

Evidential Relational-Graph Convolutional Networks for Entity Classification in Knowledge Graphs

Tobias Weller

Data and Web Science Group, University of Mannheim
Mannheim, Germany
tobi@informatik.uni-mannheim.de

Heiko Paulheim

Data and Web Science Group, University of Mannheim
Mannheim, Germany
heiko@informatik.uni-mannheim.de

ABSTRACT

Despite the vast amount of information encoded in knowledge graphs, they often remain incomplete. Neural networks, in particular Graph Convolutional Neural Networks, have been shown to be effective predictors to complete information about the class affiliation of entities in knowledge graphs. However, these models remain ignorant to their predictions confidence due to their used point estimate of a softmax output. In this paper, we combine Graph Convolutional Neural Networks with recent developments in the field of Evidential Learning by placing a Dirichlet distribution on the class probabilities to overcome this problem. We use the continuous output of a Graph Convolutional Neural Network as parameters for a Dirichlet distribution. In this way, the predictions of the model are represented as a distribution over possible softmax outputs, rather than a point estimate of a softmax output. The experiments show that a better performance in predicting class affiliations can be achieved compared to recent models. In addition, the experiments show that this approach overcomes the well-known problem of overconfident prediction of deterministic neural networks.

CCS CONCEPTS

• **Information systems** → *Information systems applications; Data mining; Information retrieval; Decision support systems*; • **Computing methodologies** → *Machine learning*.

KEYWORDS

Entity Classification, Graph Convolutional Neural Network, Evidential Learning, Knowledge Graph

ACM Reference Format:

Tobias Weller and Heiko Paulheim. 2021. Evidential Relational-Graph Convolutional Networks for Entity Classification in Knowledge Graphs. In *Proceedings of the 30th ACM International Conference on Information and Knowledge Management (CIKM '21), November 1–5, 2021, Virtual Event, QLD, Australia*. ACM, New York, NY, USA, 5 pages. <https://doi.org/10.1145/3459637.3482102>

1 INTRODUCTION

Knowledge Graphs (KGs) encode factual knowledge in the form of triples (subject-relation-object pairs), which is used in various applications, including Question Answering [22] and information

retrieval [20, 21] to enhance the performance of the systems. Despite the enormous effort made to keep the knowledge encoded in the KGs up-to-date and consistent [5], the KGs are often incomplete. Due to the huge amount of information, manual maintenance to complete missing factual knowledge is not feasible [11]. In particular, encoding factual knowledge about the class affiliation of entities is of great importance for automatic reasoning and inference of information from the knowledge graph.

In recent years, Graph Convolutional Neural Networks (GCN) [6–8, 16] have been used in particular to predict the missing class affiliations of entities. Other methods that use distance-based scoring functions, such as TransE [2], learn embeddings for entities, but unlike GCN, these are no end-to-end models, but a second model must be learned subsequently to predict missing class affiliations. Further methods like RDF2Vec [14, 15] or Weisfeiler-Lehman Graph Kernels (WL) [3, 19] are neither end-to-end models, but use graph-specific features to predict missing class affiliations.

Due to the superior performance of GCN models in predicting class affiliations of entities resulting from the end-to-end approach, these models have been widely used in recent years. However, these deterministic models¹ are often overconfident with their predictions and ignorant to their predictions confidence due to the use of a point estimate of a softmax output. We therefore address these two issues and combine GCN with recent developments in the field of evidential learning [1, 17, 18]. In this work, we propose Evidential Graph Convolutional Neural Networks (*E-R-GCNs*), which can be considered as an extension to previous R-GCNs [16]. Instead of using a softmax output layer and thus a single point estimate, we use a conjugate prior distribution to represent the predictions of the model as a distribution over possible softmax outputs. Given that the prediction of class affiliations of entities is considered as a discrete multinomial distribution, we use the Dirichlet distribution as the conjugate prior distribution accordingly.

In the next section, we will introduce the model (Section 2), followed by conducted experiments (Section 3), highlighting in particular the improved performance compared to the previous R-GCN model, as well as showing that *E-R-GCN* does not suffer from being overconfident in predictions compared to the current R-GCN model. We will with an outline of the benefits of this modeling in real-world use cases and an outlook on future work.

2 APPROACH

The approach of the introduced model for entity classification (*E-R-GCN*) is shown in Figure 1 and combines methods of GCN [16] and quantifying uncertainty in neural networks [18]. This figure



This work is licensed under a Creative Commons Attribution-ShareAlike International 4.0 License.

CIKM '21, November 1–5, 2021, Virtual Event, QLD, Australia

© 2021 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-8446-9/21/11.

<https://doi.org/10.1145/3459637.3482102>

¹Once Neural Networks are trained they are deterministic.

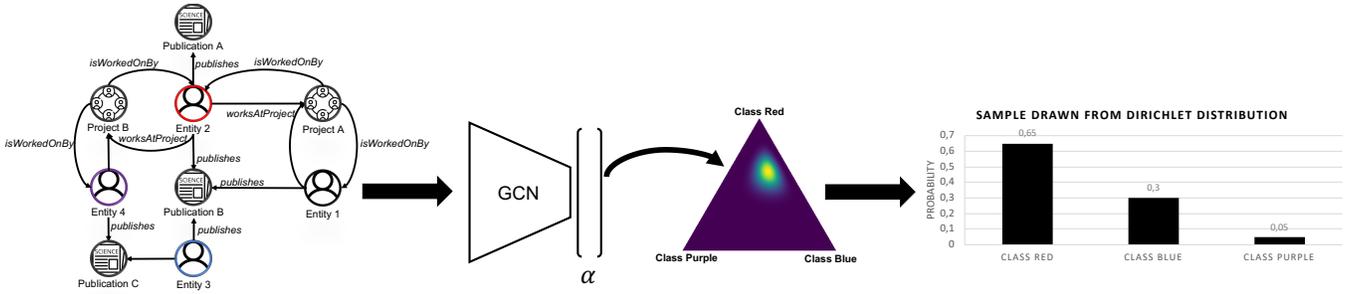


Figure 1: Approach for entity classification using E-R-GCN. The E-R-GCN predicts the parameters α of a Dirichlet distribution given the KG and an entity (e.g. Entity 1). Based on the Dirichlet distribution, categorical distributions can be drawn, representing the class assignment for the given input.

shows a subgraph of the dataset AIFB [13] in which various entities have written different publications and are working in projects. The class affiliation of the entities are encoded in colors. In general, a softmax function is typically used in the output layer of GCNs for classification tasks, but even with a high softmax output for a particular class, the model can be uncertain in its predictions and it is a point estimator. Therefore, the idea is to predict the parameters of a Dirichlet distribution from which multiple categorical distributions can be drawn. A Dirichlet distribution is used as it is the conjugate prior of a categorical distribution and a multiclass classification problem is characterized by estimating the categorical distributions of the classes. Depending on the predictions of the parameters of the Dirichlet distribution, the concentration of the drawn distributions can be on one class or, if there is a larger uncertainty, it can be spread over several classes.

In order to predict the Dirichlet parameters, we use a ReLU function instead of a softmax function in the output layer of a GCN. Since the Dirichlet distribution is characterized by the parameters $\alpha^K \in R^+$, where K denotes the number of classes² and $K > 2$ always holds, we denote the output of the GCN as evidence (e) and apply the following formula to ensure that $\alpha^K \in R^+$ holds true.

$$\tilde{\alpha}_{ik} = e_{ik} + 1 \quad (1)$$

Following previous work in the field of evidential learning, we use a loss function that minimizes the prediction error and the variance of the Dirichlet distribution, as well as a regularization term that penalizes the predictive distribution, which do not contribute to data fit. Given that we intend to design and train a GCN to form multinomial opinions for the entity classification of a given sample i as a Dirichlet distribution where \hat{p}_i is the prediction, the expected probability for the k -th output of the sample results from the mean of the respective Dirichlet distribution, where $S = \sum_{i=1}^K \tilde{\alpha}_i$ represents the Dirichlet strength.

$$\hat{p}_{ik} = \frac{\tilde{\alpha}_{ik}}{S_i} \quad (2)$$

Kullback-Leibler (KL) divergence is incorporated into the loss in order to ensure that the total evidence decreases to zero, even for samples that cannot be correctly classified. Hereby we follow

²In Figure 1 is $K = 3$, since three different classes exist in the KG, encoded in colors (red, blue and purple). No class affiliation is given for Entity 1 as we want to predict the class for this entity.

the recommendation of Sensoy et al. [17], which has been shown in empirical experiments to provide the most stable performance with respect to classification. We denote N as the total number of samples and y_i as the actual one-hot encoded class affiliation for sample i . The loss of our approach is then given by the following formula, where $\lambda_t = \min(1.0, \frac{t}{10}) \in [0, 1]$ is the annealing factor with timestep t , $\Gamma(\cdot)$ is the gamma function, and $\psi(\cdot)$ is the digamma function.

$$L(\Theta) = \sum_{i=1}^N \sum_{j=1}^K (y_{ij} - \hat{p}_{ij})^2 + \frac{\hat{p}_{ij}(1 - \hat{p}_{ij})}{(S_i + 1)} \quad + \quad (3)$$

Minimizing prediction error and variance of the Dirichlet

$$\lambda_t \sum_{i=1}^N \log \left(\frac{\Gamma(\sum_{k=1}^K \tilde{\alpha}_{ik})}{\Gamma(K) \prod_{k=1}^K \Gamma(\tilde{\alpha}_{ik})} \right) + \sum_{k=1}^K (\tilde{\alpha}_{ik} - 1) \left[\psi(\tilde{\alpha}_{ik}) - \psi \left(\sum_{j=1}^K \tilde{\alpha}_{ij} \right) \right]$$

KL divergence to penalize divergences that do not contribute to data fit

In order to specify the uncertainty and thus detect out-of-distribution samples in the predictions using a conjugate prior, we use the Dempster-Shafer Theory of Evidence (DST) [4]. A belief mass is defined, which corresponds to a Dirichlet distribution from which the uncertainty can be derived. The uncertainty, denoted as u , is defined as follows:

$$u = \frac{K}{S} \quad (4)$$

3 EXPERIMENTS

Datasets. Following related work [13, 16], we use the well-known datasets AIFB, MUTAG, BGS, and AM for evaluation. The datasets are modeled as graphs in the form of triples (subject-relation-object pairs). The class affiliation of the entities is modeled as a triple as well. For the dataset AIFB the class affiliation is modeled by the relations *employs* and *affiliation*, MUTAG by *hasLithogenesis*, BGS as *objectCategory* and for AM as *material*. These relations are removed beforehand in order not to affect the learning process. We use predefined train/test set splits for training and testing. Statistics on the datasets used are shown in Table 1 and more information about the datasets can be obtained in the originally mentioned paper [13]. **Metrics.** Following previous work [14–16], we use accuracy to measure the performance of the models. We run the experiments ten times and report the average performance on the existing test splits.

Table 1: Datasets used in the experiments along with the number of entities, relations, edges, labels and classes. *Labeled* denotes the subset of entities that have labels and that are to be classified, meaning the number of test instances.

Dataset	AIFB	MUTAG	BGS	AM
Entities	8,285	23,644	333,845	1,666,764
Relations	45	23	103	133
Edges	29,043	74,227	916,199	5,988,321
Labeled	176	340	146	1,000
Classes	4	2	2	11

Baselines. We compare our approach, with recent state-of-the-art models in entity classification. The baseline includes strong models as R-GCN [16], RDF2Vec [14, 15], Feat [12] and WL [3, 19]. R-GCN uses an end-to-end relational graph convolutional network and a cross-entropy loss for entity classification. RDF2Vec extracts random walks, which are then used in a skipgram model [9, 10] to learn embeddings. These embeddings are used for the subsequent classification. WL uses graph kernels that count substructures in graphs and Feat uses hand-designed feature extractors³.

3.1 Entity Classification

The results of the experiments on the test benchmark for entity classification is reported in Table 2. Based on the existing work on R-GCN [16], we used the same hyperparameter settings for these experiments as well, since these have been identified as the best hyperparameter choice based on a validation set. Accordingly, for R-GCN and *E-R-GCN* we trained a 2-layer model with 16 hidden units (10 for AM), a weight decay of $0.5e-3$, a learning rate of 0.01, and using Adam 50 epochs. Following experiments of R-GCN [16] and RDF2Vec [14], we used a linear SVM for classifying the entities in the RDF2Vec and WL approach.

Considering the results for entity classification in Table 2 we observe on the one hand that our model *E-R-GCN* achieves state-of-the-art results on AIFB and AM, outperforming R-GCN in particular, and on the other hand that the results reported in R-GCN [16] are confirmed. In both datasets AIFB and AM, which are originally modeled as RDF, *E-R-GCN* can outperform the baselines. Using a Dirichlet distribution on class probabilities and assigning a R-GCN to its parameters achieves a higher performance on the test dataset and, moreover, makes no overconfident predictions. We will discuss these aspects in more detail in Section 3.2.

Compared to the AIFB and AM datasets, *E-R-GCN* can outperform R-GCN on the MUTAG and BGS datasets, but not WL on MUTAG and RDF2Vec on BGS. The reason is that the underlying model of *E-R-GCN* is the same as R-GCN and therefore contains similar characteristics. Both MUTAG and BGS are not originally RDF graphs, but datasets which were converted into RDF. MUTAG represents a chemical compound, where either atomic bonds or the presence of a certain feature is indicated by the relations used. Similarly, BGS encodes information about various rocks using hierarchical feature descriptions, where either the presence of a certain feature or feature hierarchy is indicated by the relations used. As

³<https://github.com/TobiWeller/E-R-GCN>

Table 2: Averaged entity classification results in accuracy over ten runs on the test split.

Model	AIFB	MUTAG	BGS	AM
Feat	55.55	77.94	72.41	66.66
WL	80.55	80.88	86.20	87.37
RDF2Vec	88.88	67.20	87.24	88.33
R-GCN	92.22	73.97	75.86	89.14
<i>E-R-GCN</i>	95.56	74.56	76.55	89.85

a result of this modeling, the labeled entities in MUTAG and BGS are connected only by high-degree hub nodes in which a particular feature is present for more precise determination of the label. Due to the accumulation of messages from neighboring nodes in R-GCN and *E-R-GCN*, these features are not allocated sufficient value, resulting in them being neglected at high-degree nodes.

3.2 Predictive Uncertainty Performance

In order to first understand the output and the predicted uncertainty, we visualize the Dirichlet distribution using the predicted α parameters. For this purpose, we use the AIFB dataset and only three of the four classes, since the Dirichlet distribution is usually visualized as a triangle. Figure 2 shows the predicted α parameter for three entities of the AIFB dataset. In the first case, we observe a clear assignment of the entity *id57instance* to the class affiliation4 ($\alpha = (1, 1, 77.37877)$, $u = 0.051$). In the second example, the entity *id2096instance*, the predicted alpha parameters indicate a definite assignment to the class affiliation2 ($\alpha = (31.56, 4.35, 1.00)$, $u = 0.10$). The uncertainty is slightly increased, yet low that the model does not consider this entity to be an out-of-distribution sample, not following the training data. In the case of the third entity, *id2094instance*, the assignment between the classes affiliation2 and affiliation3 is not definite ($\alpha = (6.50, 8.74, 1.52)$, $u = 0.20$). The entity *id2094instance* belongs to the affiliation2, but due to the predicted alpha parameters and the Dirichlet distribution, an assignment to the class affiliation3 is most likely predicted. This indecision between the two classes can be visualized very well with the Dirichlet distribution. The uncertainty is slightly increased, which is due to the indecisiveness of the two class affiliations. Nevertheless, the uncertainty is not too high so that it cannot be assumed that this entity does not follow the distribution of the other entities of the training set, i.e. it is an outlier. The higher uncertainty is due to

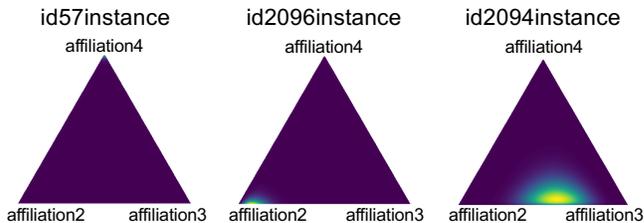


Figure 2: Visualizations of Dirichlet parameters predicted by our method to different entities in the test split.

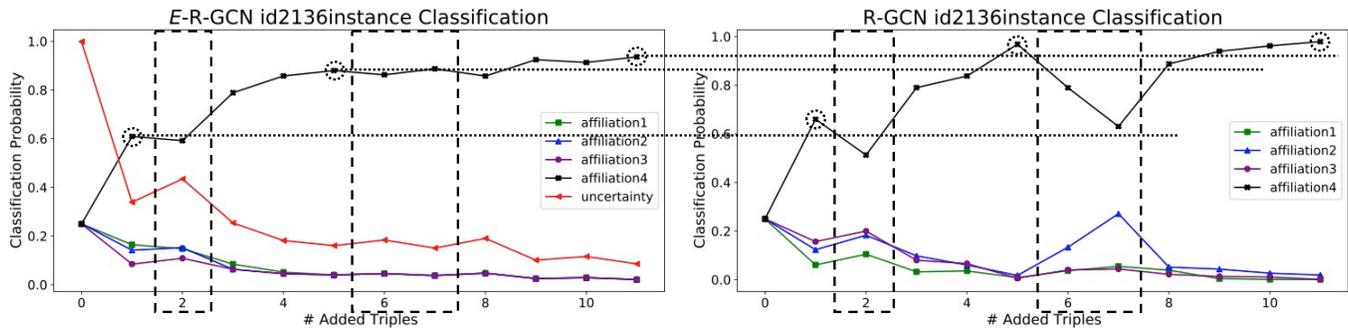


Figure 3: Comparison of predictions between *E-R-GCN* and *R-GCN* with increasing number of triples. While *R-GCN* suffers much more on collapses in its probability assignment to the class affiliation4 (right panel), these fluctuations are not equally pronounced in *E-R-GCN* (left panel).

relationships of common papers with entities of affiliation2, not due to unknown distribution.

To demonstrate validate the robustness of the evidential model and the advantage of modeling uncertainty compared to using a standard softmax output of deep neural networks, we show to what extent the classification probability and uncertainty using the proposed method change compared to existing models that use a softmax function as classification probability. For this we use a similar approach as related papers [17]. We train the models on the training dataset and apply it on an unseen subgraph, containing the entity *id2136instance* from the test split. We first remove all eleven incoming and outgoing relations of this entity and apply the models to the resulting subgraph. We then gradually randomly add the eleven relations back to the subgraph one by one and apply the models to each, measuring performance and uncertainty. This experimental setup will show how the models perform when applying them on subgraphs with different amounts of information, and how measuring the uncertainty can help to decide whether applying the model on this unknown subgraph is reasonable.

Figure 3 shows the results of this experiment. We compare *E-R-GCN* with *R-GCN*, since both use the same model but with different loss functions and are therefore most suitable for comparison. For both Models, in the absence of any information about the entity (i.e. no triples to describe the entity), the probabilities of the assignments are equal to 0.25 across all classes. With increasing number of triples and thus information about the entity, the probability of the assignment to the class affiliation4, which is the correct class, increases. At the same time, the uncertainty of the *E-R-GCN* is 1.0 when no information about the entity is given (i.e., it is an out-of-distribution sample w.r.t. the training set), and drops successively when adding information about the entity.

Considering Figure 3, *R-GCN* is generally overconfident in its predictions. For better visualization, the predictions of both models have been marked at the locations 1, 5, and 11. *R-GCN* assigns significantly higher probabilities to the class affiliation4, especially at position 1, when the entity is described by only one triple, the probability is already significantly higher than that of *E-R-GCN*. Due to this overestimation there are stronger drops in the probability assignment to the class affiliation4, in particular at position 2 and in [5, 8], when adding further triples about the entity. In

contrast, the probability assignments to the classes in the model *E-R-GCN* are much more robust, so that the fluctuations are not as significant. This moderate probability progression leads to much more trustworthy and comprehensible predictions, especially when considering an online learning approach. To validate the model, we also applied it on a completely different knowledge graph for which it was not trained, i.e., a subgraph of DBpedia to predict the class affiliation of the entity Barack Obama⁴. The uncertainty was 1.0, as expected, since the model was never trained on this graph. However, based on the specification of this uncertainty of the model, this can be used as an indicator of whether or not the model at hand is suitable for application on this graph⁵.

4 CONCLUSIONS

In this work, we have shown how to combine Relational Graph Convolutional Neural Networks with recent research in the field of Evidential Learning for entity classification. Instead of using a softmax function as an output layer like in previous *R-GCN* models, we have designed the model by placing a Dirichlet distribution on the class probabilities. The experimental results show a performance increase compared to the previous *R-GCN* model. The use of the evidential learning approach makes no overconfident predictions and allows to specify uncertainties of the model and thus to accurately determine whether a model is suitable for prediction on a subgraph, which is currently not considered in classical ML approaches. This is particularly important when considering dynamic changes in knowledge graphs or to detect out-of-distribution samples. We would like to pursue this approach further in the future, both applying the learning approach to extensions to *R-GCN*, e.g., with attention layers, and modifying the model to allow knowledge graph completions in dynamic knowledge graphs. Furthermore, we would like to apply this approach within an active learning framework, using the uncertainty metric for determining which data points should be labeled.

ACKNOWLEDGMENTS

This work is supported by the European Commission in the Horizon 2020 project *Teaming.AI* under grant number 957402.

⁴https://dbpedia.org/resource/Barack_Obama

⁵In that case, it indicates that is not

REFERENCES

- [1] Alexander Amini, Wilko Schwarting, Ava Soleimany, and Daniela Rus. 2020. Deep Evidential Regression. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 14927–14937. <https://proceedings.neurips.cc/paper/2020/file/aab085461de182608ee9f607f3f7d18f-Paper.pdf>
- [2] Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems*, C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K. Q. Weinberger (Eds.), Vol. 26. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2013/file/1cecc7a77928ca8133fa24680a88d2f9-Paper.pdf>
- [3] Gerben Klaas Dirk de Vries and Steven de Rooij. 2015. Substructure counting graph kernels for machine learning from RDF data. *Journal of Web Semantics* 35 (2015), 71–84. <https://doi.org/10.1016/j.websem.2015.08.002> Machine Learning and Data Mining for the Semantic Web (MLDMSW).
- [4] Arthur P. Dempster. 2008. *A Generalization of Bayesian Inference*. Springer Berlin Heidelberg, Berlin, Heidelberg, 73–104. https://doi.org/10.1007/978-3-540-44792-4_4
- [5] Nicolas Heist, Sven Hertling, Daniel Ringler, and Heiko Paulheim. 2020. Knowledge Graphs on the Web—An Overview. In *Knowledge Graphs for eXplainable Artificial Intelligence: Foundations, Applications and Challenges*. IOS Press, 3–22.
- [6] Hailong Jin, Lei Hou, Juanzi Li, and Tiansi Dong. 2019. Fine-Grained Entity Typing via Hierarchical Multi Graph Convolutional Networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 4969–4978. <https://doi.org/10.18653/v1/D19-1502>
- [7] Thomas N Kipf and Max Welling. 2016. Semi-Supervised Classification with Graph Convolutional Networks. *arXiv preprint arXiv:1609.02907* (2016).
- [8] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard S. Zemel. 2016. Gated Graph Sequence Neural Networks. In *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). <http://arxiv.org/abs/1511.05493>
- [9] Tomáš Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient Estimation of Word Representations in Vector Space. In *1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.). <http://arxiv.org/abs/1301.3781>
- [10] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed Representations of Words and Phrases and Their Compositionality. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2 (Lake Tahoe, Nevada) (NIPS'13)*. Curran Associates Inc., Red Hook, NY, USA, 3111–3119.
- [11] Heiko Paulheim. 2017. Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic web* 8, 3 (2017), 489–508.
- [12] Heiko Paulheim and Johannes Fümkrantz. 2012. Unsupervised Generation of Data Mining Features from Linked Open Data. In *Proceedings of the 2nd International Conference on Web Intelligence, Mining and Semantics (Craiova, Romania) (WIMS '12)*. Association for Computing Machinery, New York, NY, USA, Article 31, 12 pages. <https://doi.org/10.1145/2254129.2254168>
- [13] Petar Ristoski, Gerben Klaas Dirk de Vries, and Heiko Paulheim. 2016. A Collection of Benchmark Datasets for Systematic Evaluations of Machine Learning on the Semantic Web. In *The Semantic Web – ISWC 2016*, Paul Groth, Elena Simperl, Alasdair Gray, Marta Sabou, Markus Krötzsch, Freddy Lecue, Fabian Flöck, and Yolanda Gil (Eds.). Springer International Publishing, Cham, 186–194.
- [14] Petar Ristoski and Heiko Paulheim. 2016. RDF2Vec: RDF Graph Embeddings for Data Mining. In *The Semantic Web – ISWC 2016*, Paul Groth, Elena Simperl, Alasdair Gray, Marta Sabou, Markus Krötzsch, Freddy Lecue, Fabian Flöck, and Yolanda Gil (Eds.). Springer International Publishing, Cham, 498–514.
- [15] Petar Ristoski, Jessica Rosati, Tommaso Di Noia, Renato De Leone, and Heiko Paulheim. 2019. RDF2Vec: RDF graph embeddings and their applications. *Semantic Web* 10, 4 (2019), 721–752. <https://doi.org/10.3233/SW-180317>
- [16] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling Relational Data with Graph Convolutional Networks. In *The Semantic Web*, Aldo Gangemi, Roberto Navigli, Maria-Esther Vidal, Pascal Hitzler, Raphaël Troncy, Laura Hollink, Anna Tordai, and Mehwish Alam (Eds.). Springer International Publishing, Cham, 593–607.
- [17] Murat Sensoy, Lance Kaplan, and Melih Kandemir. 2018. Evidential Deep Learning to Quantify Classification Uncertainty. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2018/file/a981f2b708044d6fb4a71a1463242520-Paper.pdf>
- [18] Murat Sensoy, Lance Kaplan, and Melih Kandemir. 2018. Evidential Deep Learning to Quantify Classification Uncertainty. In *Advances in Neural Information Processing Systems*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Eds.), Vol. 31. Curran Associates, Inc. <https://proceedings.neurips.cc/paper/2018/file/a981f2b708044d6fb4a71a1463242520-Paper.pdf>
- [19] Nino Shervashidze, Pascal Schweitzer, Erik Jan van Leeuwen, Kurt Mehlhorn, and Karsten M. Borgwardt. 2011. Weisfeiler-Lehman Graph Kernels. *Journal of Machine Learning Research* 12, 77 (2011), 2539–2561. <http://jmlr.org/papers/v12/shervashidze11a.html>
- [20] Chenyan Xiong and Jamie Callan. 2015. EsdRank: Connecting Query and Documents through External Semi-Structured Data. In *Proceedings of the 24th ACM International Conference on Information and Knowledge Management (Melbourne, Australia) (CIKM '15)*. Association for Computing Machinery, New York, NY, USA, 951–960. <https://doi.org/10.1145/2806416.2806456>
- [21] Chenyan Xiong, Jamie Callan, and Tie-Yan Liu. 2017. Word-Entity Duet Representations for Document Ranking. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (Shinjuku, Tokyo, Japan) (SIGIR '17)*. Association for Computing Machinery, New York, NY, USA, 763–772. <https://doi.org/10.1145/3077136.3080768>
- [22] Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering. In *North American Chapter of the Association for Computational Linguistics (NAACL)*.