Word Relation Autoencoder for Unseen Hypernym Extraction Using Word Embeddings

Hong-You Chen, Cheng-Syuan Lee and **Keng-Te Liao**
Dept of Computer Science & Information Engineering
National Taiwan University
{b03902128}, {b03902009}, {d05922001}@ntu.edu.tw

Shou-De Lin
Dept of Computer Science & Information Engineering
National Taiwan University
Research Center for Information Technology, Academia Sinica
sdlin@csie.ntu.edu.tw

Abstract

Lexicon relation extraction given distributional representation of words is an important topic in NLP. We observe that the state-of-the-art projection-based methods cannot be generalized to handle unseen hypernyms. We propose to analyze it in the perspective of pollution, that is, the predicted hypernyms are limited to those appeared in training set. We propose a word relation autoencoder (WRAE) model to address the challenge and construct the corresponding indicator to measure the pollution. Experiments on several hypernym-like lexicon datasets show that our model outperforms the competitors significantly.

1 Introduction

This paper discusses the inference of relations between words. For the hypernym *beer IsA drink*, , denoted as *IsA(x, y)*, *beer* is the hyponym x and *drink* serves as the hypernym y. Relation lexicons are precious resource for NLP systems, while constructing the semantic graphs such as WordNet (Fellbaum, 1998) and ConceptNet (Speer and Havasi, 2012) requires expensive human efforts for labeling.

Recently, researchers have started working on extracting word relations based on pre-trained word embedding without the need of an existing corpus, thanks to the success of distributional word representation models such as GloVe (Pennington et al., 2014).

Query	Answer
$beef \rightarrow meat \approx crab \rightarrow ?$	seafood
$tiger \rightarrow zoo \approx dolphin \rightarrow ?$	aquarium
$paint \rightarrow artist \approx book \rightarrow ?$	writer
$japan \rightarrow asia \approx italy \rightarrow ?$	europe

Table 1: Unseen relation extraction examples for IsA, AtLocation, CreatedBy, and PartOf (top row to bottom row) in ConceptNet. The answers are not appeared in training.

Comparing with hypernym classification models (Lenci and Benotto, 2012; Weeds et al., 2014; Levy et al., 2015; Vylomova et al., 2016) that take a pair of entities *(x,y)* as inputs and output a binary decision about the existence of relation, there has been less work focusing on hypernym extraction task. It is a challenging task to automatically extract all possible hypernyms of a given hyponym query, especially the unlabeled ones, from the vocabularies.

Classification-based models are not applicable for this task because the complexity of inference is $\mathcal{O}(V)$, where V is the size of vocabulary that often scales to billions.

Among the existing solutions, projection-based methods (Fu et al., 2014; Yamane et al., 2016; Espinosa-Anke et al., 2016; Ustalov et al., 2017) emphasize on hypernym extraction which intuitively represent a relation as $y - x$ according to the linear structure of word embedding. By directly learning a linear mapping Φ between two words such that $x\Phi = y$, the prediction $\hat{y}$ can be obtained with nearest neighbor search for $x\Phi$ in the word embedding space. Moreover, the potential candidates of y are not required to be seen in advance so that the method can be used to predict unseen hypernym directly.

Fu et al. (2014) further observe the existence of

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 4834–4839
Brussels, Belgium, October 31 - November 4, 2018.

cluster structures in relation representation $y - x$ and propose to learn a piecewise linear mapping such that $x\Phi_k = y$ for each cluster C_k. Their experiments show that domain clustering on training offset is very useful for hypernym identification.

However, we observe that each cluster contains very few distinct hypernyms. For instance, about 83% of the clusters contain fewer than 5 hypernyms for ConceptNet-IsA in our experiments. Hypernyms can be seen as the collections of related word pairs, e.g., *IsA(dog, animal)*, *IsA(cat, animal)*, *IsA(horse, animal)*, ... etc. The piecewise projection matrices can hardly learn the inference between hyponyms and hypernyms but only *memorize* some words which serve as the hypernyms in the training data. Inevitably, the state-of-the-art models using piecewise projection learning face generalization problem and fail to predict unseen hypernyms correctly.

We design a novel **Word Relation Autoencoder (WRAE)** framework, which adopts the conditional autoencoder structure ($x \to r \to x'$) that encodes hyponyms and reconstructs itself by decoding from $r = y - x$. The weights of encoder are further tied with decoder which is imposed to learn how to separate the hypernym and the hyponym from the relation vectors and extract the hyponym x with the intention to optimize reconstruction loss, thus effectively mitigates the mentioned generalization problem.

We summarize our main contributions as follows: (1) We propose a novel, yet more general scenario for relation extraction to handle unseen hypernyms. (2) We propose an intuitive pollution indicator that allows us to empirically measure whether the model learns the inference between a relation pair or not. (3) We propose a novel Word Relation Autoencoder (WRAE) which can effectively reduce pollution. We conduct thorough experiments to show that our model outperforms the competitors, and can be applied to other hypernym-like relations.

2 Related Work

Fu et al. (2014) first apply projection learning for generalized hypernym extraction by learning a linear transformation from a hyponym word embedding to the corresponding hypernym word vector. They further conduct piecewise projection learning, i.e., learning a projection matrix for each cluster and harvest significant improvements by first applying k-means clustering. They perform training with stochastic gradient descent methods, implying good potential for attaching different regularizers for optimization. Several recent works also follow the schema as the one proposed by Espinosa-Anke et al. (2016) and operate the similar model at the sense level and took advantage of domain clustering to discover hypernyms through domain adaption between different topics. Yamane et al. (2016) focus on improving the performance through better cluster assignments by learning clustering and the projections jointly. Ustalov et al. (2017) propose several regularization terms in addition to the original loss function (Fu et al., 2014) using extra synonym pairs or the asymmetric property of hypernym. Nayak (2015) provides detailed technical studies on piecewise projection models.

Our work differs from all of them, as we emphasize on the setting that all hyponyms and hypernyms in testing vocabulary are not seen in training.

3 Model Formulation

3.1 Piecewise Projection

Piecewise projection learning (Fu et al., 2014) serves as our baseline. The objective is to learn a relation transform from x to y on training pairs (x, y). Piecewise projection matrix Φ_k is learned separately for each cluster, after applying k-means clustering on the offset of training using $y - x$ between each pair.

$$\min_{\Phi_k} \frac{1}{|C_k|} \sum_{(x,y)\in C_k} \|x\Phi_k - y\|_2^2, \qquad (1)$$

where C_k represents the size of the k^{th} cluster. In addition, we also examine a simple solution of L2-penalized projection learning model which imposes a L2 constraint on Φ in Equation 1, i.e., $\alpha \|\Phi_k\|_2^2$.

3.2 Word Relation Autoencoder (WRAE)

Our model takes the form of an autoencoder. As shown in Equation 2,

$$\min_{\Phi_k} \frac{1}{|C_k|} \sum_{(x,y)\in C_k} \|x - x\Phi_k\Phi_k^*\|_2^2, \qquad (2)$$

where $x\Phi_k = y - x$. Here we adopt the simplifying trick (Kodirov et al., 2017a) to tie with the constraint (Ranzato et al., 2008) $\Phi^* = \Phi^T$. Note that the L2-norm regularization term is not necessary for WRAE to avoid overfitting since the constraint of $\Phi^* = \Phi^T$ guarantees $\|\Phi\|_2^2$ cannot be large otherwise the reconstruction loss will be bad. Also, the learning process is more efficient.

To release the constraint of $x\Phi_k = y - x$, the objective can be further split into two terms:

$$\min_{\Phi_k} \frac{1}{|C_k|} \sum_{(x,y)\in C_k} (\|x\Phi_k - (y-x)\|_2^2 + \lambda \left\|(y-x)\Phi_k^T - x\right\|_2^2), \tag{3}$$

where λ is a weighting constant.

We find that learning relation mapping from $x \to (y-x)$ instead of $x \to y$ effectively mitigates the pollution problem (Lazaridou et al., 2015). A prediction is said to be polluted if the nearest neighbor of predicted $\hat{y}$ matches a hypernym appeared in training set. The operation fundamentally solves the cause of pollution since each pair of input and output becomes $(x, y-x)$ instead of (x, y). Unlike projecting to a small number of target y, the target $y-x$ obviously differs from pair to pair thus avoiding simply overfitting the lexicons.

Conceptually, WRAE learns to extract the hyponym x from the relation vectors $r = y - x$ to optimize on the reconstruction loss. By encouraging the projection to learn the relationship between a word pair, WRAE effectively mitigates the mentioned generalized problem.

Our model is related to Semantic Autoencoder (SAE) (Kodirov et al., 2017b). With the latent relation directly associates with input x, WRAE can be regarded as a special conditional SAE where the condition is the input itself and is incorporated into the middle layer.

Relation	#Pair	#Head	#Tail
IsA	78073	21714	62455
AtLocation	39916	10100	11311
PartOf	14231	9784	5519
CreatedBy	503	385	414

Table 2: Relations from ConceptNet. For a relation pair $x \to y$, x is the *head* and y is the *tail*.

4 Experiments

4.1 Setup

Different from the experimental setup in previous works (Fu et al., 2014; Ustalov et al., 2017; Yamane et al., 2016) that do not assume the candidate hypernyms are unseen, in our experiments the vocabulary sets for training and testing are completely disjoint, i.e., all vocabularies in testing are not seen in training at all.

To further examine the generality of our model, we collect several hypernym-like relations listed in Table 2 from ConceptNet semantic graph. Considering the property of these relations, we treat the head and tail words of a pair as the x and y for our models similar to hyponym and hypernym, respectively. Examples are in Table 1.

We split the datasets with ratios 0.7, 0.2, and 0.1 for training, testing, and validation, respectively. For all results, we report the mean of 30 random splits. We test two different settings, one uses k-means clustering and one does not ($k = 1$). We tune the number of cluster k unsupervisedly with the Silhouette score (Rousseeuw, 1987) on validation. The projection matrices are optimized with the Adam method (Kingma and Ba, 2014) with learning rate = $1e^{-3}$. We adopt the GloVe (Pennington et al., 2014) 300d pre-trained word embeddings[1] which are trained on 6B token corpus (Wikipedia 2014 + Gigaword 5) with 400,000 words.

4.2 Evaluation Metrics

Hit Rate

To evaluate the precision of returned hypernyms, we follow Ustalov et al. (2017) and Kodirov et al. (2017a) using the hit rate measure (Frome et al., 2013). We also adopt area under curve (AUC) measure which computes the averaged area under the $l - 1$ trapezoids of hit@l to take the ranks of ground truth into consideration:

$$AUC_l = \frac{1}{2(l-1)} \sum_{i=1}^{l-1} (hit@i + hit@(i+1)), \tag{4}$$

Soft Pollution

To evaluate the degree of pollution of the extracted hypernym, we adopt a metric similar to Lazaridou et al. (2015). A prediction is said to be polluted if the nearest neighbor of predicted $\hat{y}$ matches a hypernym appears in the training set, noted as a binary function $pol_1(\hat{y})$.

However, it is possible that ground truth unseen hypernyms are be very close to some seen hypernyms in Y_{train} in real cases. We take ground truths

[1] https://nlp.stanford.edu/projects/glove/

Relation	Hit Rate			Pollution		Hit Rate			Pollution	
Model	hit_{10}	hit_{30}	AUC_{30}	pol_1	pol_{30}^{soft}	hit_{10}	hit_{30}	AUC_{30}	pol_1	pol_{30}^{soft}
IsA*	**k=1**					**k=25**				
Proj.	.110	.157	.107	.715	.223	.090	.137	.087	.721	.232
Proj.+L2	.112	.172	.108	.712	.223	.090	.139	.088	.719	.231
WRAE-Y	.120	.190	.110	.691	.220	.096	.146	.089	.720	.230
WRAE†	**.124**	**.194**	**.122**	**.602**	**.191**	**.164**	**.249**	**.160**	**.124**	**.034**
AtLocation	**k=1**					**k=25**				
Proj.	.075	.149	.103	.707	.115	.131	.220	.149	.782	.094
Proj.+L2	.083	.161	.110	.693	.113	.129	.222	.149	.796	.096
WRAE-Y	.086	.166	.127	.699	.112	.129	.236	.153	.782	.094
WRAE†	**.122**	**.224**	**.152**	**.409**	**.063**	**.148**	**.261**	**.174**	**.191**	**.024**
CreatedBy	**k=1**					**k=10**				
Proj.	.016	.040	.026	.625	.226	.054	.103	.059	.819	.591
Proj.+L2	.016	.044	.030	.624	.227	.050	.099	.048	.818	.592
WRAE-Y	.021	.060	.031	.586	.151	.057	.103	.052	.819	.591
WRAE†	**.070**	**.131**	**.095**	**.191**	**.067**	**.142**	**.243**	**.156**	**.071**	**.048**
PartOf	**k=1**					**k=45**				
Proj.	.335	.434	.341	.660	.187	.260	.405	.294	.796	.224
Proj.+L2	.340	.439	.344	.660	.188	.263	.407	.292	.793	.224
WRAE-Y	.342	.449	.350	.644	.182	.267	.411	.297	.793	.222
WRAE†	**.355**	**.464**	**.370**	**.546**	**.149**	**.437**	**.539**	**.440**	**.117**	**.038**

Table 3: Performance on ConceptNet relation dataset. †: all results pass the hypothesis test against the other models with $p < 0.01$. *: for IsA, we report hit@5 and hit@10. For hit rates, the higher the better. For pollution, the lower the better.

into consideration:

$$pol_l^{soft}(\hat{y}, y) = \rho \cdot pol_1(\hat{y}), y \in Y_{test},$$
$$\rho = \begin{cases} 1, & if \quad NN_l(y) \cap Y_{train} = \phi, \\ 2^{\frac{n-1}{l-1}} - 1, & otherwise, \end{cases} \tag{5}$$

where n is for the top n nearest neighbors (from 1 to l) of y that appears in Y_{train} and ρ is a factor term exponentially decreases from 1 to 0 along with the increase of n therefore provides a smoother estimation. With pollution indications, one can understand to what degree the model suffers from overfitting on the seen examples. ϕ is the empty set. Note that it is reasonable to set l equal for both hit rate and soft pollution.

4.3 Results: Unseen General Hypernym-Like Relation Extraction

We report two sets of results for all models, one with clustering and one without (k = 1). As shown in Table 3, WRAE outperforms the competitors significantly with and without clustering. The naive application of Equation 2 which set $x\Phi_k = y$, denoted as *WRAE-Y*, consistently ranks second. The $y - x$ operation is crucial to avoid pollution thus guarantees the generalization power of the mapping. Apply simple L2-norm regularization Equation 1 for *Proj.*, denoted as *Proj.+L2*, only slightly improves the performance. The results in Table 3 supports our hypothesis that Proj. models deteriorate significantly for larger k, due to lack of training examples for hypernyms in each cluster. We prove that WRAE is effective against pollution. The role of regularizer is important for decoders to optimize towards better objective.

The negative effects derived from pollution impact accuracy. We observe severe pollution problem in simple projection learning. Take *IsA* as example, in $k = 1$ group the pol_1 is about 71% for Proj., which implies about two of out of three returned predictions are data points from training data. Our WRAE reduces the pollution pol_1 to 60% and 12% after clustering. The improvement on accuracy supports that pollution indication reflects the inherent overfitting problem. In general, results are consistent with our claims that pollution can be viewed as valid negative indicators.

Across the board, the performance should benefit from domain clustering if pollution is handled properly as the experiments showed.

5 Conclusion

We present an unseen hypernym extraction framework and analyze the pollution problem with this setup. Consequently we argue that only by using unseen candidates in evaluation can truly test whether the model learns the true relation representation, instead of being polluted by the seen training examples. Future work includes relation discovery, which is to identify new relations besides hypernyms in an unsupervised manner.

Acknowledgments

This material is based upon work supported by Taiwan Ministry of Science and Technology (MOST) under grant number 106-2218-E-002 - 042 (machine learning related).

References

Luis Espinosa-Anke, José Camacho Collados, Claudio Delli Bovi, and Horacio Saggion. 2016. Supervised Distributional Hypernym Discovery via Domain Adaptation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP 2016)*, pages 424–435.

C Fellbaum. 1998. *WordNet: An Electronic Lexical Database: Bradford Book*. Cambridge, MA: MIT Press.

Andrea Frome, Gs Corrado, and Jonathon Shlens. 2013. Devise: A deep visual-semantic embedding model. *Advances in neural information processing systems*, pages 1–11.

Ruiji Fu, Jiang Guo, Bing Qin, Wanxiang Che, Haifeng Wang, and Ting Liu. 2014. Learning semantic hierarchies via word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1199–1209. Baltimore, Maryland.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A Method for Stochastic Optimization. *arXiv:1412.6980 [cs.LG]*.

Elyor Kodirov, Tao Xiang, and Shaogang Gong. 2017a. Semantic Autoencoder for Zero-Shot Learning. In *Computer Vision and Pattern Recognition*. Computer Vision Foundation.

Elyor Kodirov, Tao Xiang, and Shaogang Gong. 2017b. Semantic autoencoder for zero-shot learning.

Angeliki Lazaridou, Georgiana Dinu, and Marco Baroni. 2015. Hubness and Pollution: Delving into Cross-Space Mapping for Zero-Shot Learning. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pages 270–280. Association for Computational Linguistics.

Alessandro Lenci and Giulia Benotto. 2012. Identifying hypernyms in distributional semantic spaces. In *Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 75–79, Stroudsburg, PA, USA. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Neha Nayak. 2015. Learning hypernymy over word embeddings. Technical report, Stanford Univ.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global Vectors for Word Representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 1532–1543.

Marc'aurelio Ranzato, Y lan Boureau, and Yann Lecun. 2008. Sparse Feature Learning for Deep Belief Networks. In *Advances in Neural Information Processing Systems 20 (NIPS 2007)*, pages 1185–1192. Curran Associates, Inc.

Peter J Rousseeuw. 1987. Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, 20:53–65.

Robert Speer and Catherine Havasi. 2012. Representing General Relational Knowledge in ConceptNet 5. In *International conference on language resources and evaluation (LREC)*, pages 3679–3686.

Dmitry Ustalov, Nikolay Arefyev, Chris Biemann, and Alexander Panchenko. 2017. Negative Sampling Improves Hypernymy Extraction Based on Projection Learning. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics (EACL 2017)*, pages 543–550, Valencia, Spain. Association for Computational Linguistics.

Ekaterina Vylomova, Laura Rimell, Trevor Cohn, and Timothy Baldwin. 2016. Take and took, gaggle and goose, book and read: Evaluating the utility of vector differences for lexical relation learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL 2016)*, pages 1671–1682, Berlin, Germany. Association for Computational Linguistics.

Julie Weeds, Daoud Clarke, Jeremy Reffin, David Weir, and Bill Keller. 2014. Learning to Distinguish Hypernyms and Co-Hyponyms. In *Proceedings of the 25th International Conference on Computational Linguistics (COLING 2014)*, pages 2249–2259, Dublin, Ireland. Dublin City University and Association for Computational Linguistics.

Josuke Yamane, Tomoya Takatani, Hitoshi Yamada, Makoto Miwa, and Yutaka Sasaki. 2016. Distributional hypernym generation by jointly learning clusters and projections. In *Proceedings of 26th International Conference on Computational Linguistics (COLING 2016)*, pages 1871–1879, Osaka, Japan. The COLING 2016 Organizing Committee.