An Open Challenge for Inductive Link Prediction on Knowledge Graphs

Mikhail Galkin
Mila & McGill University
Montreal, QC, Canada
mikhail.galkin@mila.quebec

Max Berrendorf
LMU Munich
München, Germany
berrendorf@dbs.ifi.lmu.de

Charles Tapley Hoyt
Harvard Medical School
Boston, MA, USA
cthoyt@gmail.com

ABSTRACT

An emerging trend in representation learning over knowledge graphs (KGs) moves beyond transductive link prediction tasks over a fixed set of known entities in favor of inductive tasks that imply training on one graph and performing inference over a new graph with unseen entities. In inductive setups, node features are often not available and training shallow entity embedding matrices is meaningless as they cannot be used at inference time with unseen entities. Despite the growing interest, there are not enough benchmarks for evaluating inductive representation learning methods. In this work, we introduce *ILPC 2022*, a novel open challenge on KG inductive link prediction.

To this end, we constructed two new datasets based on Wikidata with various sizes of training and inference graphs that are much larger than existing inductive benchmarks. We also provide two strong baselines leveraging recently proposed inductive methods. We hope this challenge helps to streamline community efforts in the inductive graph representation learning area. *ILPC 2022* follows best practices on evaluation fairness and reproducibility, and is available at https://github.com/pykeen/ilpc2022.

CCS CONCEPTS

• **Computing methodologies → Neural networks**; **Knowledge representation and reasoning**.

KEYWORDS

Knowledge graph, dataset, neural networks, link prediction

1 INTRODUCTION

The transductive link prediction (LP) task over knowledge graphs (KGs) has dominated the evaluation of knowledge graph embedding models (KGEMs) since their first appearance in 2011 [17]. Because the same graph is used for training and inference in the transductive
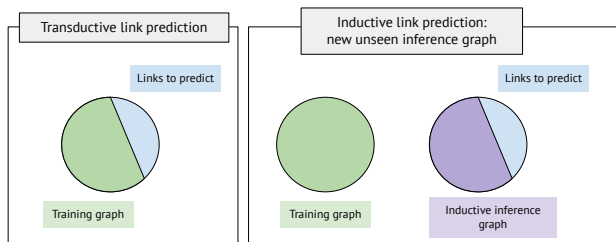
Figure 1: Transductive vs. inductive link prediction

setting, a majority of KGEMs use the *shallow embedding* paradigm in which a learnable (or infrequently, a pre-computed) vector is assigned to each entity seen during training. Most large benchmarking studies [2, 15, 18] focus solely on the transductive LP task following the shallow embedding paradigm.

In the inductive setting, a different graph can be used for inference that either contains a combination of seen and unseen entities (i.e., semi-inductive) or only unseen entities (i.e., fully inductive) and either the same or a subset of relations from the training graph [1]. During evaluation, there are two scoring tasks between entities seen and unseen during training: *seen ↔ unseen* and *unseen ↔ unseen*. In this work, we focus on the fully inductive setting where node features are not given and links to be predicted fall into the *unseen ↔ unseen* category (Figure 1). Because the presence of unseen entities in the inference graph is fundamentally incompatible with the shallow embedding paradigm, several non-shallow methods compatible with the inductive setting have been recently proposed [12, 21, 27], many of which leverage recent advancements in graph neural networks and geometric deep learning [5].

In this work, we outline the open Inductive Link Prediction Challenge (*ILPC 2022*) which aims to support community efforts to create more efficient and performant inductive link prediction models for KGs. With it, we introduce two novel inductive link prediction datasets of different sizes leveraging Wikidata [24] along with two strong baselines employing the recently proposed NodePiece [12] representation learning method.

2 RELATED WORK

There are far fewer high quality inductive link prediction benchmarks than transductive link prediction benchmarks. For example, the Open Graph Benchmark [15] contains two datasets for transductive link prediction, `WikiKG2` and `WikiKG90Mv2` [14], but none for inductive link prediction. `Wikidata5M` [25] and inductive splits published in [10] demonstrate encoding entities by applying a language model to their respective descriptions. They evaluate inductive LP

performance only on the single triple level without an inference graph. Therefore, those tasks can be seen as a probe of textual encoders and richness of their features rather than evaluation of graph representation learning capabilities.

To the best of our knowledge, the only notable inductive LP benchmarks comprise inductive versions of three popular transductive LP datasets, FB15k-237 [22], WN18RR [11], and NELL-995 [8] proposed in in [21]. Each KG accommodates four splits of various training and inference graph sizes making overall 12 distinct datasets. Most of those datasets are rather small where the largest one has 27K training triples and 12K in the inference graph. While the modest sizes were originally motivated by the computational complexity of the proposed method, some aspects have been resolved in subsequent works [12, 27].

While our tasks draw inspiration from [21], the new *ILPC 2022* datasets are different in several ways: 1) the graphs are excerpts from Wikidata, the largest publicly available KG actively maintained by a large community with numerous practical applications [6, 7, 9], hence, new results are more likely to be apprehended by a wider audience; 2) our largest graph is almost an order of magnitude larger than that of [21] and its inference graph size is challenging for modern GNN methods with inductive capabilities; 3) we make sure training and inference graphs are connected components without dangling nodes; 4) our evaluation criteria assumes ranking over all nodes in the inference graph instead of 50 random samples as done in [21]. We describe the dataset construction process and design decisions in more detail in Section 4.

3 TASK DEFINITION

Let $\mathbb{T}(\mathcal{E}, \mathcal{R}) := \{(h, r, t) \mid h, t \in \mathcal{E}, r \in \mathcal{R}\}$ denote the set of possible triples between entities from $\mathcal{E}$ and relations from $\mathcal{R}$. Let $\mathcal{E}_\bullet$ denote the entities occurring in training and $\mathcal{E}_\circ$ the entities in the inference graph. Then, since we are in the fully inductive setting, we have a set of training triples $\mathcal{T}_{train} \subseteq \mathbb{T}(\mathcal{E}_\bullet, \mathcal{R})$ (comprising the *training graph*), and a set of inference, validation and test triples $\mathcal{T}_{inf}, \mathcal{T}_{test}, \mathcal{T}_{validation} \subseteq \mathbb{T}(\mathcal{E}_\circ, \mathcal{R})$ where $\mathcal{T}_{inf}$ represents the *inference graph*. Note that sets of training and inference entities are disjoint, i.e., $\mathcal{E}_\bullet \cap \mathcal{E}_\circ = \emptyset$. The inductive link prediction task is defined as training a model on $\mathcal{T}_{train}$, running inference over a new graph $\mathcal{T}_{inf}$ and predicting missing links in the inference graph.

We use the two-sided workflow for evaluation that comprises both the head prediction and tail prediction task, i.e., for each evaluation triple $(h, r, t) \in \mathcal{T}_{eval}$, we consider the ranking tasks $(h, r, ?)$ and $(?, r, t)$, where we need to provide scores for each candidate entity $e \in \mathcal{E}_\circ$. We use *realistic* ranks, i.e., in case of ties, the assigned rank is the average of the best/smallest and the worst/largest rank in any ordering respecting the sort criterion [4]. As single-figure aggregations of the individual ranks, we use the following metrics, all sharing that larger numbers indicate better performance, and with an optimal value of 1.

H@k. The Hits at k (H@k) metric denotes the relative frequency of the correct entity's rank being at most k, for different $k \in \{1, 3, 5, 10, 100\}$. Its value range is $[0, 1]$.

MRR. The mean reciprocal rank (MRR) is the inverse of the harmonic mean of the individual ranks. In contrast to H@k, which

Split	ILPC22-S			ILPC22-L		
	$\|\mathcal{E}\|$	$\|\mathcal{R}\|$	$\|\mathcal{T}\|$	$\|\mathcal{E}\|$	$\|\mathcal{R}\|$	$\|\mathcal{T}\|$
Training	10,230	96	78,616	46,626	130	202,446
Inference	6,653	96	20,960	29,246	130	77,044
Inference (val.)	6,653	96	2,908	29,246	130	10,179
Inference (test.)	6,653	96	2,902	29,246	130	10,184
Inference (hold)	6,653	96	2,894	29,246	130	10,172
Total	16,883	96	108,280	75,872	130	310,045

Table 1: Statistics for ILPC22-S and ILPC22-L

only considers the first k entries, MRR considers all entries. However, due to the reciprocal transformation, there are diminishing returns, and, inherited by its foundation on the harmonic mean, MRR mostly focuses on small ranks. Its value range is $[0, 1]$.

AMRI. The adjusted mean rank index (AMRI) [4] is an affine transformation of the (arithmetic) mean rank, which is normalized for the chance effect: It is given as $AMRI = 1 - \frac{MR-1}{\mathbb{E}[MR]-1}$, where MR denotes the arithmetic mean of ranks, and $\mathbb{E}[MR]$ its expectation under random ordering. Thereby, it is implicitly normalized by the number of candidates, and thus allows comparison across cases with differing number of candidates, e.g., due to different number of entities, or filtered candidates. Its value range is $[-1, 1]$, where 0 corresponds to a performance equal to the expected performance under random ordering of entities for the ranking task.

4 DATASET CONSTRUCTION

The design principles behind creating *ILPC 2022* include that the dataset: i) should represent a real-world KG used in many practical downstream tasks; ii) should be challenging for modern inductive LP approaches and GNN architectures; iii) should be larger than existing benchmarks; iv) should allow for faster iteration and hypothesis testing; v) training and inference graphs should each be a connected component.

We chose CoDEx [19], a high-quality subset of Wikidata for transductive link prediction, as a source of triples then sampled two inductive datasets then using the following procedure:

(1) Join the training, validation, and test triples of the original transductive dataset into a single graph.
(2) Sample a set of entities $\mathcal{E}_\circ$ for the inference graph. The remaining entities $\mathcal{E}_\bullet$ comprise the training graph.
(3) Sample subgraphs induced by $\mathcal{E}_\bullet$ (training graph) and $\mathcal{E}_\circ$ (inference graph) from the total set of triples $\mathcal{T}$.
(4) Fix the set of relations $\mathcal{R}$ in the training graph and ensure the relations in the inference graph are a subset of $\mathcal{R}$, otherwise remove the triple from the inference graph.
(5) Remove disconnected nodes from both training and inference graphs to ensure the graphs are connected components.
(6) Sample validation triples $\mathcal{T}_{validation}$ and test triples $\mathcal{T}_{test}$ from the inference graph $\mathcal{T}_{inf}$ such that they only contain nodes from $\mathcal{T}_{inf}$.

Following this procedure, we sample two datasets, ILPC22-S (small) and ILPC22-L (large). For both datasets, the size of the inference graph is about 66% of the training graph in the number of entities and about 33% of the training graph in the number of triples. The sizes of the validation and test sets are about 10% of the inference graph. While the validation and test sets are openly available for participants comprising a *public* leaderboard, we also sample a *hold-out test set* of triples from the inference graph $\mathcal{T}_{inf}$ for the final evaluation of model submissions. The hold-out test set is kept privately by the authors and is not publicly available in the challenge repository.

The smaller ILPC22-S is designed for faster hypothesis testing and model iteration and consists of 10K entities and 96 relations in the training graph with 6.6K entities in the inference graph. The larger ILPC22-L is a challenging inductive dataset with 46K training and 30K inference entities with 130 relations, where the size of the inference graph (77K triples) is comparable to the whole training graph of smaller ILPC22-S. The overall statistics of ILPC22-S and ILPC22-L are presented in Table 1 and Figure 2.
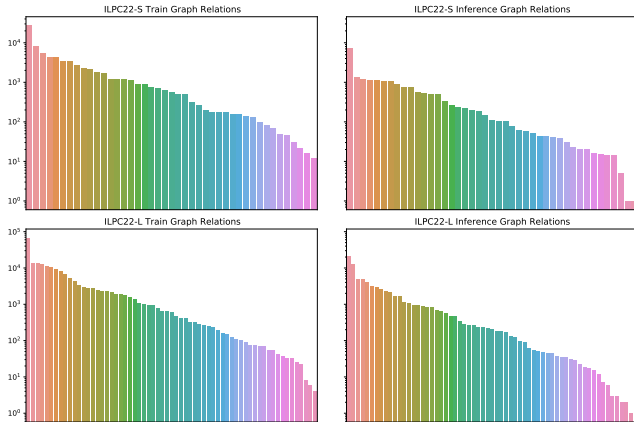

Figure 2: Distribution of relation type frequencies (without inverses) in training and inference graphs of ILPC22 datasets.

We used the Zenodo integration with GitHub to assign a DOI to each release to maximize reproducibility. Users should report/cite the versioned DOI of the dataset they used for training along with their results, such as [13].

5 EXPERIMENTS AND BASELINES

We provide two strong baselines employing NodePiece [12], a recently proposed inductive graph representation learning method that learns a fixed-size vocabulary of *tokens* that remain invariant at training and inference time. This way, both seen and unseen nodes can be uniformly *tokenized* using the same basic vocabulary and their representations are obtained via encoding the resulting *tokens set*. With the fully inductive setup of *ILPC 2022*, the vocabulary consists of unique relation types, their inverses, and an auxiliary padding token making the vocabulary size $|\mathcal{V}| = 2 \cdot |\mathcal{R}| + 1$ constant to the number of entities or total triples in graphs. For all baselines, NodePiece is configured to tokenize each entity with 5 unique outgoing relations and use a 2-layer MLP as a token encoder.

Parameter	ILPC22-S		ILPC22-L	
	-GNN	+GNN	-GNN	+GNN
Dimension		32		
Tokens per node		5		
GNN layers	-	2	-	2
Composition function	-	DistMult	-	DistMult
Scoring function		DistMult		
Loss function		Self-adversarial (NSSAL)		
Margin	5.0	2.0	15.0	20.0
Learning rate		0.0001		
Epochs	50	50	17	53
# Negatives		16		
# Parameters	15.5K	24K	15.5K	24K
Training (minutes)	6	77	5	480

Table 2: Hyperparameters

Metric	ILPC22-S		ILPC22-L	
	-GNN	+GNN	-GNN	+GNN
MRR	0.0381	0.1326	0.0651	0.0705
H@100	0.4678	0.4705	0.287	0.374
H@10	0.0917	0.2509	0.1246	0.1458
H@5	0.0500	0.1899	0.0809	0.0990
H@3	0.0219	0.1396	0.0542	0.0730
H@1	0.007	0.0763	0.0373	0.0319
AMRI	0.666	0.730	0.646	0.682

Table 3: Baseline evaluation of two variations of the Node-Piece model on both datasets

The first baseline uses just the NodePiece tokenization with the MLP encoder to obtain entity representations to be sent to the scoring function. It is designed to be the faster method for hypothesis testing and model iteration achieving high recall at moderate computational costs. The second baseline adds a message passing GNN encoder on top of the obtained NodePiece representations to further enrich entity and relation representations. Here, we use a 2-layer CompGCN [23] relational GNN architecture with the DistMult [26] composition function. We denote the first plain NodePiece baseline as -GNN and the second GNN-enabled baseline as +GNN.

Both implemented baselines have the same internal dimension of 32 (we did not find any noticeable performance improvement when increasing the dimensionality) and use the same DistMult scoring function (decoder). All models were trained in the negative sampling setup with self-adversarial loss (NSSAL) [20] using the Adam optimizer [16] on a single RTX 8000 GPU. During training, models on ILPC22-S consumed at most 2 GB VRAM, and at most 3 GB on larger ILPC22-L. Other hyperparameters are listed in Table 2.

Experiments were conducted using PyKEEN [3] and reported in Table 3. We observe that despite short training times, the plain NodePiece baselines (-GNN) already achieves relatively high recall in terms of H@100. We attribute that to the effect of relation-based entity tokenization, i.e., a sample of incident relations is discriminative enough for broad separation of possibly correct links which

is captured by H@100. While the GNN-enabled baseline exhibits 4-10× performance improvements ILPC22-S, it does not demonstrate such high gains for ILPC22-L, suggesting that the large dataset still poses challenges for state of the art GNNs and methodological improvements are still needed.

5.1 Participation

We solicit submissions from both individual researchers and teams, the submission instructions and leaderboards are provided in the official repository https://github.com/pykeen/ilpc2022.

6 CONCLUSION

We presented *ILPC 2022*, a new open challenge on inductive link prediction on knowledge graphs. For this challenge, we sampled two new datasets from Wikidata larger than existing benchmarks. We make sure the target metrics are meaningful, interpretable, and follow the best reproducibility criteria. For starters, we provided two baselines leveraging recently proposed inductive models that are fast to train on one hand or achieve higher performance on the other hand. We invite more submissions for this challenge extending the baselines or proposing new inductive LP models where the authors could estimate their performance on the public leaderboard. The challenge is expected to run for 6 months and we plan to report the intermediate results of submitted models on the private test set at the GLB workshop at the WebConf 2022.

ACKNOWLEDGMENTS

REFERENCES

[1] Mehdi Ali, Max Berrendorf, Mikhail Galkin, Veronika Thost, Tengfei Ma, Volker Tresp, and Jens Lehmann. 2021. Improving Inductive Link Prediction Using Hyper-relational Facts. In *International Semantic Web Conference*. Springer, 74–92.

[2] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Mikhail Galkin, Sahand Sharifzadeh, Asja Fischer, Volker Tresp, and Jens Lehmann. 2021. Bringing Light Into the Dark: A Large-scale Evaluation of Knowledge Graph Embedding Models under a Unified Framework. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2021), 1–1. https://doi.org/10.1109/TPAMI.2021.3124805

[3] Mehdi Ali, Max Berrendorf, Charles Tapley Hoyt, Laurent Vermue, Sahand Sharifzadeh, Volker Tresp, and Jens Lehmann. 2021. PyKEEN 1.0: A Python Library for Training and Evaluating Knowledge Graph Embeddings. *J. Mach. Learn. Res.* 22, 82 (2021), 1–6. http://jmlr.org/papers/v22/20-825.html

[4] Max Berrendorf, Evgeniy Faerman, Laurent Vermue, and Volker Tresp. 2020. On the Ambiguity of Rank-Based Evaluation of Entity Alignment or Link Prediction Methods. (feb 2020). arXiv:2002.06914 http://arxiv.org/abs/2002.06914

[5] Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. 2021. Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges. arXiv:2104.13478 [cs.LG]

[6] Nicola De Cao, Gautier Izacard, Sebastian Riedel, and Fabio Petroni. 2021. Autoregressive Entity Retrieval. In *International Conference on Learning Representations*. https://openreview.net/forum?id=5k8F6UU39V

[7] Nicola De Cao, Ledell Wu, Kashyap Popat, Mikel Artetxe, Naman Goyal, Mikhail Plekhanov, Luke Zettlemoyer, Nicola Cancedda, Sebastian Riedel, and Fabio Petroni. 2021. Multilingual Autoregressive Entity Linking. In *arXiv pre-print 2103.12528*. https://arxiv.org/abs/2103.12528

[8] Wenhu Chen, Wenhan Xiong, Xifeng Yan, and William Yang Wang. 2018. Variational Knowledge Graph Reasoning. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 1823–1832.

[9] Philipp Christmann, Rishiraj Saha Roy, Abdalghani Abujabal, Jyotsna Singh, and Gerhard Weikum. 2019. Look Before You Hop: Conversational Question Answering over Knowledge Graphs Using Judicious Context Expansion. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management* (Beijing, China) (CIKM '19). 729–738.

[10] Daniel Daza, Michael Cochez, and Paul Groth. 2021. Inductive Entity Representations from Text via Link Prediction. In *Proceedings of The Web Conference 2021*. https://doi.org/10.1145/3442381.3450141

[11] Tim Dettmers, Pasquale Minervini, Pontus Stenetorp, and Sebastian Riedel. 2018. Convolutional 2D Knowledge Graph Embeddings. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, Sheila A. McIlraith and Kilian Q. Weinberger (Eds.). AAAI Press, 1811–1818.

[12] Mikhail Galkin, Etienne Denis, Jiapeng Wu, and William L. Hamilton. 2022. NodePiece: Compositional and Parameter-Efficient Representations of Large Knowledge Graphs. In *International Conference on Learning Representations*. https://openreview.net/forum?id=xMJWUKJnFSw

[13] Michael Galkin and Charles Tapley Hoyt. 2022. *migalkin/ilpc2022: Initial pre-release*. https://doi.org/10.5281/zenodo.6213459

[14] Weihua Hu, Matthias Fey, Hongyu Ren, Maho Nakata, Yuxiao Dong, and Jure Leskovec. 2021. OGB-LSC: A Large-Scale Challenge for Machine Learning on Graphs. *arXiv preprint arXiv:2103.09430* (2021).

[15] Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. 2020. Open graph benchmark: Datasets for machine learning on graphs. *Advances in neural information processing systems* 33 (2020), 22118–22133.

[16] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

[17] Maximilian Nickel, Volker Tresp, and Hans-Peter Kriegel. 2011. A Three-Way Model for Collective Learning on Multi-Relational Data. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, Lise Getoor and Tobias Scheffer (Eds.). Omnipress, 809–816.

[18] Daniel Ruffinelli, Samuel Broscheit, and Rainer Gemulla. 2020. You CAN Teach an Old Dog New Tricks! On Training Knowledge Graph Embeddings. In *International Conference on Learning Representations*. https://openreview.net/forum?id=BkxSmlBFvr

[19] Tara Safavi and Danai Koutra. 2020. CoDEx: A Comprehensive Knowledge Graph Completion Benchmark. (2020). arXiv:2009.07810 http://arxiv.org/abs/2009.07810

[20] Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. RotatE: Knowledge Graph Embedding by Relational Rotation in Complex Space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

[21] Komal Teru, Etienne Denis, and Will Hamilton. 2020. Inductive relation prediction by subgraph reasoning. In *International Conference on Machine Learning*. PMLR, 9448–9457.

[22] Kristina Toutanova and Danqi Chen. 2015. Observed versus latent features for knowledge base and text inference. In *Proceedings of the 3rd Workshop on Continuous Vector Space Models and their Compositionality*. Association for Computational Linguistics, Beijing, China, 57–66. https://doi.org/10.18653/v1/W15-4007

[23] Shikhar Vashishth, Soumya Sanyal, Vikram Nitin, and Partha P. Talukdar. 2020. Composition-based Multi-Relational Graph Convolutional Networks. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

[24] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM* 57, 10 (2014), 78–85.

[25] Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. KEPLER: A Unified Model for Knowledge Embedding and Pre-trained Language Representation. *Trans. Assoc. Comput. Linguistics* 9 (2021), 176–194.

[26] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. 2015. Embedding Entities and Relations for Learning and Inference in Knowledge Bases. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, Yoshua Bengio and Yann LeCun (Eds.).

[27] Zhaocheng Zhu, Zuobai Zhang, Louis-Pascal A. C. Xhonneux, and Jian Tang. 2021. Neural Bellman-Ford Networks: A General Graph Neural Network Framework for Link Prediction. In *Advances in Neural Information Processing Systems*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (Eds.). https://openreview.net/forum?id=DEsIX_D_vR