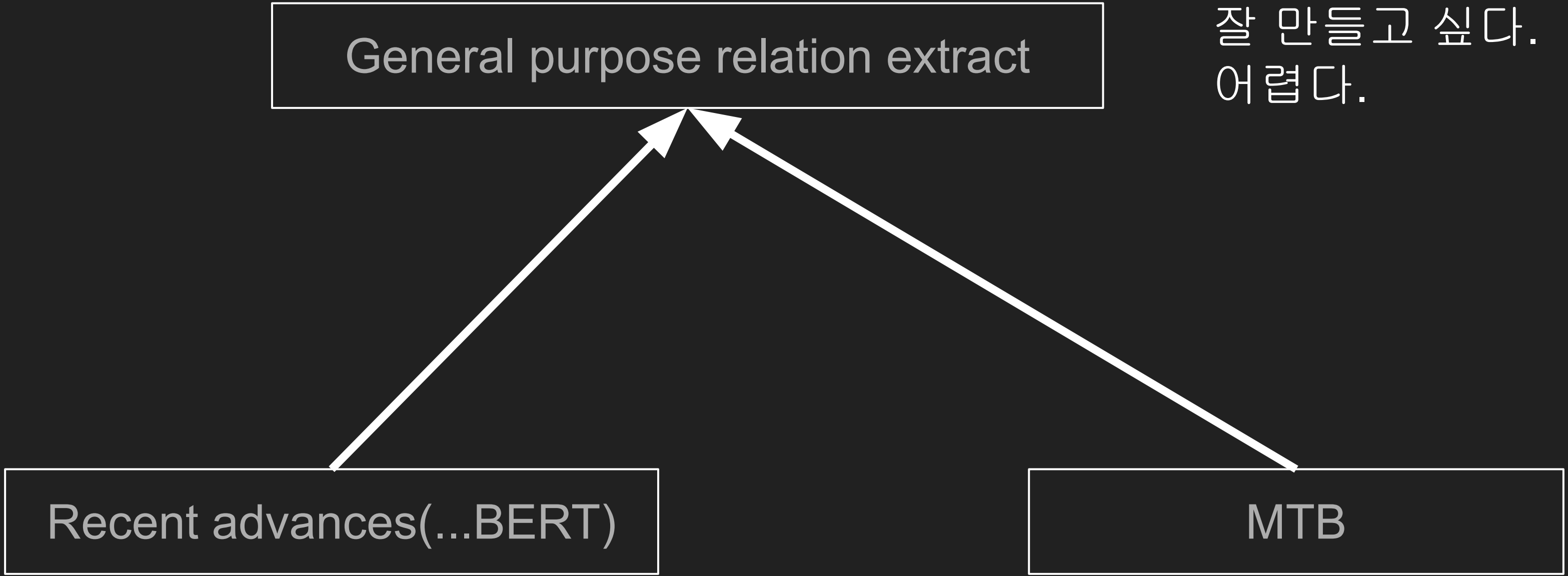
Matching the Blanks

Matching the Blanks: Distributional Similarity for Relation Learning

ABSTRACT



잘 만들고 싶다.
어렵다.

Overviews

Sequence of tokens

$\mathbf{x} = [x_0 \ldots x_n]$

Statement

$\mathbf{s}_1 = (i, j)$ and $\mathbf{s}_2 = (k, l)$

Relation statement

A relation statement is a triple $\mathbf{r} = (\mathbf{x}, \mathbf{s}_1, \mathbf{s}_2)$

Goal

Our goal is to learn a function $\mathbf{h}_r = f_\theta(\mathbf{r})$

Overview

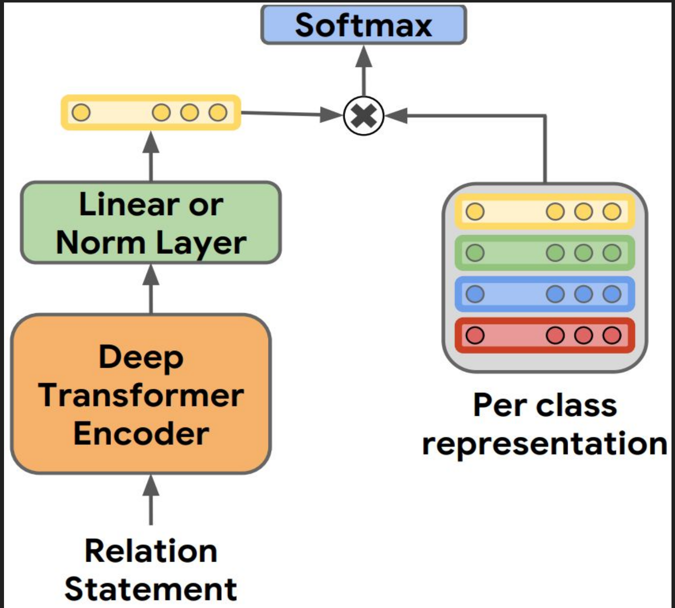
Contributions

1. Investigate different architectures on Transformer.
2. Show that model can be learned from widely available distant supervision in the form of entity linked text.

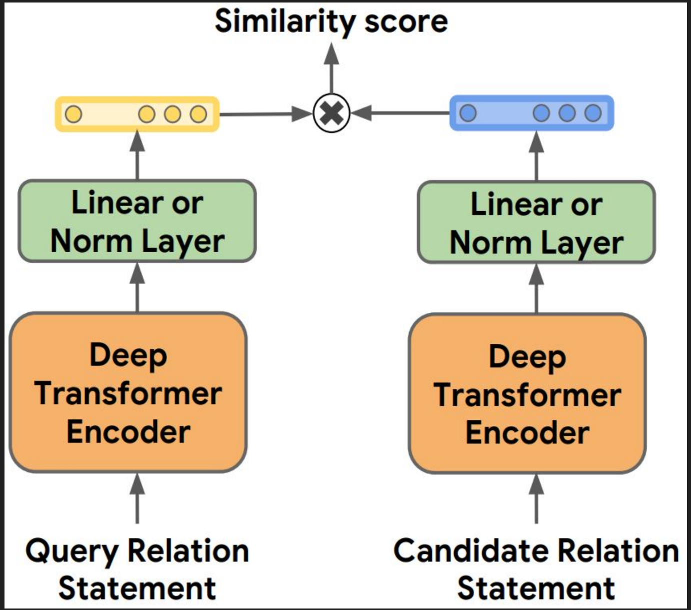
Architectures for Relation Learning

relation extractions tasks:

1. Fully supervised relation extraction

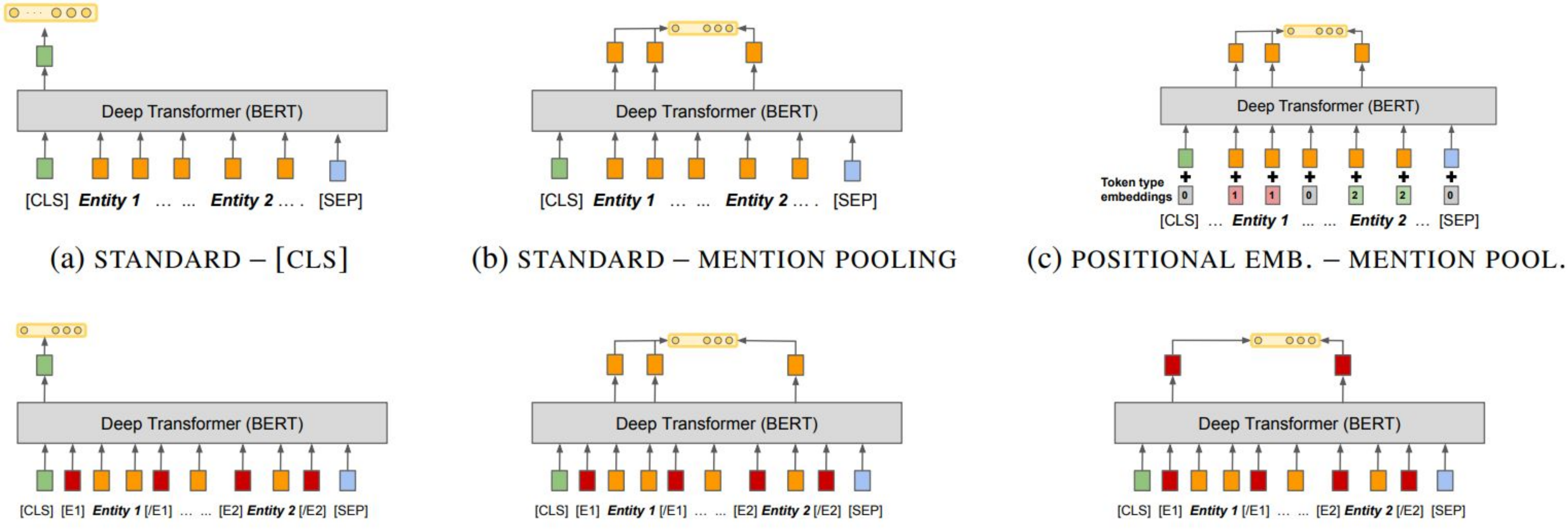


2. Few-shot relation matching



Architectures for Relation Learning

Different architectures on Transformer



(a) STANDARD – [CLS]
(b) STANDARD – MENTION POOLING
(c) POSITIONAL EMB. – MENTION POOL.
(d) ENTITY MARKERS – [CLS]
(e) ENTITY MARKERS – MENTION POOL.
(f) ENTITY MARKERS – ENTITY START

Architectures for Relation Learning

		SemEval 2010 Task 8		KBP37		TACRED		FewRel 5-way-1-shot
# training annotated examples		8,000 (6,500 for dev)		15,916		68,120		44,800
# relation types		19		37		42		100
		Dev F1	Test F1	Dev F1	Test F1	Dev F1	Test F1	Dev Acc.
Wang et al. (2016)*		–	88.0	–	–	–	–	–
Zhang and Wang (2015)*		–	79.6	–	58.8	–	–	–
Bilan and Roth (2018)*		–	84.8	–	–	–	68.2	–
Han et al. (2018)		–	–	–	–	–	–	71.6
Input type	**Output type**							
STANDARD	[CLS]	71.6	–	41.3	–	23.4	–	85.2
STANDARD	MENTION POOL.	78.8	–	48.3	–	66.7	–	87.5
POSITIONAL EMB.	MENTION POOL.	79.1	–	32.5	–	63.9	–	87.5
ENTITY MARKERS	[CLS]	81.2	–	68.7	–	65.7	–	85.2
ENTITY MARKERS	MENTION POOL.	80.4	–	68.2	–	69.5	–	87.6
ENTITY MARKERS	ENTITY START	**82.1**	**89.2**	**70**	**68.3**	**70.1**	**70.1**	**88.9**

Table 1: Results for supervised relation extraction tasks. Results on rows where the model name is marked with a * symbol are reported as published, all other numbers have been computed by us. SemEval 2010 Task 8 does not establish a default split for development; for this work we use a random slice of the training set with 1,500 examples.

Learning by Matching the Blanks

MTB: learning f_{θ} from entity linked text

$\mathbf{r}_A$	In 1976, e_1 (then of Bell Labs) published e_2, the first of his books on programming inspired by the Unix operating system.
$\mathbf{r}_B$	The "e_2" series spread the essence of "C/Unix thinking" with makeovers for Fortran and Pascal. e_1's Ratfor was eventually put in the public domain.
$\mathbf{r}_C$	e_1 worked at Bell Labs alongside e_3 creators Ken Thompson and Dennis Ritchie.
Mentions	e_1 = Brian Kernighan, e_2 = Software Tools, e_3 = Unix

Table 2: Example of "matching the blanks" automatically generated training data. Statement pairs r_A and r_B form a positive example since they share resolution of two entities. Statement pairs r_A and r_C as well as r_B and r_C form strong negative pairs since they share one entity in common but contain other non-matching entities.

Learning by Matching the Blanks

BERT pre-training use 2 loss: Loss_{MLM}, Loss_{MTB}

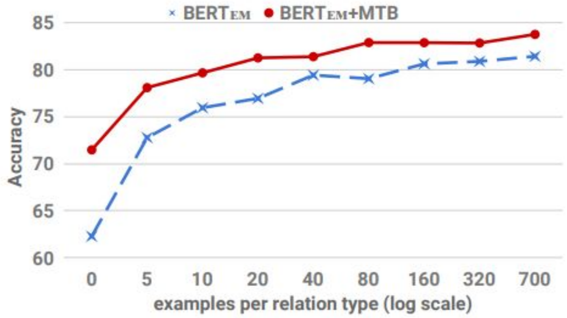
minimizes the loss

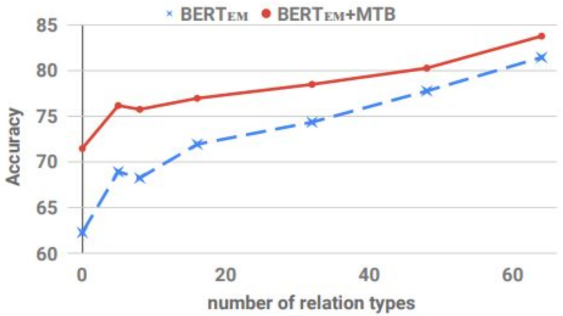
$$p(l = 1|\mathbf{r}, \mathbf{r}') = \frac{1}{1 + \exp f_\theta(\mathbf{r})^\top f_\theta(\mathbf{r}')}$$

$$\mathcal{L}(\mathcal{D}) = -\frac{1}{|\mathcal{D}|^2} \sum_{(\mathbf{r}, e_1, e_2) \in \mathcal{D}} \sum_{(\mathbf{r}', e_1', e_2') \in \mathcal{D}} \delta_{e_1, e_1'} \delta_{e_2, e_2'} \cdot \log p(l = 1|\mathbf{r}, \mathbf{r}') + (1 - \delta_{e_1, e_1'} \delta_{e_2, e_2'}) \cdot \log(1 - p(l = 1|\mathbf{r}, \mathbf{r}')) \quad (1)$$

where $\delta_{e,e'}$ is the Kronecker delta that takes the value 1 iff $e = e'$, and 0 otherwise.

Experiments





5 way 1 shot						
# examples per type	0	5	20	80	320	700
Prot.Net. (CNN)	–	–	–	–	–	71.6
BERT_{EM}	72.9	81.6	85.1	86.9	88.8	88.9
BERT_{EM}+MTB	80.4	85.5	88.4	89.6	89.6	90.1

10 way 1 shot						
# examples per type	0	5	20	80	320	700
Prot.Net. (CNN)	–	–	–	–	–	58.8
BERT_{EM}	62.3	72.8	76.9	79.0	81.4	82.8
BERT_{EM}+MTB	71.5	78.1	81.2	82.9	83.7	83.4

5 way 1 shot					
# training types	0	5	16	32	64
Prot.Net. (CNN)	–	–	–	–	71.6
BERT_{EM}	72.9	78.4	81.2	83.4	88.9
BERT_{EM}+MTB	80.4	84.04	85.5	86.8	90.1

10 way 1 shot					
# training types	0	5	16	32	64
Prot.Net. (CNN)	–	–	–	–	58.8
BERT_{EM}	62.3	68.9	71.9	74.3	81.4
BERT_{EM}+MTB	71.5	76.2	76.9	78.5	83.7

Figure 4: Comparison of classifiers tuned on FewRel. Results are for the development set while varying the amount of annotated examples available for fine-tuning. On the left, we display accuracies while varying the number of examples per relation type, while maintaining all 64 relations available for training. On the right, we display accuracy on the development set of the two models while varying the total number of relation types available for tuning, while maintaining all 700 examples per relation type. In both graphs, results for the 10-way-1-shot variant of the task are displayed.

Experiments

% of training set	1%	10%	20%	50%	100%
SemEval 2010 Task 8					
BERT_{EM}	28.6	66.9	75.5	80.3	82.1
BERT_{EM}+MTB	31.2	70.8	76.2	80.4	82.7
KBP-37					
BERT_{EM}	40.1	63.6	65.4	67.8	69.5
BERT_{EM}+MTB	44.2	66.3	67.2	68.8	70.3
TACRED					
BERT_{EM}	32.8	59.6	65.6	69.0	70.1
BERT_{EM}+MTB	43.4	64.8	67.2	69.9	70.6

Table 5: F1 scores on development sets for supervised relation extraction tasks while varying the amount of tuning data available to our BERT_{EM} and BERT_{EM}+MTB models.

Conclusion

Producing useful relation representations directly from text.

We describe a novel training setup, which we call matching the blanks.

1. Our models achieves state of the art results on three relation extraction tasks, and outperforms human accuracy on few-shot relation matching.
2. We argue that it could significantly reduce the amount of human effort required to create relation extractors.