Relation Extraction with Prompt

——探讨prompt方法在mention-level实体关系抽取任务上的应用

梁怡萍

2021.11.25

paper list

题目	单位机构	年份	页码	作者
AUTOPROMPT: Eliciting Knowledge from Language Modelswith Automatically Generated Prompts	University of California	ACL 2020	14	Taylor Shin Yasaman Razeghi et al.
AdaPrompt: Adaptive Prompt-based Finetuning for Relation Extraction	Zhejiang University Alibaba Group	2021	9	Xiang Chen Xin Xie et al.
PTR: Prompt Tuning with Rules for Text Classification	Tsinghua University	preprint 2021	10	Xu Han Weilin Zhao Ning Ding et al.
Label Verbalization and Entailmentfor Effective Zero- and Few-Shot Relation Extraction	University of the Basque Country	EMNLP 2021	14	Oscar Sainz Oier Lopez de Lacalle et al.
KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimizationfor Relation Extraction	Zhejiang University Alibaba Group	AAAI 2022	9	Xiang Chen Ningyu Zhang et al.

Mention-level Relation Extraction

预测一段文本中两个指定实体<sub, obj>之间的关系，两实体顺序不能颠倒

Context	Entity-pair	Relation
Jobs founded Apple.	<Apple, Jobs>	org:founded_by
She gave birth to Johnin a hospital in Michigan.	<she, John>	per:parent
She gave birth to Johnin a hospital in Michigan.	<John, she>	per:children

以TACRED数据集为例

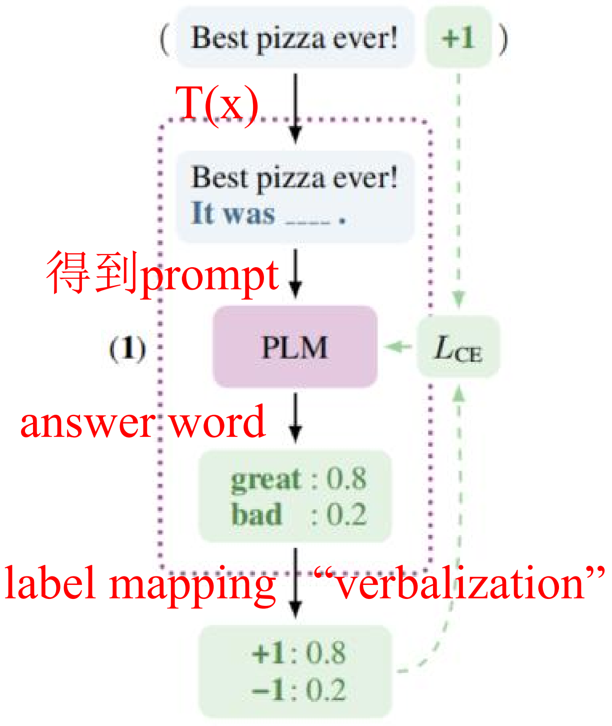
Scenario： 实体的类别已知；
实体对 被标记了一种预定义的关系类型，如 org:founded_by

key points in this scenario：

1. Relation Detection and Characterization 关系检测与表征
2. RE严重依赖于提及实体的信息：实体表示 和 上下文 是支持预测的主要来源
3. 对 no_relation 的检测
4. 语义相似的关系的检测：如：per:country_of_residence / per: origin
5. 含义宽泛的关系的检测：如：per:other_family
6. 在low-resourse setting 下的效果、对未标注数据的处理和利用

RE with Prompt method

general idea



$$p(y|X_{\text{prompt}}) = \sum_{w \in \mathcal{V}_y} p(\texttt{[MASK]} = w|X_{\text{prompt}})$$

advantages

- 不引入大量的额外参数，只存储每个单独任务的提示，并对任何任务的输入使用相同的预训练模型
- low-resource setting下，prompt引导的LM知识可以弥补数据少而缺乏的信息

Apply to RE：

- RE的标签空间很大、复杂，使得verbalization过程困难
- 手动为每种关系找到合适的模板、合适的标签来区分不同的类需要领域知识和成本
- 对于自动生成的大量prompt，验证它们的有效性需要大量额外数据和计算成本，在few-shot setting下不可靠，且效果与手工模板基本相当，很少有能超过手工模板；
- 而完全连续的提示在具有数十亿个参数的大规模plm上很有效，但不能在规模一般的plm上稳定工作

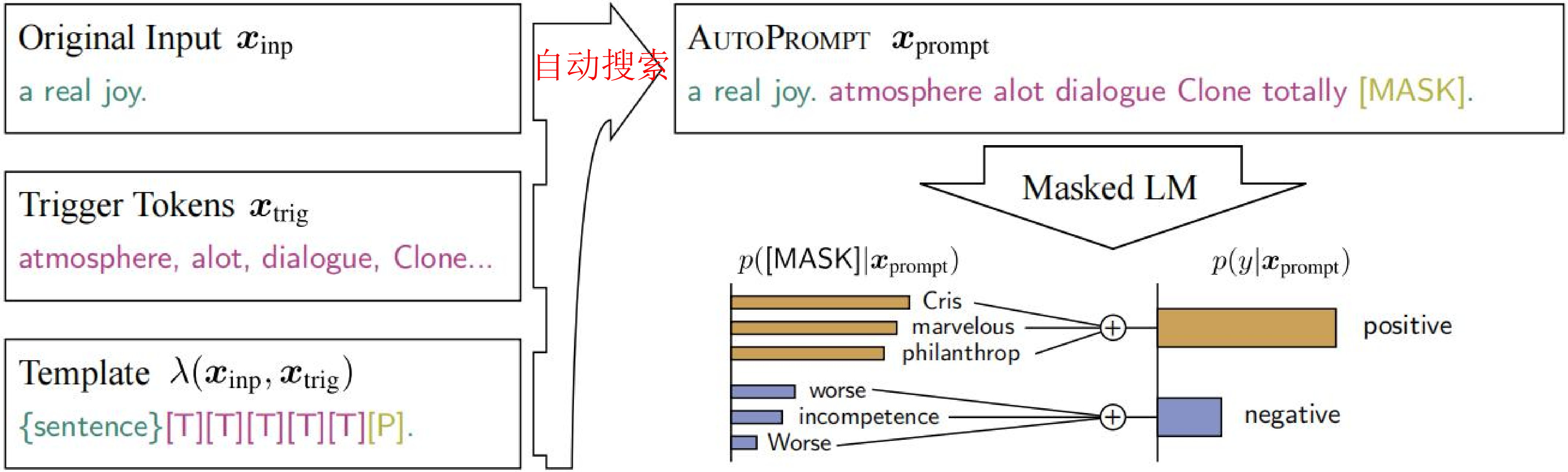
Overview

	AutoPrompt	AdaPrompt	PTR	LVE	KnowPrompt
关系表征方式	用trigger word引导LM中的关系知识	分解relation label	手工设计自然语言表示的同义短语	手工设计template代替关系信息	借助relation embedding
Template设计	基于梯度自动搜索hard tokens	所有关系使用相似手工设计的template：The relation between A and B is [MASK] / A is [MASK] B	手工设计sub-prompt用逻辑规则构造prompt	每种关系手工构造1~8个	Soft template words with knowledge
Verbalization设计	自动搜索 hard token	分解relation label后的token作为answer word	手工设计与relation同义的短语	用NLI label代替	Relation embedding

Overview

<table>
<tr><th colspan="2"></th><th>AdaPrompt</th><th>PTR</th><th>LVE</th><th>KnowPrompt</th></tr>
<tr><td colspan="2">no_relation检测</td><td>直接由“no_relation”对应的template、Verbalization判定</td><td>直接由“no_relation”对应的template、Verbalization判定</td><td>其他关系都不明显则判定为no_relation</td><td>直接由“no_relation”对应的template、Verbalization判定</td></tr>
<tr><td rowspan="3">容易混淆的关系</td><td>state/city/country
per:parent/org:parent</td><td>鼓励LM还原mask掉的实体token</td><td>限制template word中实体类别[MASK]的Verbalization</td><td>指明关系实体类别[ORG:CITY][ORG:COUNTRY]</td><td rowspan="4">1.初始化template word时融入实体类别知识
2.初始化relation embedding时融入数据集的关系统计信息
3.使用辅助KE训练目标注入实体、关系信息</td></tr>
<tr><td>parent/children</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>found_by/found
member_of/member</td><td>-</td><td>关系逆转</td><td>-</td></tr>
<tr><td colspan="2">含义宽泛的关系</td><td>-</td><td>使用语义更宽泛的一个template：
‘s reletive is</td><td>构造被该语义包括的尽可能多的细粒度template</td></tr>
</table>

Method



note: 该方法可应用在多种下游任务，该示例以情感分析说明该方法设计思路

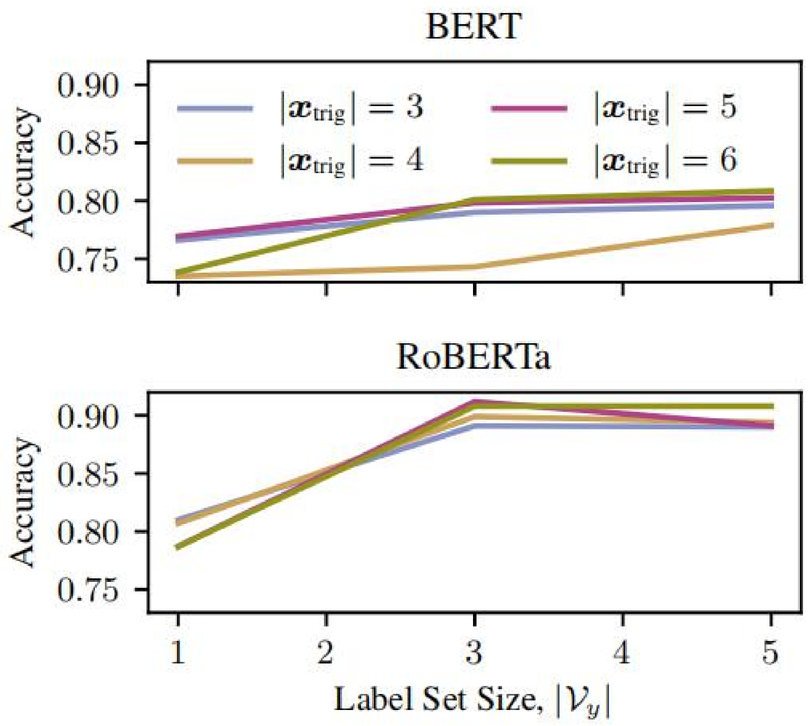
AUTOPROMPT: Eliciting Knowledge from Language Models with Automatically Generated Prompts

Template

$\{\text{sent}\}\{\text{sub}\}[\text{T}]\ldots[\text{T}][\text{P}].$

x_{prompt} 示例：

Leonard Wood (born February 4, 1942) is a former Canadian politician.
Leonard Wood gymnasium brotherdicative himself another [MASK].



$$\mathcal{V}_{\text{cand}} = \underset{w \in \mathcal{V}}{\text{top-}k} \left[\boldsymbol{w}_{\text{in}}^T \nabla \log p(y|\boldsymbol{x}_{\text{prompt}}) \right]$$

搜索方法：

$$p(y|\boldsymbol{x}_{\text{prompt}}) = \sum_{w \in \mathcal{V}_y} p([\text{MASK}] = w|\boldsymbol{x}_{\text{prompt}})$$

AUTOPROMPT: Eliciting Knowledge from Language Models with Automatically Generated Prompts

Answer-mapping

step1

$$\boldsymbol{h} = \text{Transformer}_{\text{enc}}(\tilde{\boldsymbol{x}})$$

$$p(y|\boldsymbol{h}^{(i)}) \propto \exp(\boldsymbol{h}^{(i)} \cdot \boldsymbol{y} + \beta_y)$$

step2

$$\mathcal{V}_y = \underset{w \in \mathcal{V}}{\text{top-}k} \left[s(y, w) \right]$$

$$s(y, w) \ = \ p(y|\boldsymbol{w}_{\text{out}})$$

AUTOPROMPT: Eliciting Knowledge from Language Models with Automatically Generated Prompts

Solve task

Relation	Model	Context and Prompt	Prediction
P103 (native language)	BERT	Alexandra Lamy (born 14 October 1971) is a French actress. Alexandra Lamy speaks airfield dripping % of [MASK].	French
P36 (capital)	RoBERTa	Kirk was born in Clinton County, Ohio, and he entered service in Wilmington, Ohio. Clinton County famously includes the zoo influencing [MASK].	Wilmington
P530 (diplomatic relation)	BERT	The Black Sea forms in an east-west trending elliptical depression which lies between Bulgaria, Georgia, Romania, Russia, Turkey, and Ukraine. Ukraine qualified some immigration actually entered [MASK].	Russia
P106 (occupation)	RoBERTa	Spencer Treat Clark (born September 24, 1987) is an American actor who has appeared in several films, including Gladiator, Mystic River, and Unbreakable. Spencer Treat Clark famously the famously handsome the [MASK].	Hulk

in T-REx dataset

AUTOPROMPT: Eliciting Knowledge from Language Models with Automatically Generated Prompts

Result

Model	Original	Perturbed
Supervised RE LSTM	57.95	58.81
BERT (LAMA)	69.06	28.02
BERT (LPAQA)	76.55	30.79
BERT (AUTOPROMPT)	90.73	56.43
RoBERTa (AUTOPROMPT)	60.33	28.95

P@1指标 in T-REx

Perturbed examples：

The Immortal Game was a chess game played by Adolf Anderssen and Lionel Kieseritzky on 21 June 1851 in ~~London~~Seoul, during a break of the first international tournament. The Immortal Game locatedstered regardless streets in [MASK].	Seoul
The Honda Civic del Sol is a 2-seater front-engined, front wheel drive, targa top car manufactured by ~~Honda~~Toyota in the 1990s. Honda Civic del Sol defy trademarks of namesake manufacturer [MASK].	Toyota
Mizeria is a Polish ~~salad~~sandwich consisting of thinly sliced or grated cucumbers, often with sour cream though in some cases oil. Mizeria is calls direcend altitude [MASK].	food

AUTOPROMPT: Eliciting Knowledge from Language Models with Automatically Generated Prompts

Template&Answer-mapping

x and y have the relation “per:parent”

<==>$f_{e_s}(x, person) \wedge f_{e_s,e_o}(x, \text{'s } parent\ was, y) \wedge f_{e_o}(y, person)$

给定context： $x =$ “$\ldots e_s \ldots e_o \ldots$” :

$T_{f_{e_s}}(x) =$ “x the [MASK] e_s”,

$\mathcal{V}_{f_{e_s}} = \{$“person”, “organization”, $\ldots\}$.

$T_{f_{e_s,e_o}}(x) =$ “ x e_s [MASK] e_o”,

$\mathcal{V}_{f_{e_s,e_o}} = \{$“’s parent was”, “was born in”, $\ldots\}$.

$T(x) = [T_{f_{e_s}}(x); T_{f_{e_o}}(x); T_{f_{e_s,e_o}}(x)] =$

“ x the [MASK]$_1$ e_s $\boxed{\text{[MASK]}_2}$ the [MASK]$_3$ e_o ”

$$p(y|x) = \prod_{j=1}^{n} p(\text{[MASK]}_j = \phi_j(y)|T(x))$$

V[MASK]1 = {“person”, “organization”, . . .}

V[MASK]2 = {“’s parent was”, “was born in”, . . .}

V[MASK]3 = {“person”, “organization”, . . .}

PTR: Prompt Tuning with Rules for Text Classification

Template&Answer-mapping

关系逆转：

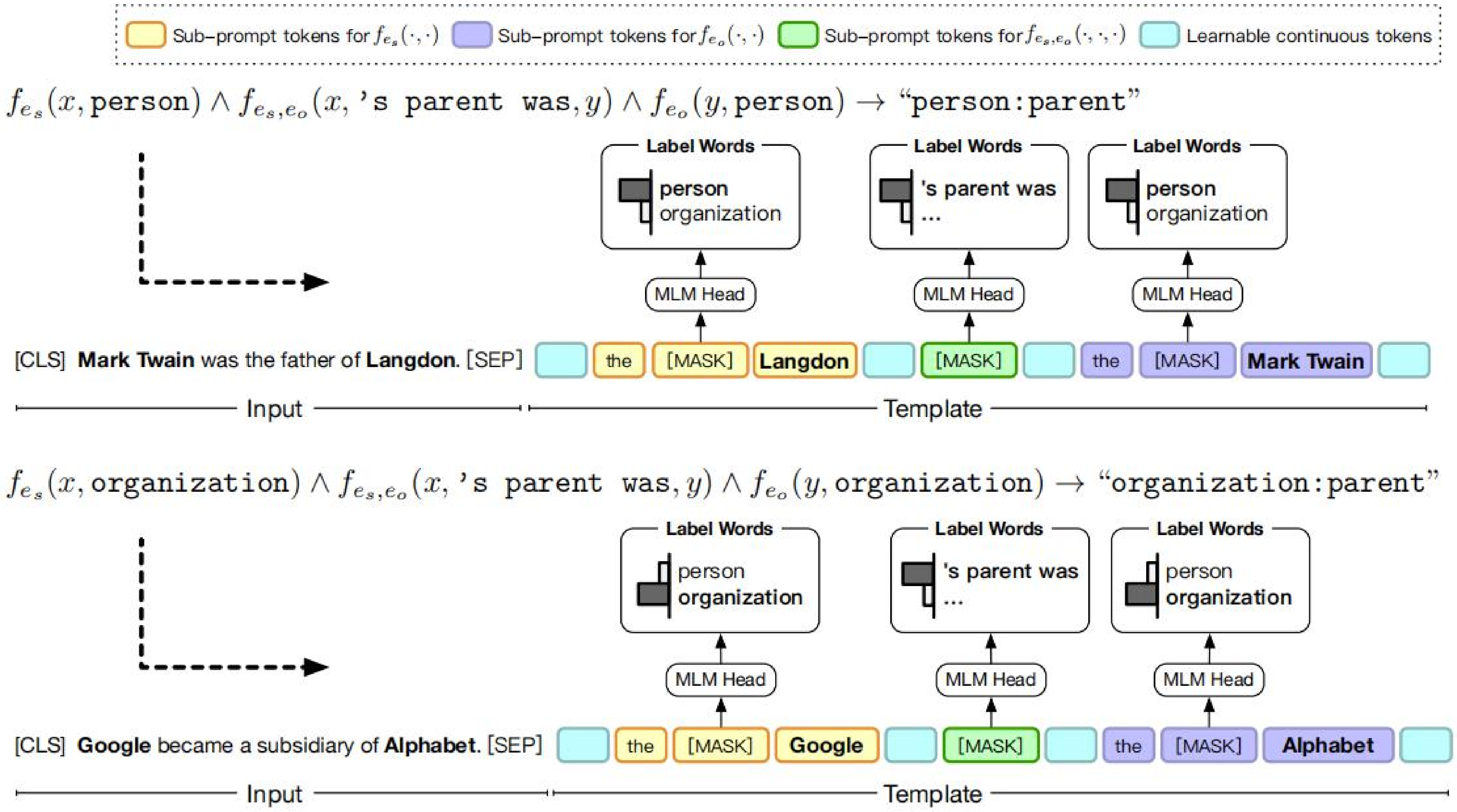
$$(E_s, \text{“}org\text{:}member\_of\text{”}, E_o) -> (E_o, \text{“}org\text{:}member\text{”}, E_s)$$

PTR	[CLS] Mark Twain was born in Florida . [P1] the [MASK] Mark Twain [MASK] the [MASK] Florida [P2] [SEP]	w/o	w/o	72.4
PTR (REVERSED)	[CLS] Mark Twain was born in Florida . [P1] the [MASK] Florida [MASK] the [MASK] Mark Twain [P2] [SEP]	w/o	w/o	**75.9**

手动逆转一部分关系(in TACRED)

PTR: Prompt Tuning with Rules for Text Classification

Solve task



PTR: Prompt Tuning with Rules for Text Classification

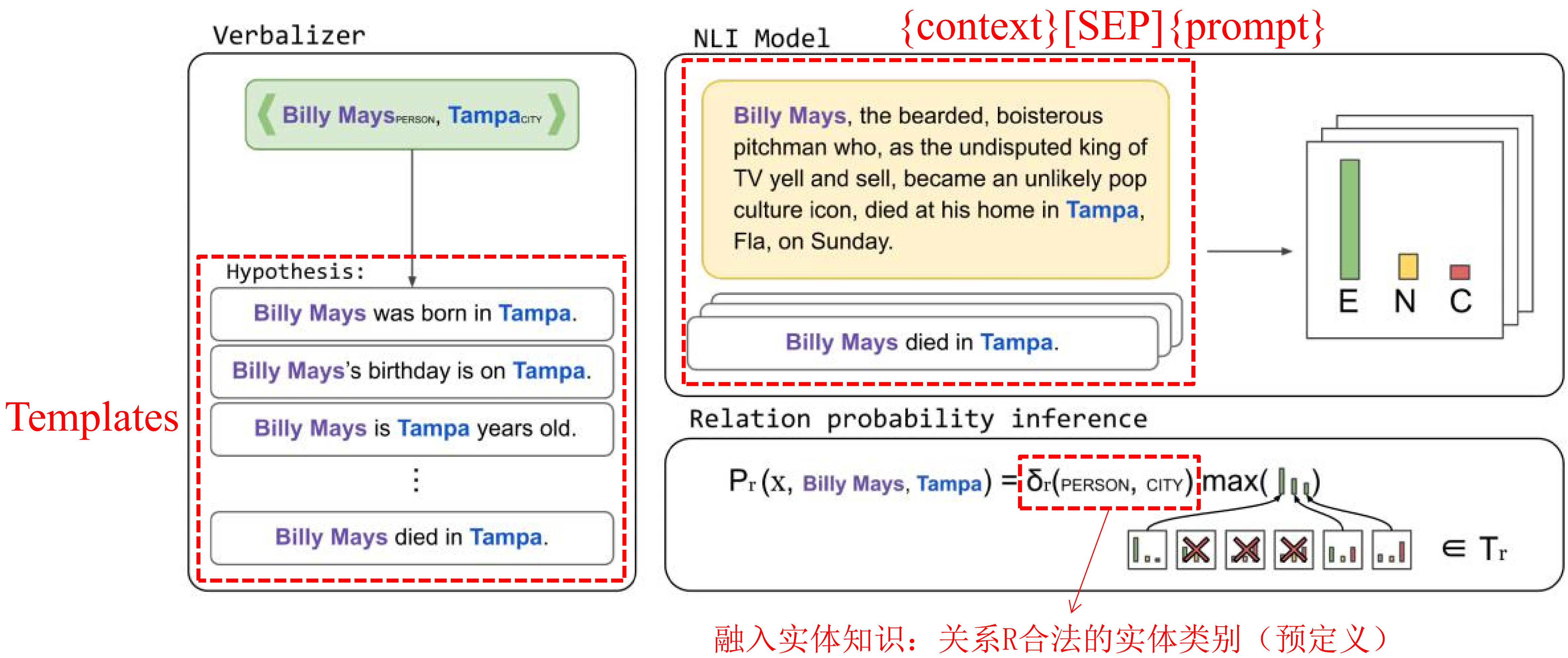
Result

Model	Extra Data	TACRED	TACREV	Re-TACRED	SEMEVAL
Learning models from scratch					
PA-LSTM (Zhang et al., 2017)	w/o	65.1	73.3	79.4	84.8
C-GCN (Zhang et al., 2018)	w/o	66.3	74.6	80.3	-
Fine-tuning pre-trained models					
RoBERTa_LARGE (Liu et al., 2019)	w/o	70.5	80.6	89.3	88.0
SpanBERT (Joshi et al., 2020)	w/	70.8	78.0	85.3	-
KnowBERT (Peters et al., 2019)	w/	71.5	79.3	-	89.1
LUKE (Yamada et al., 2020)	w/	72.7	80.6	90.3	-
MTB (Baldini Soares et al., 2019)	w/	70.1	-	-	89.5
Prompt tuning pre-trained models					
PTR	w/o	72.4	81.4	90.9	**89.9**
PTR (Reversed)	w/o	**75.9**	**83.9**	**91.9**	-

F1 分数

PTR: Prompt Tuning with Rules for Text Classification

Method



Label Verbalization and Entailment for Effective Zero- and Few-Shot Relation Extraction

$$\delta_r(e_1, e_2) = \begin{cases} 1 & e_1 \in E_{r1} \wedge e_2 \in E_{r2} \\ 0 & \text{otherwise} \end{cases}$$

Result

zero-shot setting

NLI Model	# Param.	MNLI Acc.	No Dev ($\mathcal{T}=0.5$) Pr.	Rec.	F1	1% Dev Pr.	Rec.	F1
ALBERT$_{\text{xxLarge}}$	223M	90.8	32.6	**79.5**	46.2	55.2	58.1	56.6 ±1.4
RoBERTa	355M	90.2	32.8	75.5	45.7	58.5	53.1	55.6 ±1.3
BART	406M	89.9	39.0	63.1	48.2	60.7	46.0	52.3 ±1.8
DeBERTa$_{\text{xLarge}}$	900M	91.7	40.3	77.7	53.0	**66.3**	59.7	**62.8** ±1.7
DeBERTa$_{\text{xxLarge}}$	1.5B	91.7	**46.6**	76.1	**57.8**	63.2	**59.8**	61.4 ±1.0

few-shot setting

Model	1% Pr.	Rec.	F1	5% Pr.	Rec.	F1	10% Prec.	Rec.	F1
SpanBERT	0.0	0.0	0.0 ±0.0	36.3	23.9	28.8 ±13.5	3.2	1.1	1.6 ±20.7
RoBERTa	56.8	4.1	7.7 ±3.6	52.8	34.6	41.8 ±3.3	61.0	50.3	55.1 ±0.8
K-Adapter	73.8	7.6	13.8 ±3.4	56.4	37.6	45.1 ±0.1	62.3	50.9	56.0 ±1.3
LUKE	61.5	9.9	17.0 ±5.9	57.1	47.0	51.6 ±0.4	60.6	60.6	60.6 ±0.4
NLI$_{\text{RoBERTa}}$ (ours)	56.6	55.6	56.1 ±0.0	60.4	68.3	64.1 ±0.2	**65.8**	69.9	67.8 ±0.2
NLI$_{\text{DeBERTa}}$ (ours)	**59.5**	**68.5**	**63.7** ±0.0	**64.1**	**74.8**	**69.0** ±0.2	62.4	**74.4**	**67.9** ±0.5

Label Verbalization and Entailment for Effective Zero- and Few-Shot Relation Extraction

Template&Answer-mapping

$$X_{\text{prompt}} = \texttt{[CLS]}\, X_{\text{in}}\, \texttt{[SEP]}\, \mathcal{T}\, \texttt{[SEP]}$$

$\boldsymbol{X_{in}}$： context

$\boldsymbol{T}$ ： template with entity mark

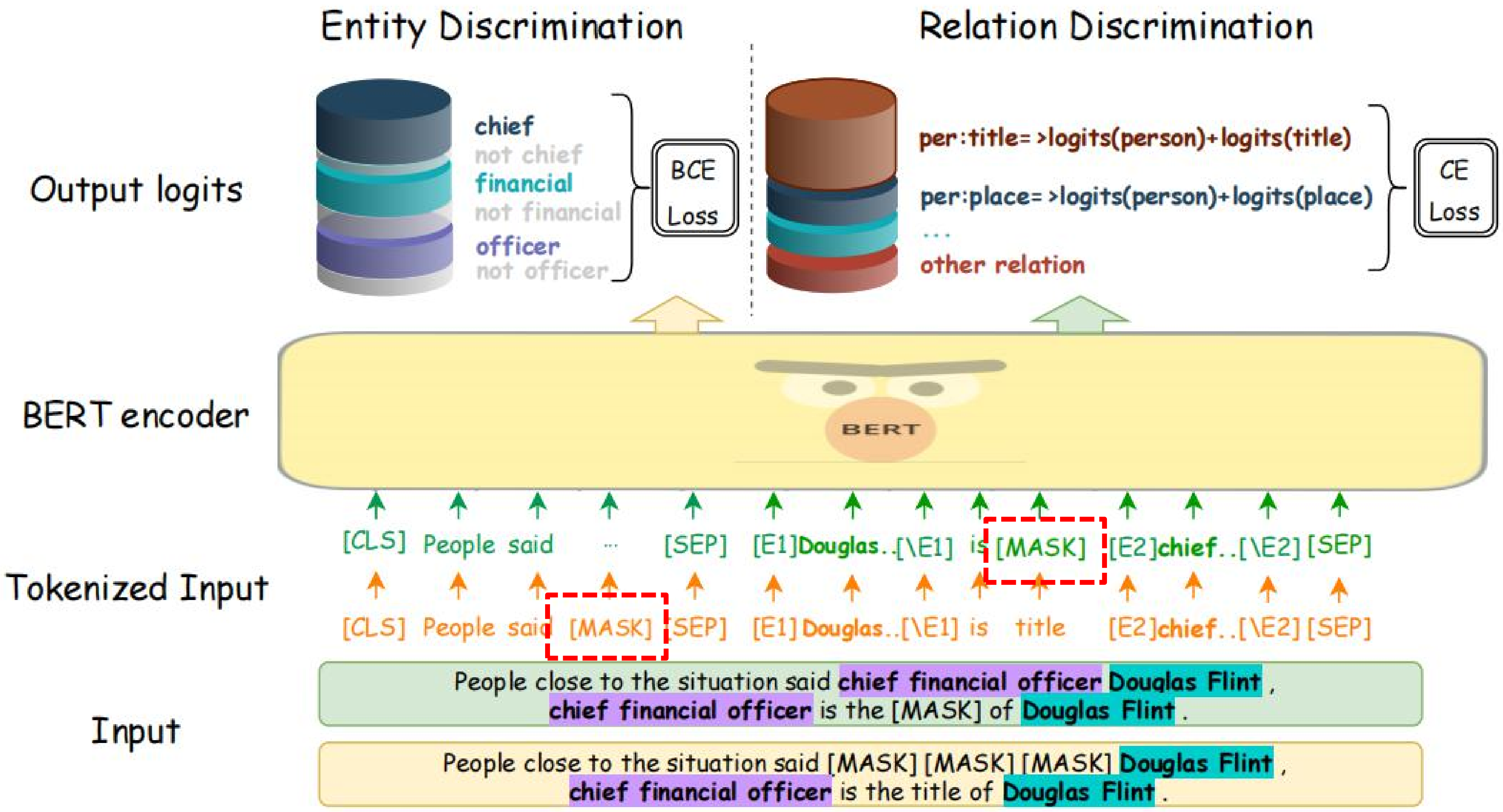
如：The relation between [E1] Xs [/E1] and [E2] Xo [/E2] is [MASK].

---

y = “per:city_of_death” $\longrightarrow$ $\mathcal{M}(y) = \{person, city, death\}$

AdaPrompt: Adaptive Prompt-based Finetuning for Relation Extraction

Solve task



AdaPrompt: Adaptive Prompt-based Finetuning for Relation Extraction

Solve task

$$p(y|X_{\text{prompt}}) = \frac{\exp\left(\mathbf{w}_{\mathcal{M}(y)} \cdot \mathbf{h}_{\texttt{[MASK]}}\right)}{\sum_{y' \in \mathcal{Y}} \exp\left(\mathbf{w}_{\mathcal{M}(y')} \cdot \mathbf{h}_{\texttt{[MASK]}}\right)}$$

$$\mathcal{L} = \mathcal{L}_R + \mathcal{L}_E.$$

$$\mathcal{L}_R = \mathrm{CE}(p(y|X_{\text{prompt}})).$$ 关系检测目标

$$q(x^m|x', y) = \frac{\exp([\![L_(x', y)]\!]_{x^m})}{\sum_{v' \in \mathcal{V}} \exp([\![L(x', y)]\!]_{v'})}$$ $$L_E = \sum_{m \in M} BCE(q(x^m|x^{'}, y))$$ 实体类别检测目标

AdaPrompt: Adaptive Prompt-based Finetuning for Relation Extraction

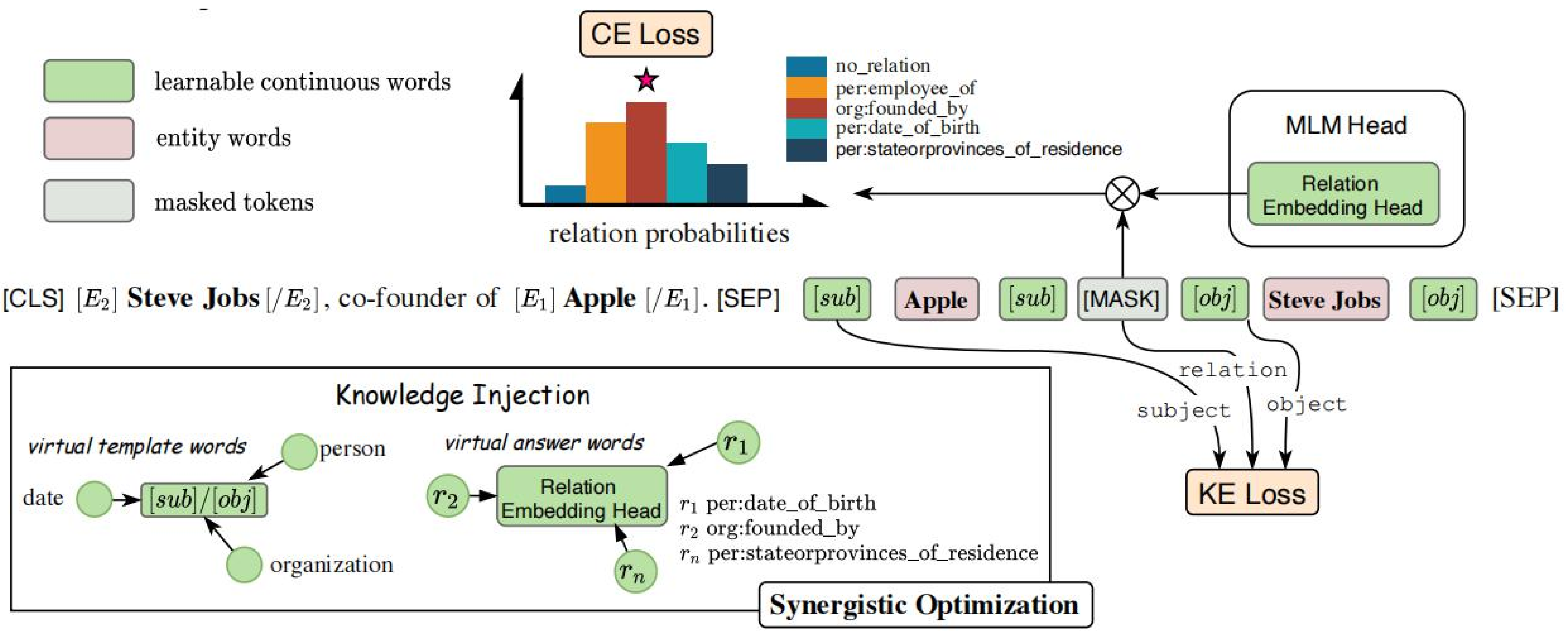
Result

Dataset	Model	$K = 8$		$K = 16$		$K = 32$		$Full$	
		Dev	**Test**	**Dev**	**Test**	**Dev**	**Test**	**Dev**	**Test**
TACRED-Revisit	SpanBERT (Joshi et al., 2020)	9.4	7.2	18.3	16.2	29.8	25.8	-	78.0
	GDPNet (Xue et al., 2020)	9.1	7.3	19.3	17.8	30.2	26.1	-	80.2
	AdaPrompt-tuning	**26.6**	**25.2**	**29.5**	**27.3**	**32.9**	**30.8**	**81.3**	**80.8**
DialogRE	BERTs (Yu et al., 2020)	26.6	26.1	41.1	40.8	47.7	47.7	63.0	57.3
	GDPNet (Xue et al., 2020)	23.4	23.6	41.5	48.5	47.9	47.1	67.1	61.5
	AdaPrompt-tuning	**36.7**	**36.6**	**44.8**	**43.3**	**49.3**	**49.1**	**67.0**	**65.8**

F1分数

AdaPrompt: Adaptive Prompt-based Finetuning for Relation Extraction

Method



KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction

Template&Answer-mapping

实体类别信息

$$\hat{\mathbf{e}}_{[sub]} = \phi_{sub} \cdot \mathbf{e}\left(\mathbf{C}_{sub}\right),$$
$$\hat{\mathbf{e}}_{[obj]} = \phi_{obj} \cdot \mathbf{e}\left(\mathbf{C}_{obj}\right),$$

Φ_sub：每个relation的数据里出现的sub的类别（C_sub）的分布

Φ_obj：每个relation的数据里出现的obj的类别(C_obj)的分布

关系语义信息

$$\hat{\mathbf{e}}(v_1') = \phi_{rel} \cdot \mathbf{e}\left(C_{rel_1}\right),$$

e.g.
rel_1 = “per:countries_of_residence”
Crel_1 = {“person”, “contries”, “residence”}
Φ_rel：每个基本词在所有relation里的分布

KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction

Solve task

待优化参数： $\{\hat{\mathbf{e}}_{[sub]}, \hat{\mathbf{e}}_{[obj]}, \hat{\mathbf{e}}_{[rel]}(\mathcal{V}')\}$

$$J_{[MASK]} = -\frac{1}{|X|}\sum_{x \in X} \boldsymbol{y} \, log \, softmax(p(y|x))$$

$$p(y|x) = p([MASK] = rel\_embedding|x_{prompt})$$

$$\mathcal{J}_{\mathrm{KE}} = -\log sigmoid(\gamma - d_r(\mathbf{h}, \mathbf{t})) - \sum_{i=1}^{n} \frac{1}{n} \log sigmoid(d_r(\mathbf{h}_\mathbf{i}', \mathbf{t}_\mathbf{i}') - \gamma),$$

$$d_r(\mathbf{h}, \mathbf{t}) = \|\mathbf{h} + \mathbf{r} - \mathbf{t}\|_p,$$

$$\mathcal{J} = \mathcal{J}_{[\mathrm{MASK}]} + \lambda \mathcal{J}_{KE},$$

KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction

Solve task

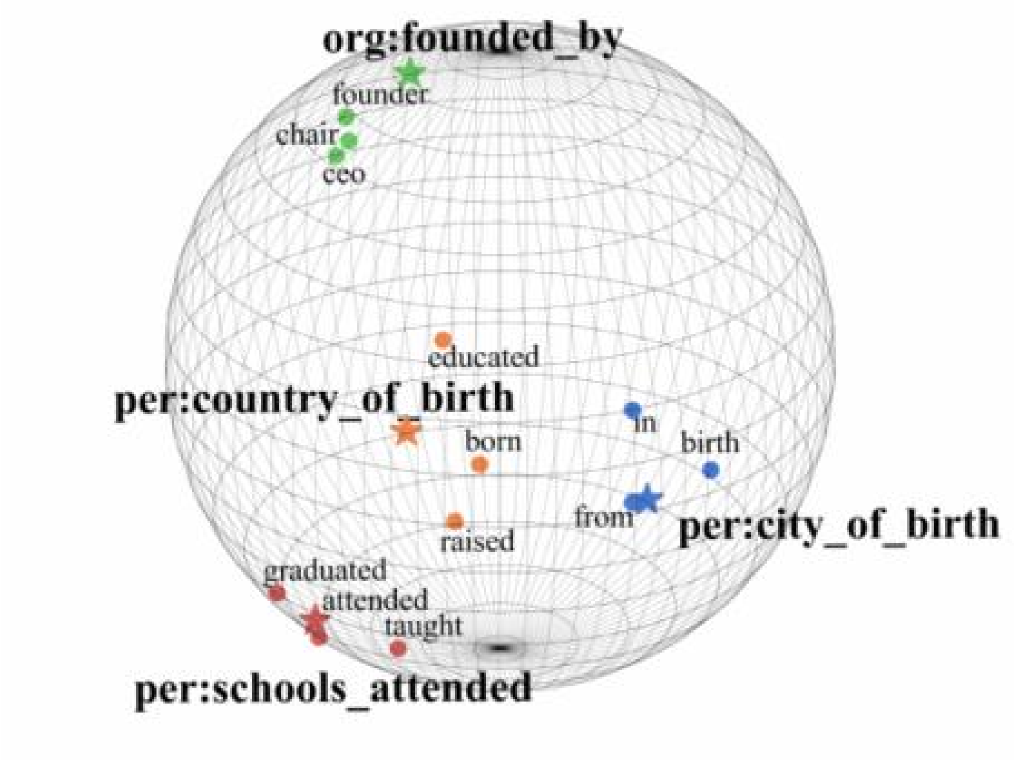
待优化参数： $\{\hat{\mathbf{e}}_{[sub]}, \hat{\mathbf{e}}_{[obj]}, \hat{\mathbf{e}}_{[rel]}(\mathcal{V}')\}$

Input Example of our KnowPrompt	Top 3 words around [sub]	Top 3 words around [obj]
x: [CLS] It sold [E_1] **ALICO** [/E_1] to [E_2] **MetLife Inc** [E_2] for $ 162 billion. [SEP] [sub] **ALICO** [sub] [MASK] [obj] **MetLife Inc** [obj]. [SEP] **y**: "*org : member_of*"	organization group corporation	company plc organization
x: [CLS] [E_1] **Ismael Rukwago** [/E_1], a senior [E_2] **ADF** [E_2] commander, denied any involvement. [SEP] [sub] **Ismael Rukwago** [sub] [MASK] [obj] **ADF** [obj]. [SEP] **y**: "*per : employee_of*"	person commander colonel	intelligence organization command

KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction

Solve task

待优化参数： $\{\hat{\mathbf{e}}_{[sub]}, \hat{\mathbf{e}}_{[obj]}, \hat{\mathbf{e}}_{[rel]}(\mathcal{V}')\}$



KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction

low-Resource Result

Low-Resource Setting							
Split	Methods	SEMEVAL	DialogRE	TACRED-Revisit	Re-TACRED	WiKi80	Average
K=8	FINE-TUNING	28.8	26.1	10.5	20.1	47.6	26.6
	GDPNET	27.3	23.6	8.3	18.8	45.7	24.7
	PTR	61.9	35.5	25.3	43.6	67.6	46.8
	KNOWPROMPT	**64.5** (+35.7)	**40.8** (+14.7)	**28.6** (+18.1)	**45.8** (+25.7)	**71.8** (+24.2)	**50.3** (+23.7)
K=16	FINE-TUNING	45.7	40.8	19.2	47.4	59.4	42.5
	GDPNET	45.5	38.5	20.8	48.0	61.2	42.8
	PTR	71.8	43.5	27.2	51.8	75.6	53.8
	KNOWPROMPT	**73.8** (+28.1)	**47.7** (+6.9)	**30.8** (+11.6)	**53.8** (+6.4)	**78.8** (+19.4)	**56.9** (+14.4)
K=32	FINE-TUNING	65.4	47.7	26.0	53.6	69.9	52.5
	GDPNET	67.2	47.1	28.1	54.8	72.3	53.9
	PTR	78.3	49.5	33.1	54.8	78.8	59.3
	KNOWPROMPT	**79.8** (+14.4)	**53.2** (+5.5)	**34.2** (+8.2)	**55.2** (+1.6)	**81.3** (+11.4)	**60.7** (+8.2)

F1分数

KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction