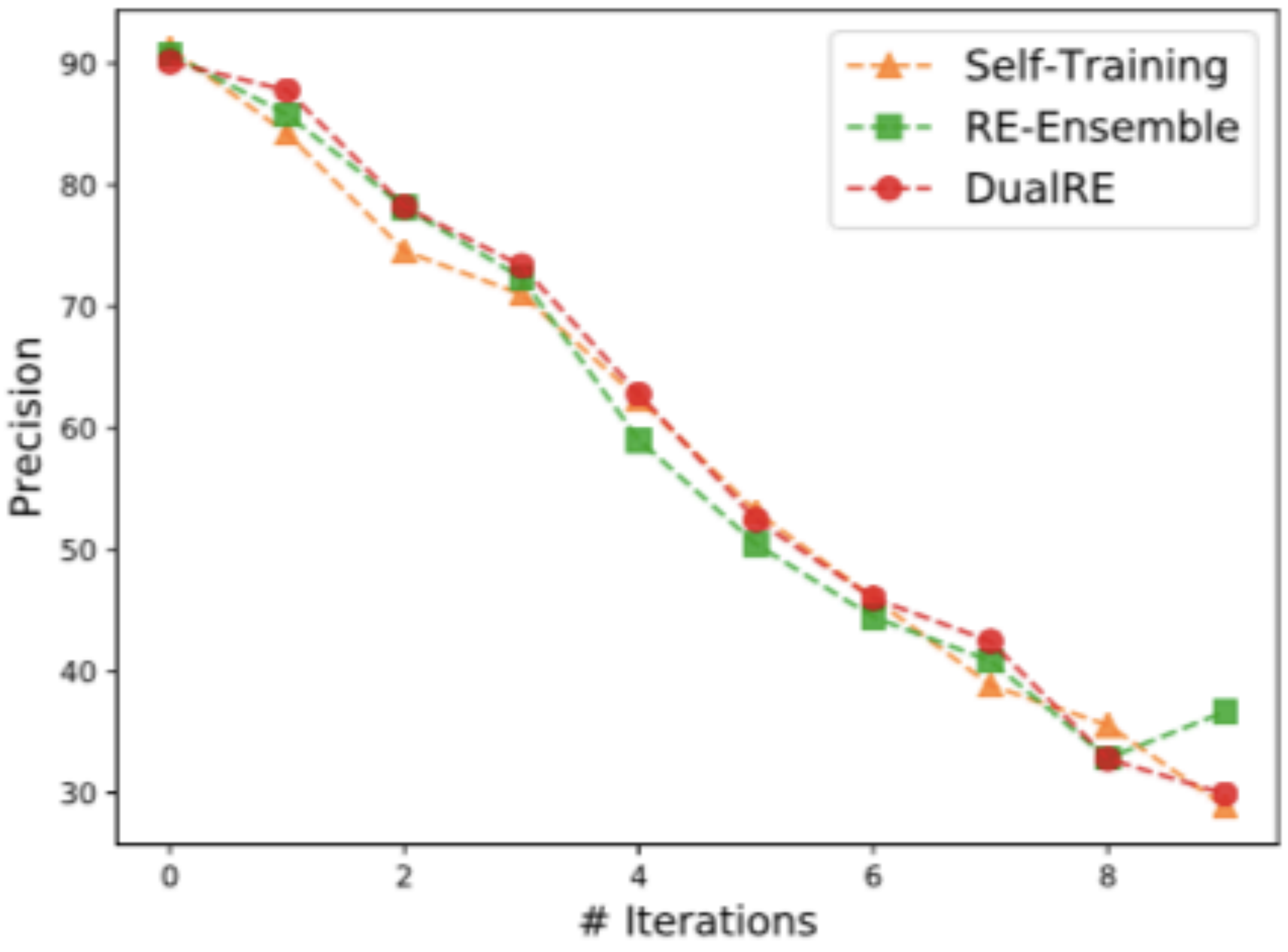
Methods / % Labeled Data	3%			10%			15%		
	Precision	Recall	F_1	Precision	Recall	F_1	Precision	Recall	F_1
LSTM [17]	40.76 ± 6.62	23.05 ± 4.48	28.62 ± 3.01	50.56 ± 0.98	43.09 ± 1.26	46.51 ± 0.90	55.11 ± 0.56	44.41 ± 0.58	49.18 ± 0.52
PCNN [37]	58.16 ± 7.74	37.49 ± 6.16	44.64 ± 1.59	64.32 ± 7.78	42.06 ± 4.94	50.16 ± 1.15	67.10 ± 0.44	42.88 ± 0.42	52.32 ± 0.30
PRNN [39]	48.93 ± 5.72	33.05 ± 2.19	39.16 ± 0.90	53.44 ± 2.82	51.77 ± 1.88	52.49 ± 0.64	58.88 ± 2.32	51.30 ± 2.04	54.76 ± 0.93
Mean-Teacher (PRNN) [33]	53.08 ± 3.55	41.81 ± 0.61	**46.74 ± 1.70**	58.53 ± 2.56	50.08 ± 1.14	53.94 ± 0.91	57.90 ± 1.09	52.64 ± 0.97	55.13 ± 0.05
Self-Training (PRNN) [29]	49.89 ± 1.05	39.23 ± 2.26	43.86 ± 1.26	56.54 ± 0.72	53.00 ± 0.49	54.71 ± 0.09	60.09 ± 0.43	54.77 ± 0.55	57.31 ± 0.47
RE-Ensemble (PRNN)	56.48 ± 0.95	36.90 ± 0.91	44.62 ± 0.39	61.26 ± 0.58	52.51 ± 0.56	55.54 ± 0.29	60.76 ± 0.78	55.00 ± 1.04	57.72 ± 0.38
DualRE-Pairwise (PRNN)	58.97 ± 0.96	34.55 ± 1.18	43.55 ± 0.67	63.10 ± 0.94	48.91 ± 0.93	55.09 ± 0.25	60.99 ± 1.39	54.04 ± 0.46	57.30 ± 0.81
DualRE-Pointwise (PRNN)	52.76 ± 2.58	38.99 ± 2.08	44.73 ± 0.66	61.61 ± 1.30	52.30 ± 0.89	**56.56 ± 0.42**	60.66 ± 1.57	56.65 ± 0.37	**58.58 ± 0.69**
RE-Gold (PRNN w. gold labels)*	64.38 ± 0.46	60.35 ± 0.81	62.30 ± 0.29	65.88 ± 0.66	61.65 ± 0.42	63.70 ± 0.53	66.95 ± 2.78	59.97 ± 3.12	63.13 ± 0.56

Table 3: Performance comparison on TACRED [39] with various amounts of labeled data and 50% unlabeled data. We report the mean and standard deviation of the evaluation metrics by conducting 3 runs of training and testing using different random seeds. DualRE outperforms all the baseline methods except Mean-Teacher at one data point.

Experiments

- Analysis on Quality of Retrieved Instances





Conclusion

- Dual Task
  - the primal Prediction Task & the dual Retrieval Task
  - mutually enhance each other
  - Extending to deal with various text classification tasks…

Q & A