THE UNIVERSITY
of LIVERPOOL

Context-Guided Self-Supervised Relation Embeddings

Huda Hakami and Danushka Bollegala

Department of Computer Science, University of Liverpool, United Kingdom

Introduction

Semantic relations

- Semantic relations: the connections that present between words
  - Hyponym/Hypernym (*eagle*, *bird*)
  - Causality (*smoking*, *cancer*)
  - Meronym/Holonym (*tire*, *car*)
  - Antonym (*hot*, *cold*)
- Relation learning and representation can help:
  - Knowledge base completion
  - Question answering
  - Textual Entailment
  - Word-sense disambiguation

Relation Representation Approaches

Aim: representing the semantic relations of word-pairs in a vector space

Pattern-based approaches:

Rely on the contexts in which two words co-occur in a text corpus.

- *lion* is a large *cat*, *ostrich* is a large *bird* $\Rightarrow$ *X* is a large *Y*
- *water* flows in *pipe*, *electricity* flows in *wire* $\Rightarrow$ *X* flows in *Y*

- *Latent relation hypothesis*: word-pairs that co-occur in similar patterns tend to have similar semantic relations (Turney et al., 2003).
- Sparsity: fail to represent relations between words that never co-occur.

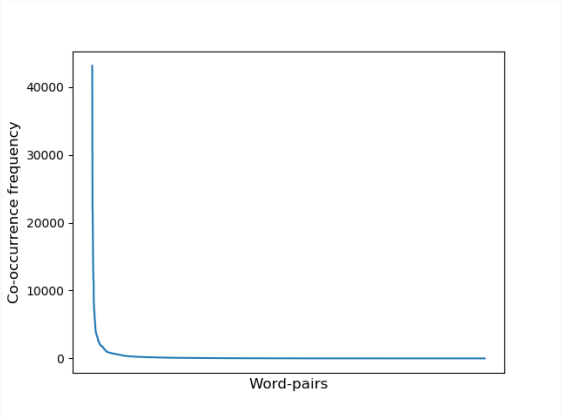


Figure 1: Co-occurrence frequency for word-pairs in SemEval-2012 task2 from Wikipedia corpus.

Relation Representation Approaches

Aim: representing the semantic relations of word-pairs in a vector space

Compositional-based approaches:

A relation between two words is represented by employing the corresponding word embeddings which consider global contexts of words in a corpus.

- Relations between words could be represented by the difference of the corresponding word embeddings (Mikolov et al., 2013).
  - ***king*** $-$ ***queen*** $\approx$ ***man*** $-$ ***woman***
  - ***London*** $-$ ***England*** $\approx$ ***Rome*** $-$ ***Italy***
- The relation representation is *composed* using the semantic representations of the two constituent words of the related pairs.
- Overcome the sparseness issues in the pattern-based methods.
- Lack of relational information : word embeddings are not trained to capture such linguistic regularities

Research Contribution

Hybrid approaches

Combine the strengths of pattern-based and compositional-based approaches

- Prior works on relation embeddings have pre-dominantly focused on either one type of those two resources exclusively
- We propose a self-supervised Context-Guided Relation Embedding (CGRE):
  - CGRE is a compositional operator for representing relations between words.
  - CGRE is learnt using word embeddings for related pairs and is **guided** by the relational patterns of the pairs extracted from a corpus.
  - After training, CGRE **do not require the two words to co-occur within a contextual window** to represent the relation between them.
  - CGRE **generalises** to word-pairs that do not co-occur or belong to unseen relations (not limited to the training data) .

Method

Context-Guided Relation Embeddings (CGRE)

Goal

Learn a parametrised operator that maps an unseen word-pair (c, d) to relation embeddings $\boldsymbol{r}_{(c,d)} \in \mathbb{R}^m$

High-level procedure for CGRE:

Given: Training word-pairs $\mathcal{D} = \{(a_i, b_i)\}_i^N$, text corpus $\mathcal{C}$, pre-trained word embeddings

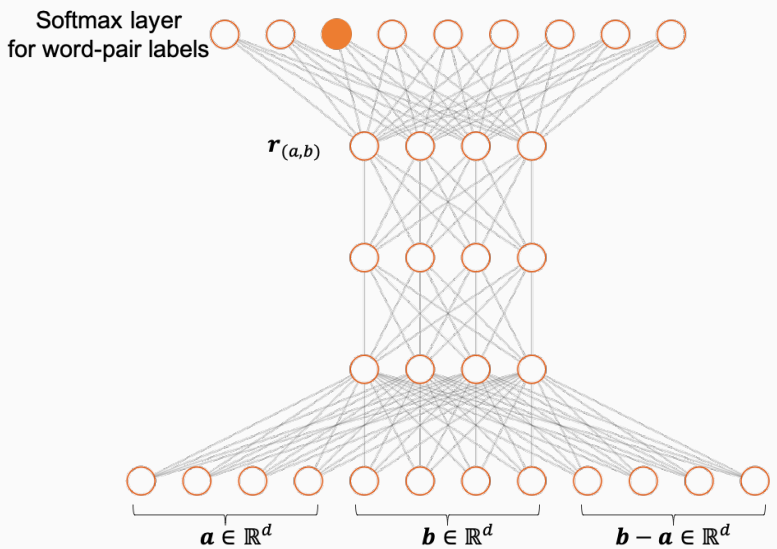
1. Generate pseudo labels for word-pairs using $\mathcal{D}$: $(a_i, b_i) \rightarrow r_i$
2. **For** each $(a_i, b_i) \in \mathcal{D}$: extract a set of patterns $\mathcal{P}(a, b)$ from $\mathcal{C}$
3. **For** each selected batch $\{(a, b, \mathcal{P}(a, b))\}$: learn word-pair mapping function $f(a, b, \theta_f)$ regularised by patterns encoder function $g(\mathcal{P}(a, b), \theta_g)$
4. **For** each pair in the testing set (c_i, d_i): generate the relation embedding $\boldsymbol{r}_i$ using $f(c_i, d_i, \theta_f)$

Word Pair Mapping $f(a, b, \theta_f)$

Compositional-based learning phase:

- $f(a, b, \theta_f)$: a word-pair (a, b) is fed to a deep multilayer neural network with a nonlinearity activation.
- The output of the last layer of the network represents $\boldsymbol{r}_{(a,b)}$.
- $\boldsymbol{r}_{(a,b)}$ is passed to a softmax layer to predict the label for (a, b).
- **Training objective**: ℓ_2 regularised cross-entropy loss

$$\mathcal{J}_C = -\sum_{(a,b,r)\in\mathcal{D}} \log p(r|f(a, b, \theta_f)) \quad (1)$$



Patterns Encoder $g(\mathcal{P}(a,b),\theta_g)$

Pattern-based learning phase

$$g(\mathcal{P}(a,b),\theta_g) = \sum_{p\in\mathcal{P}(a,b)} w(a,p,b)\boldsymbol{h}(a,p,b,\theta_h) \quad (2)$$

$$w(a,p,b) = \frac{c(a,p,b)}{\sum_{t\in\mathcal{P}(a,b)} c(a,t,b)} \quad (3)$$

- $\mathcal{P}(a,b)$: we need to encode a set of contextual co-occurrences between a and b into a fixed-length vector
- $\boldsymbol{h}(a,p,b,\theta_h)$: LSTM to map a sequences of words of a pattern p using the sequence of the corresponding pre-trained word embeddings
- $w(a,p,b)$: strength association between (a,b) and p using co-occurrence statistics.

CGRE Objective

- Because the pattern-based and compositional-based methods represent the same semantic relation, we require them to be close in the ℓ_2 space:

$$\mathcal{J}_{Patt} = \frac{1}{2} ||f(a, b, \theta_f) - g(\mathcal{P}(a, b), \theta_g)||_2^2 \quad (4)$$

CGRE objective function

Learn word pair embeddings that simultaneously minimise both $\mathcal{J}_C$ and $\mathcal{J}_{Patt}$

$$\mathcal{J} = \mathcal{J}_C + \lambda \mathcal{J}_{Patt}$$

$$\mathcal{J}_C = -\sum_{(a,b,r)\in\mathcal{D}} \log \; p(r|f(a, b, \theta_f)) \qquad \mathcal{J}_{Patt} = \frac{1}{2} ||f(a, b, \theta_f) - g(\mathcal{P}(a, b), \theta_g)||_2^2$$

Pseudo Relation Labels

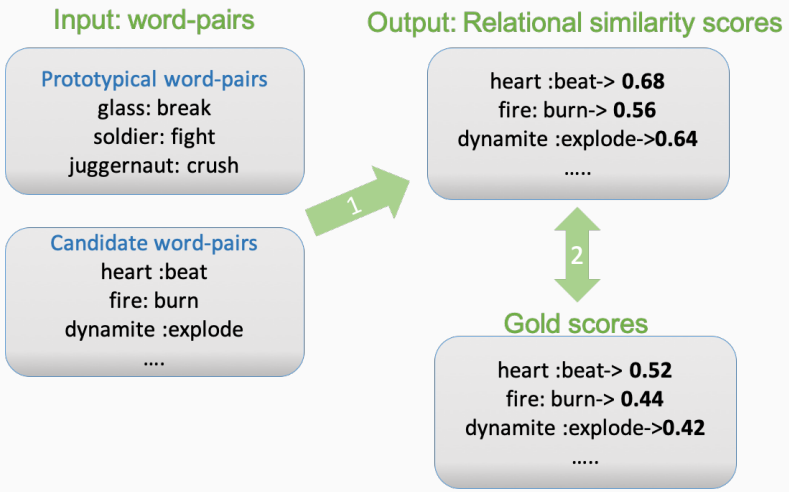
- To train CGRE, we require a dataset containing word-pairs annotated with relation labels
- To make CGRE self-supervised, we induce pseudo labels for word-pairs via k-means clustering.
  - We cluster the PairDiff vectors of the training word pairs $\boldsymbol{b} - \boldsymbol{a}$
  - We evaluate the quality of the generated clusters using the V-measure using the ground truth labels (harmonic mean between homogeneity and completeness of the clusters)
  - Empirically, $k = 50$ clusters performs well with a V-measure of 0.416.

Experimental Settings

Evaluation

Measuring Degrees of Relational Similarity:

- **Dataset**: SemEval-2012 Task2
- Instances of relations can have different degrees of prototypicality
- **Task**: rank pairs based on the extent to which they exhibit a relation.
- 69 fine-grained relations in the test set
- **Metrics:** macro-averaged MaxDiff scores and Spearman correlations



Training data

- **Training data** $\mathcal{D}$: DiffVec dataset that contains $12,458$ triples and 36 fine-grained relations
  - Exclude word-pairs in DiffVec that appear in SemEval test data
- **For relational patterns** $\mathcal{P}$:
  - We use the English Wikipedia corpus (ca. 337M sentences)
  - We use the word-pairs set in DiffVec and their reverse pairs
  - $\mathcal{P}(a, b)$: consists of the contexts of one to five words in which a occurs before b
  - To reduce noise: filter out the patterns that occur between less than ten distinct pairs
  - We obtain $5,017$ contextual patterns

Comparison Methods

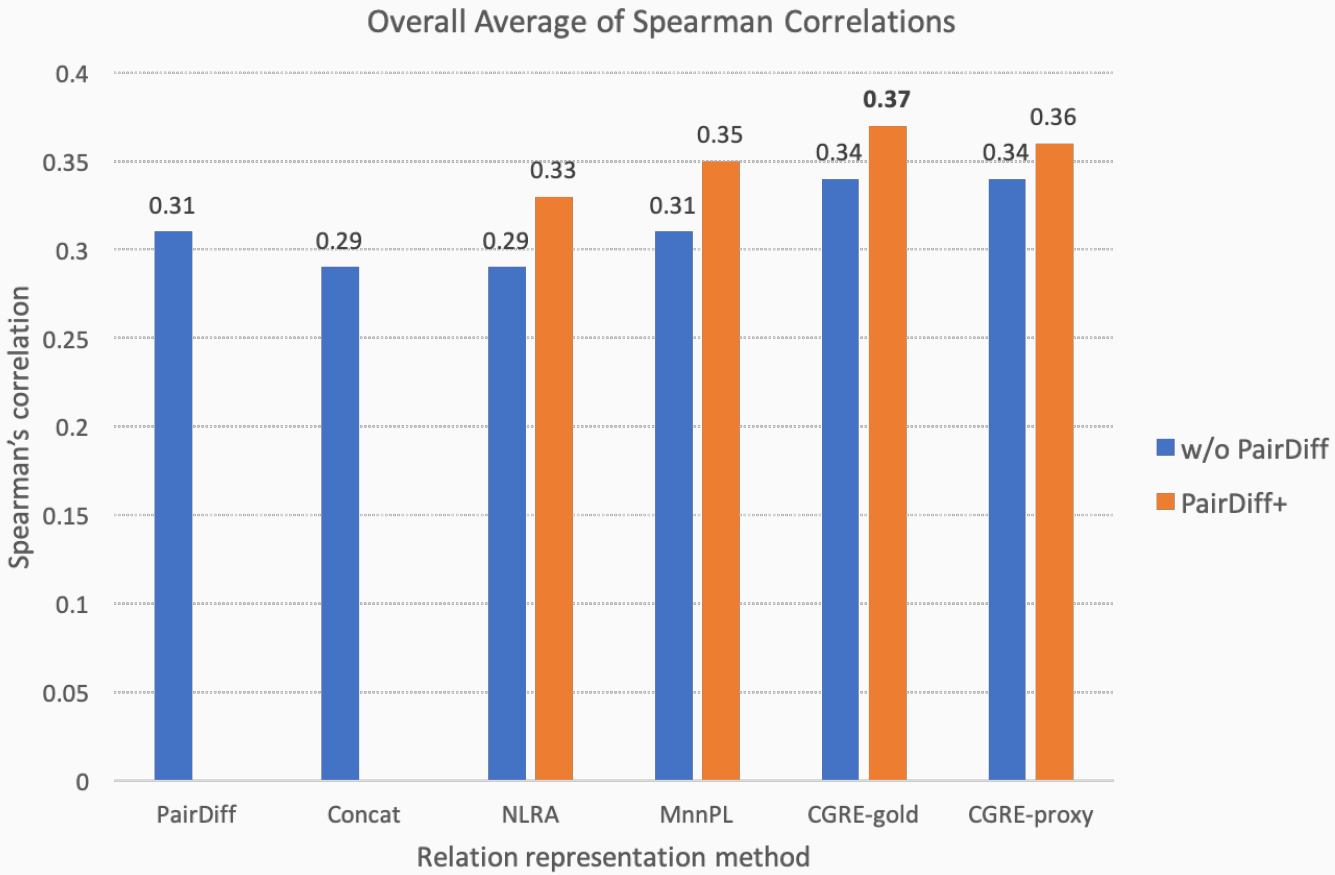
- Non-parametric compositional operators:
  - Vector offset (PairDiff)
  - Concatenation (Concat)
- Supervised methods:
  - MnnPL: learn a multi-class relation classification neural netowork f using a relation labelled word-pairs and does not use contextual patterns (Hakami and Bollegala, 2018)
  - Neural Laten Relational Analysis (Washio and Kato, 2018): we re-train NLRA using the same training data that we used for our proposed method
  - +PairDiff: averaging the scores of a learnt method and the PairDiff score for each target word-pair
  - CGRE is learnt on gold labels (CGRE-gold) and pseudo labels (CGRE-proxy)

Implementation Details

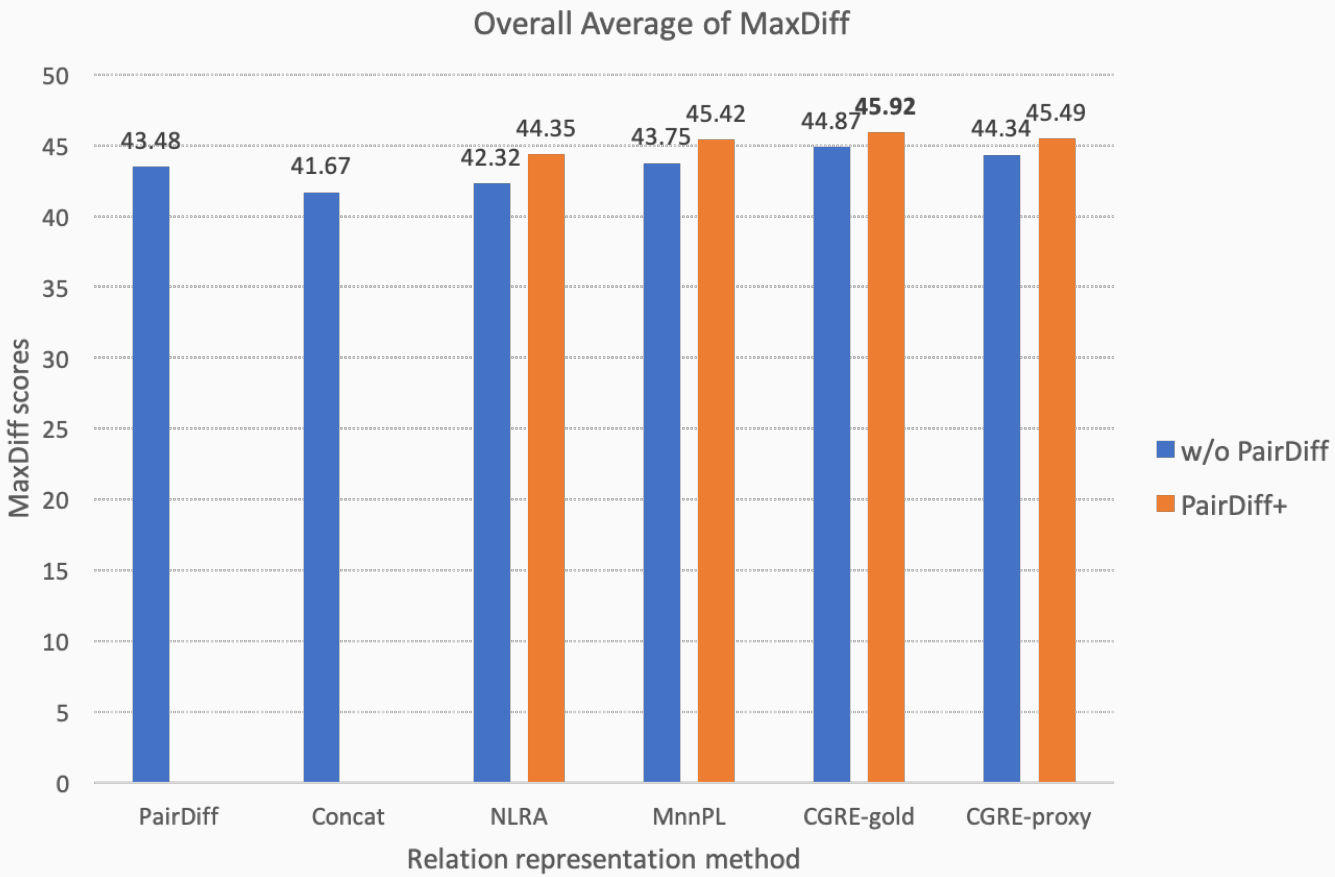
- We use pre-trained 300-dimensional GloVe embeddings (not trainable while learning CGRE)
- $f(a, b, \theta_f)$:
  - Feedforward neural networks with three hidden layers followed by the batch normalisation
  - *tanh* nonlinearity function
  - The size of the hidden layers is set to 300
- $g(\mathcal{P}(a, b), \theta_g)$:
  - A unidirectional LSTM with a 300-dimensional hidden state
- Optimisation algorithm: AdaGrad with mini-batch size 100
- The best model was selected by early stopping using the MaxDiff on the SemEval train set (10 relations)

Experimental Results

Results (1)


Overall Average of Spearman Correlations
Spearman's correlation
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
0.31
0.29
0.29
0.33
0.31
0.35
0.34
0.37
0.34
0.36
PairDiff
Concat
NLRA
MnnPL
CGRE-gold
CGRE-proxy
Relation representation method
w/o PairDiff
PairDiff+

Results (2)


Overall Average of MaxDiff
MaxDiff scores
50
45
40
35
30
25
20
15
10
5
0
43.48
41.67
42.32
44.35
43.75
45.42
44.87
45.92
44.34
45.49
PairDiff
Concat
NLRA
MnnPL
CGRE-gold
CGRE-proxy
Relation representation method
w/o PairDiff
PairDiff+

Results (3)

Breakdown the results for the ten major relations in the 69 SemEval test set

Table II
AVERAGE MAXDIFF AND SPEARMAN CORRELATION FOR EACH MAJOR RELATION IN THE TEST SET OF SEMEVAL 2012-TASK2. THE VALUES BETWEEN PARENTHESES INDICATE THE PERFORMANCE OF A METHOD COMBINED WITH PAIRDIFF.

	MaxDiff				**Correlation**			
Relation	PairDiff	MnnPL	CGRE-Gold	CGRE-Proxy	PairDiff	MnnPL	CGRE-Gold	CGRE-Proxy
CLASS-INCLUSION	48.50	**52.00** (51.60)	51.40 (51.67)	50.45 (49.35)	0.375	0.519 (**0.537**)	0.533 (0.516)	0.515 (0.462)
PART-WHOLE	43.50	41.33 (43.36)	39.61 (42.80)	43.35 (**44.38**)	0.287	0.245 (0.288)	0.228 (0.292)	0.314 (**0.321**)
SIMILAR	41.26	36.20 (41.15)	40.02 (40.82)	**41.68** (41.10)	0.252	0.186 (0.260)	0.245 (**0.286**)	0.280 (0.282)
CONTRAST	33.72	38.57 (38.73)	**40.21** (38.44)	36.39 (36.67)	0.113	0.160 (0.202)	0.209 (**0.226**)	0.157 (0.171)
ATTRIBUTE	46.32	44.84 (47.23)	46.19 (**47.97**)	45.44 (47.83)	0.410	0.351 (0.409)	0.396 (**0.444**)	0.387 (0.437)
NON-ATTRIBUTE	39.11	42.45 (41.82)	42.41 (42.79)	**43.00** (41.85)	0.209	0.264 (0.265)	0.287 (0.279)	**0.313** (0.274)
CASE RELATIONS	46.49	49.53 (49.57)	**52.04** (51.67)	49.46 (50.21)	0.383	0.425 (0.467)	**0.475** (0.466)	0.419 (0.445)
CAUSE-PURPOSE	44.43	44.17 (46.89)	47.57 (**48.59**)	47.74 (48.17)	0.343	0.332 (0.384)	0.422 (**0.436**)	0.400 (0.404)
SPACE-TIME	49.48	45.53 (48.50)	48.62 (**50.21**)	45.36 (49.79)	0.422	0.373 (0.433)	0.432 (**0.455**)	0.385 (0.437)
REFERENCE	41.92	45.94 (**47.84**)	41.32 (44.74)	41.52 (45.74)	0.303	0.323 (**0.377**)	0.212 (0.323)	0.295 (0.375)

Conclusion

Recap

- We consider the problem of representing relations between words.
- We proposed a method that uses the contextual patterns in a corpus to improve the compositional relation representation using word embeddings of the related word-pairs.
- Experimental results show that pattern-based and compositional-based approaches have *complementary* properties when it comes to representing relations.
- Code and pre-trained word-pair embeddings: https://github.com/Huda-Hakami/Context-Guided-Relation-Embeddings

▸ Link

Thanks for your attention . . .

Questions?

h.a.hakami@liverpool.ac.uk

danushka.bollegala@liverpool.ac.uk