Within-Between Lexical Relation Classification

Oren Barkan[*,1] **Avi Caciularu**[*,2,3] **Ido Dagan**[2]
[1]Computer Science Department, Ariel University, Israel
[2]Computer Science Department, Bar-Ilan University, Ramat-Gan, Israel
[3]Allen Institute for Artificial Intelligence
{barkanoren1,avi.c33}@gmail.com, dagan@cs.biu.ac.il

Abstract

We propose the novel *Within-Between* Relation model for recognizing lexical-semantic relations between words. Our model integrates relational and distributional signals, forming an effective sub-space representation for each relation. We show that the proposed model is competitive and outperforms other baselines, across various benchmarks.

1 Introduction and Related Work

Recognizing lexical-semantic relations between words is beneficial for a variety of NLP tasks such as machine translation (Thompson et al., 2019), relation extraction (Shen et al., 2018), natural language inference (Chen et al., 2018), and question answering (Yang et al., 2017).

The lexical relation classification task assigns a word-pair (pair of words) to its corresponding relation out of a finite set of relations. This set contains lexical relations, including the *random* relation (indicating that the words are unrelated). Two main lexical relation classification techniques are studied in the literature: Path-based methods (Hearst, 1992; Snow et al., 2005; Nakashole et al., 2012; Riedel et al., 2013) and distributional methods (Mikolov et al., 2013a; Pennington et al., 2014; Bojanowski et al., 2017; Glavaš and Vulić, 2018a), with some effort for integrating the two (Shwartz et al., 2016a).

In this work we follow the distributional approach, which was shown to improve upon path-based methods. This approach considers static word embeddings such as word2vec (Mikolov et al., 2013b), GloVe (Pennington et al., 2014),and FastText (Bojanowski et al., 2017), which produce out-of-context vector representation for each word. Note here that while *contextualized embeddings* (Devlin et al., 2019; Peters et al., 2018) have replaced the use of non-contextualized embeddings in many settings, static word embeddings remain the standard choice for lexical relation classification, since in this task the input word-pair is given out-of-context. Taking the word embeddings as input, a classifier is trained while considering each word's representation in the pair. The recent SphereRE method (Wang et al., 2019), a purely distributional method that learns hyperspherical relation representation, presented state-of-the-art lexical relation classification results.

While presenting state-of-the-art performance, prior distributional methods suffer from the "lexical memorization" problem Levy et al. (2015). This problem arises when a test word-pair includes a rather frequent word in the training set, which is labeled by a dominant category in training. In such cases, the supervised model often ignores the second word in the input pair and resorts to the dominant training label according to the frequent word. Notably, lexical memorization is common for *prototypical hypernyms* — "category" words that are frequently labeled as hypernyms. For example, the vast majority of training examples that include the word *fruit* are labeled as hypernymy (*fruit* is the hypernym of *apple*, *banana*, etc.). Therefore, at inference time, the classifier is likely to predict the hypernymy relation even for unrelated word-pairs that contain *fruit*, e.g., *(fruit,chair)*.

Another relevant line of research, which inspired our work, pertains to the integration of external lexical information to improve static word embeddings (Faruqui et al., 2015; Mrkšić et al., 2016; Glavaš and Vulić, 2018b; Arora et al., 2020; Barkan et al., 2020). Most of these methods aim to modify the distributional vector space, originally learned from corpus co-occurrence data, by using additional relational constraints. To that end, these techniques rely on lexical databases, e.g., Wordnet (Miller, 1995). Notably, Arora et al. (2020) present the LEXSUB model and suggest training static word

[*] Equal contribution, order determined randomly.

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 3521–3527,
November 16–20, 2020.

embeddings by integrating lexical-relation and distributional data, through the combination of two corresponding loss terms. When modeling lexical relations data, each relation is projected to a separate subspace. Some of these ideas are adapted in certain parts of our work, while addressing the concrete goal of lexical relation classification rather than improving generic static word embeddings.

In this work, we present the novel *Within-Between* Relation (WBR) model, which is inspired both by previous relation classification models as well as by generic word embedding models that consider lexical relation constraints. This is performed through the combination of two objectives, both computed over the same projected sub-spaces, for each of the individual relations. Specifically, our *Between* objective aims to yield optimal classification of relation instances, while the *Within* objective aims to bring pairs of words sharing a relation close to each other in the corresponding relation sub-space. These objectives allow the incorporation of both relation and negative sampling signals, altogether addressing much better the lexical memorization problem.

2 The *Within-Between* Relation Model

In this section, we present the WBR model. Given a word-pair sharing a relation k, WBR is optimized to classify a word-pair to the correct relation (the *between* relation objective), and at the same time, separate it *within* the k relation space from other word-pairs that do not share the relation k. Let $\mathcal{I}$ and $\mathcal{K}$ be vocabularies of words and relations, respectively. $\mathcal{K}$ contains lexical relations such as *hypernym, antonym*, etc., including the *random* relation (words are unrelated) and the *co* (stands for co-occurrence) relation that is shared by words that co-occur in the corpus. We further denote $\mathcal{P} = \mathcal{I} \times \mathcal{I}$.

Let $\mathbf{u}_i, \mathbf{v}_i \in \mathbb{R}^d$ be random variables that form the context and target *base* representations for the word i. We further denote $\mathbf{U} = \{\mathbf{u}_i\}_{i\in\mathcal{I}}$ and $\mathbf{V} = \{\mathbf{v}_i\}_{i\in\mathcal{I}}$.

We assume normal priors $p(\mathbf{u}_i|\mathbf{a}_i, \tau) = \mathcal{N}(\mathbf{u}_i; \mathbf{a}_i, \tau^{-1}\mathbf{I})$ and $p(\mathbf{v}_i|\mathbf{b}_i, \tau) = \mathcal{N}(\mathbf{v}_i; \mathbf{b}_i, \tau^{-1}\mathbf{I})$, where $\mathbf{a}_i, \mathbf{b}_i \in \mathbb{R}^d$ are either zero or set to a pretrained embedding (that can be retrieved from any word embedding method such as FastText, word2vec, Glove, etc.), and τ is a precision hyperparameter. We further denote $\mathbf{A} = \{\mathbf{a}_i\}_{i\in\mathcal{I}}$ and $\mathbf{B} = \{\mathbf{b}_i\}_{i\in\mathcal{I}}$.

Let $I_k = \{(i,j)|i \xrightarrow{k} j\}$, where $i \xrightarrow{k} j$ means that words i and j share a directed relation k. In case of an undirected relation, it holds that $(i \xrightarrow{k} j) \leftrightarrow (j \xrightarrow{k} i)$. Note that in the specific case of the *random* relation, $I_{random} = \{(i,j)|(i,j) \notin \bigcup_{k\in\mathcal{K}\setminus\{random\}} I_k\}$. This assumption guarantees each word-pair $(i,j) \in \mathcal{P}$ is associated with a relation $k \in \mathcal{K}$.

Let $f^k : \mathcal{P} \rightarrow \mathbb{R}$ be a parametric function. Our goal is to learn parameters for f^k s.t. the score f^k_{ij} is high if and only if $(i,j) \in I_k$. In this work, we define $f^k_{ij} \triangleq \alpha \frac{\mathbf{p}^k_i \cdot \mathbf{q}^k_j}{||\mathbf{p}^k_i||_2 ||\mathbf{q}^k_j||_2}$, where $\mathbf{p}^k_i = \mathbf{\Psi}_k \mathbf{u}_i$, $\mathbf{q}^k_i = \mathbf{\Phi}_k \mathbf{v}_j$, and α is a hyperparameter. This forms a cosine similarity similarity metric with temperature, is motivated in Sec. 4. $\mathbf{\Psi}_k \in \mathbb{R}^{d_k \times d}$ and $\mathbf{\Phi}_k \in \mathbb{R}^{d_k \times d}$ are matrices whose entries have normal priors with zero mean and precision λ (hyperparameter). Therefore, f^k learns $\mathbf{\Psi}_k$ and $\mathbf{\Phi}_k$ that enable the projection to a new relation space k. In this space, word-pairs that share the relation k are separated from word-pairs that do not in terms of the angle between their repective vectors. Yet, in the general case, f^k can be a deep neural network. An exception is $k = co$, for which $\mathbf{\Psi}_k$ and $\mathbf{\Phi}_k$ are predetermined to $\mathbf{\Psi}_k = \mathbf{\Phi}_k = \mathbf{I}$ (not learned). We further denote $\mathbf{\Psi} = \{\mathbf{\Psi}_k\}_{k\in\mathcal{K}}$ and $\mathbf{\Phi} = \{\mathbf{\Phi}_k\}_{k\in\mathcal{K}}$. Finally, we denote the set of unobserved variables and the set of hyperparameters by $\mathbf{\Theta} = \{\mathbf{U}, \mathbf{V}, \mathbf{\Psi}, \mathbf{\Phi}\}$ and $\mathbf{H} = \{\mathbf{A}, \mathbf{B}, \tau, \lambda\}$, respectively.

2.1 The *Within*-Relation Likelihood

We define $\mathbf{Y}^k = \{y^k_{ij}|(i,j) \in \mathcal{P}\}$, where $y^k_{ij} : \mathcal{P} \rightarrow \{1, -1\}$ is an observed random variable, s.t. $y^k_{ij} = 1$ if $(i,j) \in I_k$, otherwise, $y^k_{ij} = -1$. We further denote $\mathbf{Y} = \{\mathbf{Y}_k\}_{k\in\mathcal{K}}$ and $\sigma(x) = (1 + e^{-x})^{-1}$. Then, the *within*-relation likelihood is given by:

$$p(\mathbf{Y}|\mathbf{\Theta}) = \prod_{(i,j)\in\mathcal{P}} \prod_{k\in\mathcal{K}} p(y^k_{ij}|\mathbf{u}_i, \mathbf{v}_j, \mathbf{\Psi}_k, \mathbf{\Phi}_k)$$
$$= \prod_{(i,j)\in\mathcal{P}} \prod_{k\in\mathcal{K}} \sigma(y^k_{ij} f^k_{ij}).$$

2.2 The *Between*-Relation Likelihood

Denote $\mathcal{K}' = \mathcal{K} \setminus \{co\}$ and let $\mathbf{R} = \{r_{ij}|(i,j) \in \mathcal{P}\}$, where $r_{ij} : \mathcal{P} \rightarrow \mathcal{K}'$ is an observed categorical random variable s.t. $r_{ij} = k$ if $(i,j) \in I_k$. Then, the *between*-relation

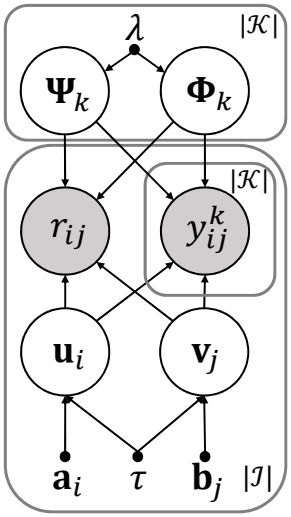

Figure 1: The WBR graphical model.

likelihood is given by:

$$p(\mathbf{R}|\boldsymbol{\Theta}) = \prod_{(i,j)\in\mathcal{P}} p(r_{ij}|\mathbf{u}_i, \mathbf{v}_j, \boldsymbol{\Psi}, \boldsymbol{\Phi}) = \prod_{(i,j)\in\mathcal{P}} \frac{\exp f_{ij}^{r_{ij}}}{\sum_{k\in\mathcal{K}'} \exp f_{ij}^k}$$

2.3 WBR Training and Inference

The *vanilla* WBR loss is derived by taking the negative log of the joint distribution as follows:

$$\begin{aligned}\mathcal{L}_{\text{vanilla}}(\boldsymbol{\Theta}) &= -\log p(\mathbf{Y}, \mathbf{R}, \boldsymbol{\Theta}|\mathbf{H}) \\ &= -\log\left[p(\mathbf{Y}|\boldsymbol{\Theta})p(\mathbf{R}|\boldsymbol{\Theta})p(\boldsymbol{\Theta}|\mathbf{H})\right] \\ &= -\sum_{(i,j)\in\mathcal{P}}\sum_{k\in\mathcal{K}} \log\sigma(y_{ij}^k f_{ij}^k) \\ &\quad -\sum_{(i,j)\in\mathcal{P}} f_{ij}^{r_{ij}} + \log\sum_{k\in\mathcal{K}'}\exp f_{ij}^k \\ &\quad +\frac{\tau}{2}\sum_{i\in\mathcal{I}} ||\mathbf{u}_i - \mathbf{a}_i||_2^2 + ||\mathbf{v}_i - \mathbf{b}_i||_2^2 \\ &\quad +\frac{\lambda}{2}\sum_{k\in\mathcal{K}'} ||\boldsymbol{\Psi}_k||_2^2 + ||\boldsymbol{\Phi}_k||_2^2 + \text{const.}\end{aligned} \tag{1}$$

A graphical model of WBR is presented in Fig. 1. The minimization of $\mathcal{L}_{\text{vanilla}}(\boldsymbol{\Theta})$ is equivalent to the Maximum A-*Posteriori* (MAP) estimation of $\boldsymbol{\Theta}$. However, the negative log likelihood terms in Eq. 1 contain a summation which is quadratic in the vocabulary size $\mathcal{I}$, implying a prohibitive computation. Therefore, we turn to a stochastic optimization: Let C be a text corpus (a sequence of words), and $Q_i^k = \{j|(i,j) \in I_k\}$. We define $s^k : \mathcal{I} \to \mathcal{I} \times \mathcal{I}$ as a sampler s.t. $s^k(i)$ retrieves a random word-pair $(i,j) \in Q_i^k$ if $Q_i^k \neq \emptyset$, otherwise, a random word-pair $(a,b) \in I_k$. The WBR stochastic optimization algorithm is described in Algorithm 1, together with the $\mathcal{L}_{\text{wbr}}$ loss function in Eq. 2.

Algorithm 1 WBR Stochastic Optimization

for $z \leftarrow 1$ to T **do**
  for i in C **do**
    for k in $\mathcal{K}$ **do**
      $\mathcal{P}_k \leftarrow \emptyset$
    end for
    Sample a positive word j (within the window around i), and a negative word $n \in \mathcal{I}$
    $y_{i,j}^{co} \leftarrow 1$, $y_{i,n}^{co} \leftarrow -1$,
    $\mathcal{I}_u \leftarrow \mathcal{I}_u \cup \{i\}$, $\mathcal{I}_v \leftarrow \mathcal{I}_v \cup \{j, n\}$
    $\mathcal{P}_{co} \leftarrow \mathcal{P}_{co} \cup \{(i,j),(i,n)\}$
    for k in $\mathcal{K}'$ **do**
      $(a,b) \leftarrow s^k(i)$
      Sample n s.t. $(a,n) \notin I_k$
      $y_{a,b}^k \leftarrow 1$, $y_{a,n}^k \leftarrow -1$
      $\mathcal{I}_u \leftarrow \mathcal{I}_u \cup \{a\}$, $\mathcal{I}_v \leftarrow \mathcal{I}_v \cup \{b, n\}$
      $\mathcal{P}_k \leftarrow \mathcal{P}_k \cup \{(a,b),(a,n)\}$
      $r_{ab} \leftarrow k$
      if $k \neq$ *random* **then**
        $r_{an} \leftarrow$ *random*
      end if
    end for
    $\boldsymbol{\Theta} \leftarrow \text{OPT}(\boldsymbol{\Theta}, \mathcal{L}_{\text{wbr}})$ (See Eq. 2)
  end for
end for

$$\begin{aligned}\mathcal{L}_{\text{wbr}}(\boldsymbol{\Theta}) &= -\sum_{k\in\mathcal{K}}\sum_{(i,j)\in\mathcal{P}_k} \log\sigma(y_{ij}^k f_{ij}^k) \\ &\quad -\sum_{k\in\mathcal{K}'}\sum_{(i,j)\in\mathcal{P}_k} f_{ij}^{r_{ij}} + \log\sum_{k\in\mathcal{K}'}\exp f_{ij}^k \\ &\quad +\frac{\tau}{2}\sum_{i\in\mathcal{I}_u} ||\mathbf{u}_i - \mathbf{a}_i||_2^2 \\ &\quad +\frac{\tau}{2}\sum_{i\in\mathcal{I}_v} ||\mathbf{v}_i - \mathbf{b}_i||_2^2 \\ &\quad +\frac{\lambda}{2}\sum_{k\in\mathcal{K}'} ||\boldsymbol{\Psi}_k||_2^2 + ||\boldsymbol{\Phi}_k||_2^2.\end{aligned} \tag{2}$$

Finally, in the inference phase, the probability of $i \xrightarrow{k} j$ is computed by $p(r_{ij} = k|\boldsymbol{\Theta}^*)$, where $\boldsymbol{\Theta}^*$ is the MAP estimate (produced by Algorithm 1).

3 Experimental Setup and Results

In this section, we present the datasets, hyperparameters, and experiments that we conducted to

Model	K&H+N			BLESS			ROOT09			EVALution		
	P	R	F1	P	R	F1	P	R	F1	P	R	F1
Concat	0.909	0.906	0.904	0.811	0.812	0.811	0.636	0.675	0.646	0.531	0.544	0.525
Diff	0.888	0.886	0.885	0.801	0.803	0.802	0.627	0.655	0.638	0.521	0.531	0.528
NPB	0.713	0.604	0.550	0.759	0.756	0.755	0.788	0.789	0.788	0.530	0.537	0.503
NPB+Aug	-	-	0.897	-	-	0.842	-	-	0.778	-	-	0.489
LexNET	0.985	0.986	0.985	0.894	0.893	0.893	0.813	0.814	0.813	0.601	0.607	0.600
LexNET+Aug	-	-	0.970	-	-	0.927	-	-	0.806	-	-	0.545
SphereRE	**0.990**	**0.989**	**0.990**	0.938	0.938	0.938	**0.860**	0.862	0.861	0.620	**0.621**	0.620
BR	0.988	0.985	0.986	0.937	0.935	0.936	0.855	0.859	0.857	0.543	0.601	0.571
BR+*co*	0.989	0.986	0.987	0.940	0.938	0.939	0.856	0.863	0.859	0.576	0.608	0.591
WBR	0.989	0.988	0.988	**0.942**	**0.940**	**0.941**	0.856	**0.872**	**0.864**	**0.636**	0.620	**0.628**

Table 1: Precision, Recall and F1 results over lexical relation classification benchmarks. Best results are bolded.

evaluate our model and compare it with other methods.

3.1 Benchmarks and Co-Occurrence Data

In order to evaluate our model, we adopted the same experimental setup from Wang et al. (2019). The lexical relation classification datasets that were considered are K&H+N (Necşulescu et al., 2015), BLESS (Baroni and Lenci, 2011), ROOT09 (Santus et al.) and EVALution (Santus et al., 2015). Since the EVALution benchmark does not contain the *random* relation, we add it artificially for the negative sampling purpose. Due to space limitations, we do not provide the datasets' statistics. The reader may refer to (Wang et al., 2019) for the full details of the datasets.

For co-occurrence data, we extracted co-occurring word-pairs from the English Wikipedia corpus. We sampled co-occurrence data that correspond to the vocabulary of the relation classification dataset, by picking the sentences form the corpus that contain these words.

3.2 Evaluated Models

For baselines, we considered both traditional distributional models: Concat (Baroni et al., 2012) and Diff (Weeds et al., 2014), and path-based models NPB (Shwartz et al., 2016b), LexNET (Shwartz and Dagan, 2016) (which integrates both distributional model and pure path-based data), NPB+Aug and LexNET+Aug (the base models are trained on augmented dependency paths, used to improve coverege) (Washio and Kato, 2018), and the recent state-of-the-art model SphereRE (Wang et al., 2019). Note that SphereRE performs a pre-training phase for generating initial pseudo labels, and the (unlabeled) test data is used for both this phase and the training. Our method does not require the test data and does not perform and initial classification before training. We refer readers to the previous works for detailed descriptions of these baselines. Note that (Washio and Kato, 2018) reported only the F1 scores over the models that were trained using augmented dependency paths.

3.2.1 Ablation Study

In order to assess the contribution of each component in our model, we perform an ablation study. First, we denote WBR as the full model that is described in Section 2. In addition, we consider the following ablated versions of WBR:

BR: In this version, we omit the within loss and do not learn $\mathbf{U}$ and $\mathbf{V}$. Instead, we set $\mathbf{U} = \mathbf{A}$ and $\mathbf{U} = \mathbf{B}$ and keep them fixed for the entire optimization procedure. This leads to the following loss:

$$\mathcal{L}_{\text{br}}(\mathbf{\Theta}) = -\sum_{k\in\mathcal{K}'}\sum_{(i,j)\in\mathcal{P}_k} f_{ij}^{r_{ij}} + \log\sum_{k\in\mathcal{K}'}\exp f_{ij}^{k} + \frac{\lambda}{2}\sum_{k\in\mathcal{K}'}||\mathbf{\Psi}_k||_2^2 + ||\mathbf{\Phi}_k||_2^2.$$

BR+*co*: In this version, we omit all the relations from the *within*-relation loss, except for the *co* (co-occurence) relation. In other words, we change the WBR loss to include the *between*-relation likelihood, co-occurrence likelihood and the priors as follows:

$$\begin{aligned}\mathcal{L}_{\text{br+}co}(\mathbf{\Theta}) = &-\sum_{(i,j)\in\mathcal{P}_{co}}\log\sigma(y_{ij}^{co}f_{ij}^{co}) \\ &-\sum_{k\in\mathcal{K}'}\sum_{(i,j)\in\mathcal{P}_k} f_{ij}^{r_{ij}} + \log\sum_{k\in\mathcal{K}'}\exp f_{ij}^{k} \\ &+\frac{\tau}{2}\sum_{i\in\mathcal{I}_u}||\mathbf{u}_i - \mathbf{a}_i||_2^2 \\ &+\frac{\tau}{2}\sum_{i\in\mathcal{I}_v}||\mathbf{v}_i - \mathbf{b}_i||_2^2 \\ &+\frac{\lambda}{2}\sum_{k\in\mathcal{K}'}||\mathbf{\Psi}_k||_2^2 + ||\mathbf{\Phi}_k||_2^2.\end{aligned}$$

3.3 Hyperparameters Configuration

We set the projection dimension to $d_k = 15$, the precisions to $\lambda = \tau = 10^{-4}$, and the temperature to $\alpha = 5$. Either increasing d_k or changing the precisions or the temperature did not improve the performance of WBR over the validation sets. We used the Adam optimizer (Kingma and Ba, 2015) (as OPT from Algorithm 1) with a minibatch size of 32. Similar to Wang et al. (2019), we initialized the word-level representations to the pretrained 300 dimensional FastText word embeddings (Bojanowski et al., 2017). The SphereRE model uses constant FastText word embeddings and only learns relations' embeddings. However, we train both the relations' projections and continued the training of the word embeddings. For the rest of the baselines, the hyperparameters are adopted from the corresponding papers. For training stopping criteria, we used the validation set within each dataset (by computing the F1 score). For each test set, we report the averaged precision, recall, and F1 score for each lexical relation.

3.4 Results

The results of WBR and all of the baselines over the datasets are summarized in Table 1. Overall, our WBR model provides competitive performance results comparing to the tested baseline models. The recent SphereRE model outperforms the basic BR model on all the datasets. Adding the *within*-relation objective, but only with co-occurrence (BR+*co*) improved the performance. The results on the other benchmarks are close to SphereRE. Adding both the *random* relation and using the *within*-relation mechanism increased the performance gain over this dataset, which is extremely imbalanced compared to the rest of the datasets (Wang et al., 2019). The improvement on EVALution is reasonable since this dataset does not contain the *random* relation. This effect can be explained by addressing the lexical memorization problem. Finally, adding the relations data to the *within*-relation objective (the full WBR model) yielded additional performance gain, causing the model to outperform the SphereRE over three datasets slightly.

4 Mitigating the Lexical Memorization Problem

The lexical memorization problem is alleviated by the introduction of the *random* relation. This feature plays a key role both for the *between* and *within* loss terms: given a word-pair (a, b) and their corresponding, ground truth relation r, we randomly sample a word n and associate the word-pair (a, n) with the *random* relation, unless r happens to be equal to the *random* relation beforehand (see Algorithm 1). This unique mechanism is designed to balance each positive word-pair with a negative one, neutralizing the effect of multiple instances of the prototypical terms (e.g., *animal*, *fruit*, etc.) on the training objective, as a kind of regularization and data augmentation. For example, consider the positive data sample $(animal, b)$. It will be balanced with a negative sample $(animal, n)$. Therefore, during the training phase, the *between* classifier learns to consider the *random* label each time it is given the *hypernymy* label. Similarly, the corresponding *within* (*hypernymy*) classifier will encounter a negative sample for each positive sample. As a result, in the inference phase, the relation classifier does not always predict the *hypernym* relation for $(animal, x)$ - the classifier will consider the features of x as well, and thus mitigates the lexical memorization problem. Another way to ensure that each relation classifier exploits both words in the given pair is splitting them into two different linear projections' relation sets, $\mathbf{\Psi}$ for the first word, and $\mathbf{\Phi}$ for the second. Further, using the cosine similarity measure for computing f_{ij}^k, rather than a dot product, provides a normalization effect which neutralizes frequency biases, caused by typical larger norms for frequent words.

5 Conclusions

We presented WBR - a novel model for lexical relation classification. WBR facilitates the novel *between-within* relation loss, enabling the exploitation of distributional information. WBR is evaluated on four different datasets, where it is shown to outperform various baselines across all evaluation metrics.

Acknowledgements

The authors would like to thank the anonymous reviewers for their comments and suggestions. This work was supported in part by grants from Intel

Labs, the Israel Science Foundation grant 1951/17 and the German Research Foundation through the German-Israeli Project Cooperation (DIP, grant DA 1600/1-1).

References

Kushal Arora, Aishik Chakraborty, and Jackie C. K. Cheung. 2020. Learning lexical subspaces in a distributional vector space. *Transactions of the Association for Computational Linguistics (TACL)*.

Oren Barkan, Idan Rejwan, Avi Caciularu, and Noam Koenigstein. 2020. Bayesian hierarchical words representation learning. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Marco Baroni, Raffaella Bernardi, Ngoc-Quynh Do, and Chung-chieh Shan. 2012. Entailment above the word level in distributional semantics. In *Conference of the European Chapter of the Association for Computational Linguistics EACL*.

Marco Baroni and Alessandro Lenci. 2011. How we BLESSed distributional semantic evaluation. In *Proceedings of the GEMS 2011 Workshop on GEometrical Models of Natural Language Semantics*, pages 1–10, Edinburgh, UK. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics (TACL)*.

Yufei Chen, Sheng Huang, Fang Wang, Junjie Cao, Weiwei Sun, and Xiaojun Wan. 2018. Neural maximum subgraph parsing for cross-domain semantic dependency analysis. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 562–572, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Manaal Faruqui, Jesse Dodge, Sujay Kumar Jauhar, Chris Dyer, Eduard Hovy, and Noah A. Smith. 2015. Retrofitting word vectors to semantic lexicons. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Goran Glavaš and Ivan Vulić. 2018a. Discriminating between lexico-semantic relations with the specialization tensor model. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Goran Glavaš and Ivan Vulić. 2018b. Explicit retrofitting of distributional word vectors. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *COLING 1992 Volume 2: The 15th International Conference on Computational Linguistics*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Omer Levy, Steffen Remus, Chris Biemann, and Ido Dagan. 2015. Do supervised distributional methods really learn lexical inference relations? In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 970–976, Denver, Colorado. Association for Computational Linguistics.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013a. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems (NIPS)*.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*.

Nikola Mrkšić, Diarmuid O Séaghdha, Blaise Thomson, Milica Gašić, Lina Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Ndapandula Nakashole, Gerhard Weikum, and Fabian Suchanek. 2012. Patty: a taxonomy of relational patterns with semantic types. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP)*.

Silvia Necşulescu, Sara Mendes, David Jurgens, Núria Bel, and Roberto Navigli. 2015. Reading between the lines: Overcoming data sparsity for accurate classification of lexical relationships. In *Proceedings of the Joint Conference on Lexical and Computational Semantics (*SEM)*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Sebastian Riedel, Limin Yao, Andrew McCallum, and Benjamin M. Marlin. 2013. Relation extraction with matrix factorization and universal schemas. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 74–84, Atlanta, Georgia. Association for Computational Linguistics.

Enrico Santus, Alessandro Lenci, Tin-Shing Chiu, Qin Lu, and Chu-Ren Huang. Nine features in a random forest to learn taxonomical semantic relations. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation LREC, year = 2016,*.

Enrico Santus, Frances Yung, Alessandro Lenci, and Chu-Ren Huang. 2015. EVALution 1.0: an evolving semantic dataset for training and evaluation of distributional semantic models. In *Proceedings of the 4th Workshop on Linked Data in Linguistics: Resources and Applications*, pages 64–69, Beijing, China. Association for Computational Linguistics.

Jiaming Shen, Zeqiu Wu, Dongming Lei, Chao Zhang, Xiang Ren, Michelle T. Vanni, Brian M. Sadler, and Jiawei Han. 2018. Hiexpan: Task-guided taxonomy construction by hierarchical tree expansion. In *Proceedings of the International Conference on Knowledge Discovery & Data Mining, (KDD), August 19-23, 2018*.

Vered Shwartz and Ido Dagan. 2016. Path-based vs. distributional information in recognizing lexical semantic relations. In *Proceedings of the Workshop on Cognitive Aspects of the Lexicon, CogALex@COLING*.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016a. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2389–2398, Berlin, Germany. Association for Computational Linguistics.

Vered Shwartz, Yoav Goldberg, and Ido Dagan. 2016b. Improving hypernymy detection with an integrated path-based and distributional method. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*.

Rion Snow, Daniel Jurafsky, and Andrew Y. Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In *Advances in Neural Information Processing Systems*.

Brian Thompson, Rebecca Knowles, Xuan Zhang, Huda Khayrallah, Kevin Duh, and Philipp Koehn. 2019. Hablex: Human annotated bilingual lexicons for experiments in machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, (EMNLP-IJCNLP)*.

Chengyu Wang, Xiaofeng He, and Aoying Zhou. 2019. SphereRE: Distinguishing lexical relations with hyperspherical relation embeddings. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1727–1737, Florence, Italy. Association for Computational Linguistics.

Koki Washio and Tsuneaki Kato. 2018. Filling missing paths: Modeling co-occurrences of word pairs and dependency paths for recognizing lexical semantic relations. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, (NAACL-HLT)*.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David J. Weir, and Bill Keller. 2014. Learning to distinguish hypernyms and co-hyponyms. In *International Conference on Computational Linguistics COLING*.

Shuo Yang, Lei Zou, Zhongyuan Wang, Jun Yan, and Ji-Rong Wen. 2017. Efficiently answering technical questions—a knowledge graph approach. In *Thirty-First AAAI Conference on Artificial Intelligence*.