A Walk-based Model on Entity Graphs for Relation Extraction

Fenia Christopoulou[1,3], Makoto Miwa[2,3], Sophia Ananiadou[1,3]

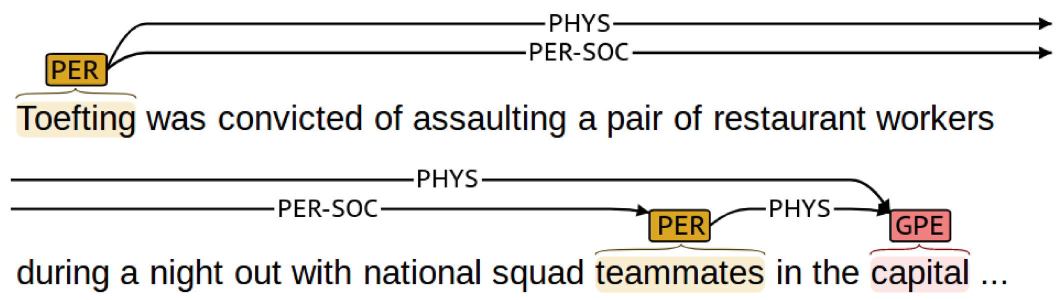
[1]School of Computer Science, The University of Manchester, UK
[2]Toyota Technological Institute, Nagoya, Japan
[3]Artificial Intelligence Research Center (AIRC)
National Institute of Advance Industrial Science and Technology (AIST)

Introduction

Task: Sentence-level, binary relation extraction between given named entities (NE)

Assumption: The relation between a pair can be supported by other pairs in the same sentence

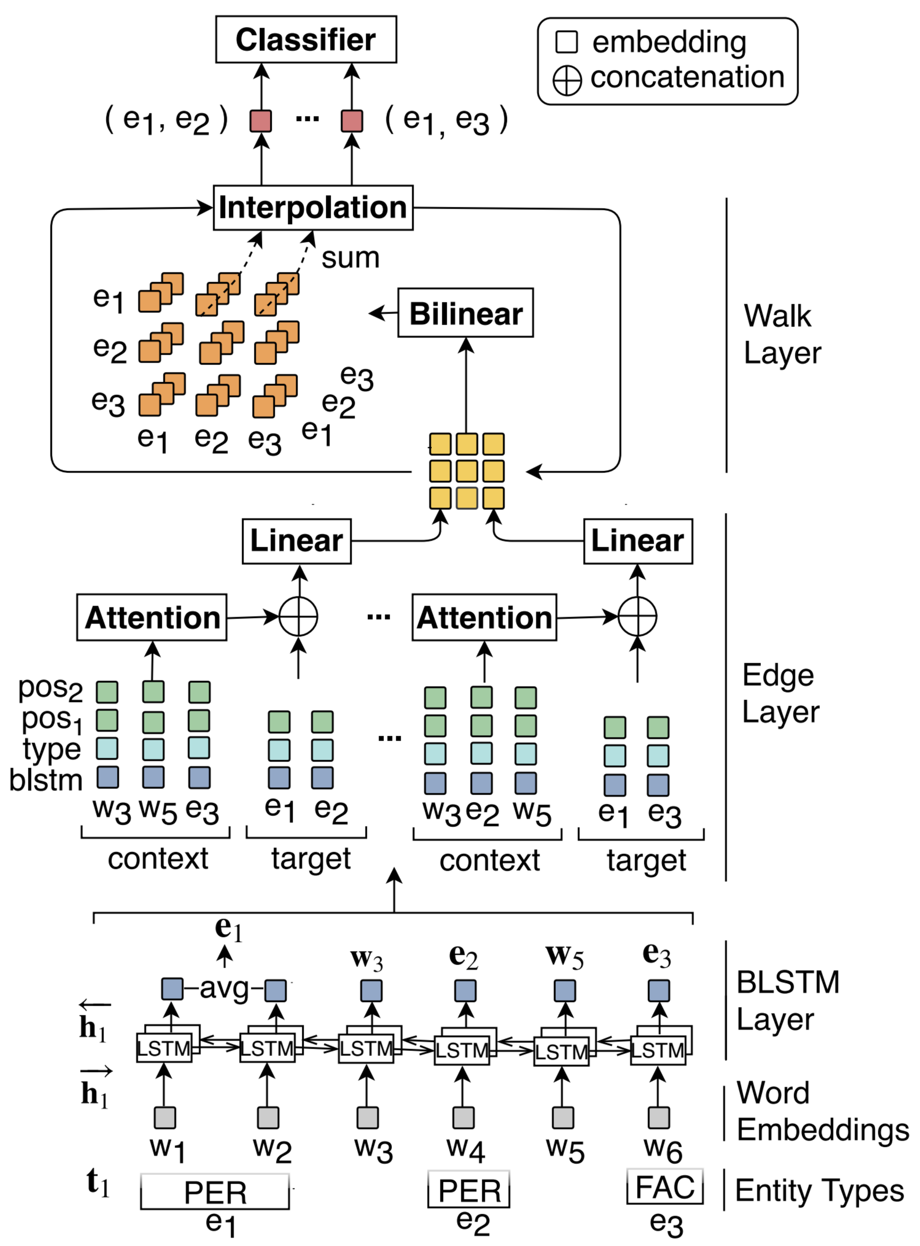
Motivating Example: *Toefting* and *capital* are related through preposition *in* (direct association) and through entity *teammates* (indirect association)



Contributions

- Simultaneously consider all pairs in a sentence
- Independence of external syntactic tools
- Single representation for up-to l-length walk relations between two named entities

Proposed Approach

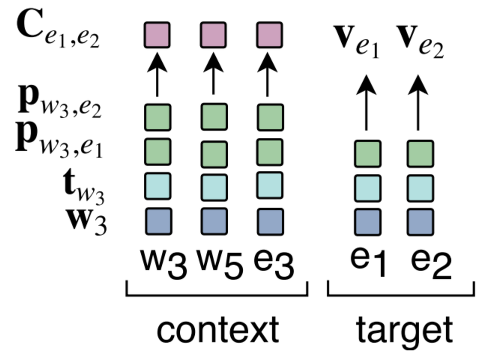


Model Architecture

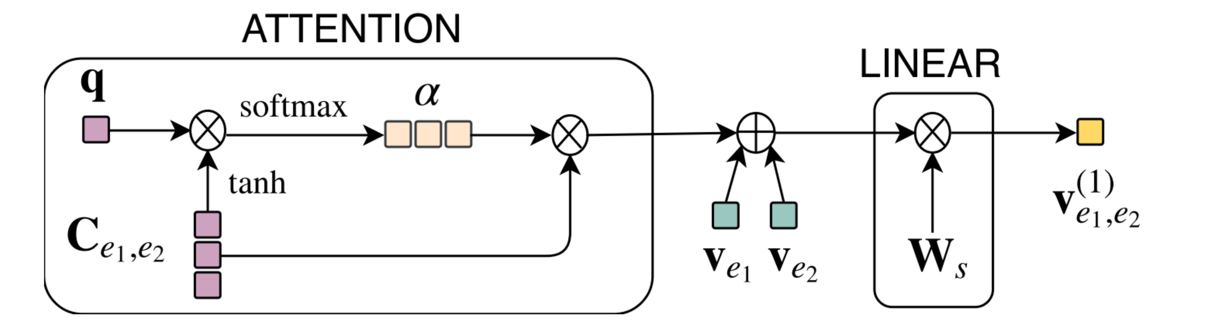
Edge Layer

1. Pair & Context Representations
   - **Target** pair: named entity pair of interest
   - Target pair **context**: sentence words excluding pair
   - Relative positions: entity e_i | word w_z to entity e_j: $\mathbf{p}_{e_i,e_j}$, $\mathbf{p}_{w_z,e_j}$
   - Target entities e_i, e_j representations: $\mathbf{v}_{e_i} = [\mathbf{e}_i; \mathbf{t}_{e_i}; \mathbf{p}_{e_i,e_j}]$
     $\mathbf{v}_{e_j} = [\mathbf{e}_j; \mathbf{t}_{e_j}; \mathbf{p}_{e_j,e_i}]$
   - Pair (e_i, e_j) context representation:

$$\mathbf{C}_{e_i,e_j} = \begin{bmatrix} \mathbf{w}_z; \mathbf{t}_{w_z}; \mathbf{p}_{w_z,e_i}; \mathbf{p}_{w_z,e_j} \\ \cdots \\ \mathbf{e}_n; \mathbf{t}_{e_n}; \mathbf{p}_{e_n,e_i}; \mathbf{p}_{e_n,e_j} \end{bmatrix}$$



1. Edge Representation



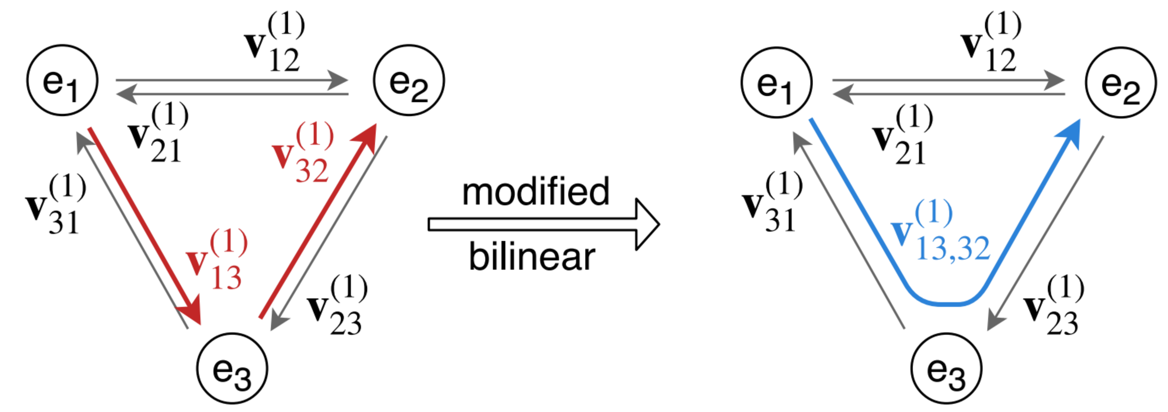
$\mathbf{q}$ attention vector, α context weights, $\otimes$ matrix multiplication

Walk Layer

1. Walk Generation
   - Combine two consecutive edges into a walk representation

$$\mathbf{v}_{ik,kj}^{(\lambda)} = f(\mathbf{v}_{ik}^{(\lambda)}, \mathbf{v}_{kj}^{(\lambda)}) = \sigma\left(\mathbf{v}_{ik}^{(\lambda)} \odot (\mathbf{W}_b\, \mathbf{v}_{kj}^{(\lambda)})\right),$$

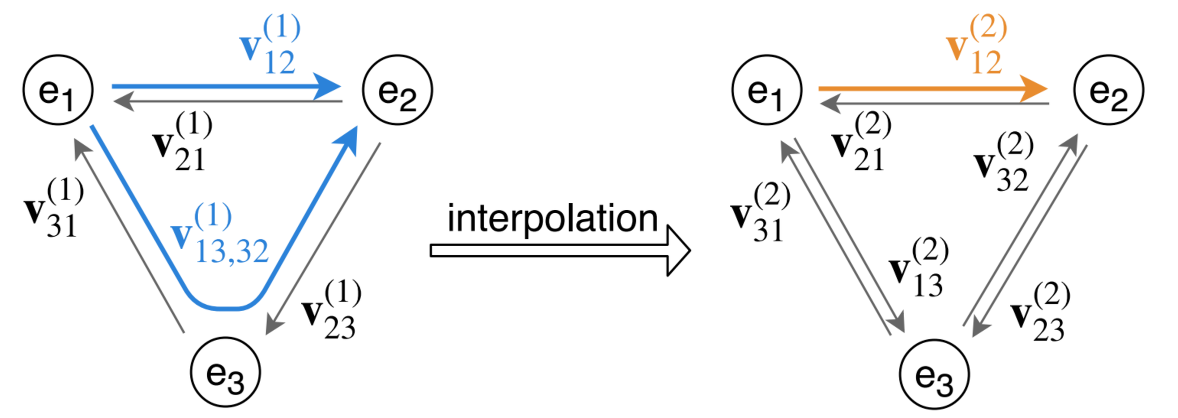
$\odot$ element-wise multiplication, $\mathbf{W}_b \in \mathbb{R}^{n_b \times n_b}$
σ sigmoid non-linear function, λ current walks length



2. Walk Aggregation
   - Linear combination of direct and indirect edge representations
   - Direct edges representations are *updated*
   - Self edges are ignored

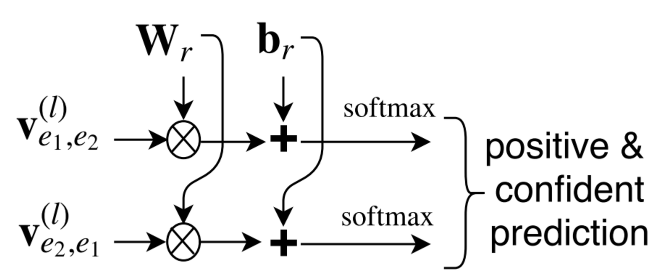
$$\mathbf{v}_{ij}^{(2\lambda)} = \beta\, \mathbf{v}_{ij}^{(\lambda)} + (1-\beta) \sum_{k \neq i,j} f(\mathbf{v}_{ik}^{(\lambda)}, \mathbf{v}_{kj}^{(\lambda)}),$$

β indicates the importance of the shorter walks

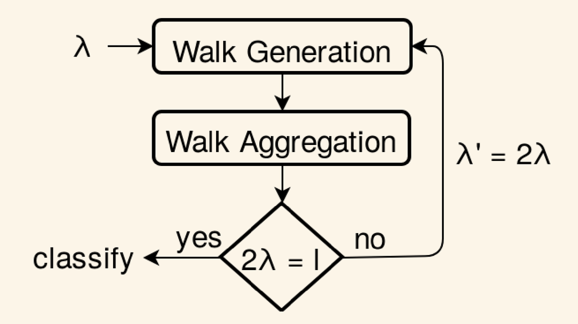


Classification Layer

Directional relations



ITERATION



N iterations $\rightarrow 2^N$ length walks

Settings

Dataset	Training
ACE 2005 [1]	Adam Optimizer
7 entity types	L2 Regularization
6 relations types	Early Stopping
	Dropout

SoA Models:	SPTree [2]	CNN [3]
One pair/sentence	✓	✓
Dependency parser	✓	✗
Data split	train/test	5-fold

Results

Model	P	R	F1 (%)
SPTree	70.1	61.2	65.3
Baseline	72.5	53.3	61.4*
No walks $l=1$	71.9	55.6	62.7
Walks $l=2$	69.9	58.4	63.6⋄
Walks $l=4$	69.7	59.5	64.2⋄
Walks $l=8$	71.5	55.3	62.4

Table 1: Performance on ACE 2005 test set.
* significance at $p < 0.05$ vs. SPTree,
⋄ significance at $p < 0.05$ vs. Baseline.

Model	P	R	F1 (%)
CNN	71.5	53.9	61.3
Walks $l=4$	65.8	58.4	61.9

Table 2: Performance on ACE 2005 test set.

- Walks model ($l = 4$) approximates the state-of-the-art
- Longer walks improve recall
- Too long walks degrade performance

Analysis

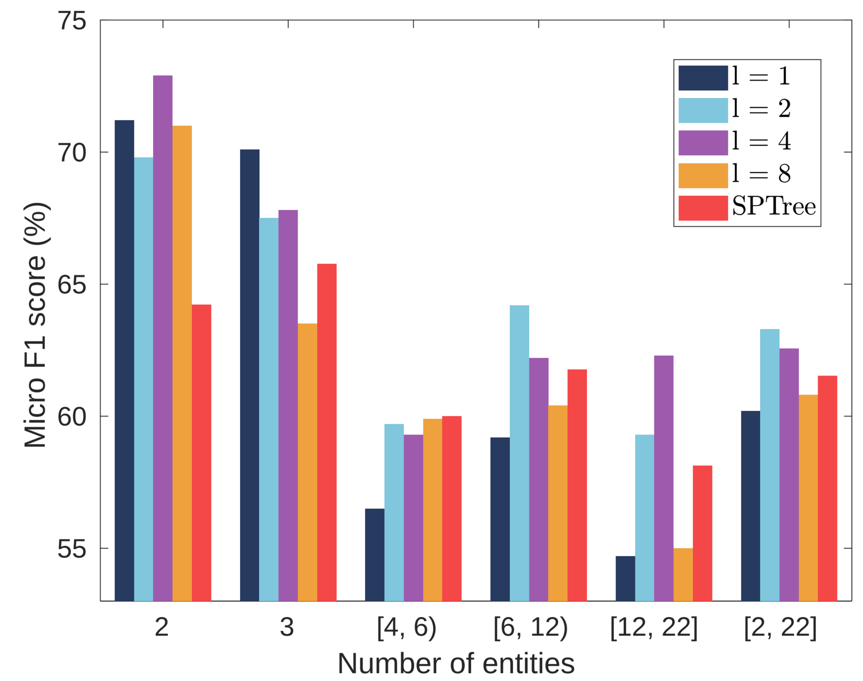


Figure 1: Performance on ACE 2005 development set for sentences that include different number of entities.

- Walks surpass SPTree performance on multi-pair sentences
- Shorter walks can work better for sentences with many entities
- Very long walks ($l = 8$) produce non-meaningful representations

Conclusions

- Relations between NE pairs are encoded with walks
- Longer walks improve performance, but
  - More entities do not necessarily need longer walks
- Walks can improve the detection of related pairs
- Too long relation walks (≥ 6-length) are hard to interpret, even by humans

References

[1] G. R. Doddington, A. Mitchell, M. A. Przybocki, L. A. Ramshaw, S. Strassel, and R. M. Weischedel. The automatic content extraction (ace) program-tasks, data, and evaluation. In *Proc. of LREC*, 2004.

[2] M. Miwa and M. Bansal. End-to-end relation extraction using lstms on sequences and tree structures. In *Proc. of ACL*, 2016.

[3] T. H. Nguyen and R. Grishman. Relation extraction: Perspective from convolutional neural networks. In *Proc. of VSM-NLP*, 2015.

Funding

Contact: efstathia.christopoulou@manchester.ac.uk