PanRep: Universal node embeddings for heterogeneous graphs

Vassilis N. Ioannidis, Da Zheng, George Karypis
Amazon Web Services AI

ABSTRACT

Learning unsupervised node embeddings facilitates several downstream tasks such as node classification and link prediction. A node embedding is universal if it is designed to be used by and benefit various downstream tasks. This work introduces PanRep, a graph neural network (GNN) model, for unsupervised learning of universal node representations for heterogenous graphs. PanRep consists of a GNN encoder that obtains node embeddings and four decoders, each capturing different topological and node feature properties. Abiding to these properties the novel unsupervised framework learns universal embeddings applicable to different downstream tasks. PanRep can be furthered fine-tuned to account for possible limited labels. In this operational setting PanRep is considered as a pretrained model for extracting node embeddings of heterogenous graph data. PanRep outperforms all unsupervised and certain supervised methods in node classification and link prediction, especially when the labeled data for the supervised methods is small. PanRep-FT (with fine-tuning) outperforms all other supervised approaches, which corroborates the merits of pretraining models. Finally, we apply PanRep-FT for discovering novel drugs for Covid-19. We showcase the advantage of universal embeddings in drug repurposing and identify several drugs used in clinical trials as possible drug candidates.

ACM Reference Format:
Vassilis N. Ioannidis, Da Zheng, George Karypis. 2020. PanRep: Universal node embeddings for heterogeneous graphs. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Learning node representations from heterogeneous graph data powers the success of many downstream machine learning tasks such as node classification [28], and link prediction [46]. Graph neural networks (GNNs) learn node embeddings by applying a sequence of nonlinear operations parametrized by the graph adjacency matrix and achieve state-of-the-art performance in the aforementioned downstream tasks. The era of big data provides an opportunity for machine learning methods to harness large datasets [48]. Nevertheless, typically the labels in these datasets are scarce due to either lack of information or increased labeling costs [9]. The lack of labeled data points hinders the performance of supervised algorithms, which may not generalize well to unseen data and motivates *unsupervised* learning.

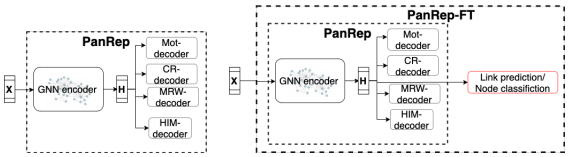

Figure 1: Illustration of the PanRep (left) and PanRep-FT (right) models. The GNN encoder processes the node features X to obtain the embeddings H. The decoders facilitate unsupervised learning of H. On the other hand, PanRep-FT is further fine-tuned for a few iterations by the task specific loss.

Unsupervised node embeddings may be used for downstream learning tasks, while the specific tasks are typically not known a priori. For example, node representations of the Amazon book graph can be employed for recommending new books as well as classifying a book's genre. This work aspires to provide *universal* node embeddings, which will be applied in multiple downstream tasks and achieve comparable performance to their supervised counterparts.

Although unsupervised learning has numerous applications, limited labels of the downstream task may be available. Refining the unsupervised universal representations with these labels could further increase the representation power of the embeddings. This can be achieved by *fine-tuning* the unsupervised model. Natural language processing methods have achieved state-of-the-art performance by applying such a fine-tuning framework [12]. Fine-tuning pretrained models is beneficial compared to end-to-end supervised learning since the former typically generalizes better especially when labeled data are limited [16] and decreases the inference time since typically just a few fine-tuning iterations typically suffice for the model to converge [12].

1.1 Contributions

This work introduces a framework for unsupervised learning of universal node representations on heterogenous graphs termed PanRep[1]. It consists of a GNN encoder that maps the heterogenous graph data to node embeddings and four decoders, each capturing different topological and node feature properties. The cluster and recover (CR) decoder exploits a clustering prior of the node attributes. The motif (Mot) decoder captures structural node properties that are encoded in the network motifs. The meta-path random walk (MRW) decoder promotes embedding similarity among nodes participating in a MRW and hence captures intermediate neighborhood structure. Finally, the heterogeneous information

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

[1] Pan: Pangkosmios (Greek for universal) and Rep: Representation

maximization (HIM) decoder aims at maximizing the mutual information among node local and the global representations per node type. These decoders model general properties of the graph data related to node homophily [19, 29] or node structural similarity [15, 36]. PanRep is solely supervised by the decoders and has no knowledge of the labels of the downstream task. The universal embeddings learned by PanRep are employed as features by models such as SVM [41] or DistMult [50] to be trained for the downstream tasks. To further accommodate the case where limited labels are available for some downstream tasks we propose fine-tuning PanRep (PanRep-FT). In this operational setting, PanRep-FT is optimized adhering to a task-specific loss. PanRep can be considered as a pretrained model for extracting node embeddings of heterogenous graph data. Figure 1 illustrates the two novel models. The contribution of this work is threefold. We introduce a novel problem formulation of universal unsupervised learning and design a tailored learning framework termed PanRep. We identify the following general properties of the heterogenous graph data: (i) the clustering of local node features, (ii) structural similarity among nodes, (iii) the local and intermediate neighborhood structure, (iv) and the mutual information among same-type nodes. We develop four novel decoders to model the aforementioned properties. We adjust the unsupervised universal learning framework to account for possible limited labels of the downstream task. PanRep-FT refines the universal embeddings and increases the model generalization capability.We compare the proposed models to state-of-the-art supervised and unsupervised methods for node classification and link prediction. PanRep outperforms all unsupervised and certain supervised methods in node classification, especially when the labeled data for the supervised methods is small. PanRep-FT outperforms even supervised approaches in node classification and link prediction, which corroborates the merits of pretraining models. Finally, we apply our method on the drug-repurposing knowledge graph for discovering drugs for Covid-19 and identify several drugs used in clinical trials as possible drug candidates.

2 RELATED WORK

Unsupervised learning. Representation learning amounts to mapping nodes in an embedding space where the graph topological information and structure is preserved [22]. Typically, representation learning methods follow the encoder-decoder framework advocated by PanRep. Nevertheless, the decoder is typically attuned to a single task based on e.g., matrix factorization [4, 8, 10, 32, 42], random walks [21, 33], or kernels on graphs [40]. Recently, methods relying on GNNs are increasingly popular for representation learning tasks [49]. GNNs typically rely on random walk-based objectives [21, 22] or on maximizing the mutual information among node representations [44]. Relational GNNs methods extend representation learning to heterogeneous graphs [14, 38, 39]. Relative to these contemporary works PanRep introduces multiple decoders to learn universal embeddings for heterogeneous graph data capturing the clustering of local node features, structural similarity among nodes, the local and intermediate neighborhood structure, and the mutual information among same-type nodes.

Supervised learning. Node classification is typically formulated as a semi-supervised learning (SSL) task over graphs, where the labels for a subset of nodes are available for training [7]. GNNs achieve state-of-the-art performance in SSL by utilizing regular graph convolution [28] or graph attention [43], while these models have been extended in heterogeneous graphs [17, 37, 47]. Similarly, another prominent supervised downstream learning task is link prediction with numerous applications in recommendation systems [46] and drug discovery [26, 54]. Knowledge-graph (KG) embedding models rely on mapping the nodes and edges of the KG to a vector space by maximizing a score function for existing KG edges [46, 50, 53]. RGCN models [37] have been successful in link prediction and contrary to KG embedding models can further utilize node features. The universal embeddings extracted from PanRep without labeled supervision offer a strong competitive to these supervised approaches for both node classification and link prediction tasks.

Pretraining. Pretraining models provides a significant performance boost compared to traditional approaches in natural language processing [12, 31, 34, 35] and computer vision [13, 18] . Pretraining offers increased generalization capability especially when the labeled data is scarce and increased inference speed relative to end-to-end training [12]. Recently, [25] introduced a framework for pretraining GNNs for graph classification. Different than [25] that focuses on graph representations, PanRep aims at node prediction tasks and obtains node representations via capturing properties related to node homophily [19, 29] or node structural similarity [36]. PanRep is a novel pretrained model for node classification and link prediction that requires significantly less labeled points to reach the performance of its fully supervised counterparts.

3 DEFINITIONS AND PROBLEM FORMULATION

A heterogeneous graph with T node types and R relation types is defined as $\mathcal{G} := \{\{\mathcal{V}_t\}_{t=1}^{T}, \{\mathcal{E}_r\}_{r=1}^{R}\}$. The node types represent the different entities and the relation types represent how these entities are semantically associated to each other. For example, in the IMDB network, the node types correspond to actors, directors, movies, etc., whereas the relation types correspond to *directed-by* and *played-in* relations. The number of nodes of type t is denoted by N_t and its associated nodal set by $\mathcal{V}_t := \{n_t\}_{n=1}^{N_t}$. The total number of nodes in $\mathcal{G}$ is $N := \sum_{t=1}^{t} N_t$. The rth relation type, $\mathcal{E}_r := \{(n_t, n'_{t'}) \in \mathcal{V}_t \times \mathcal{V}_{t'}\}$, holds all interactions of a certain type among $\mathcal{V}_t$ and $\mathcal{V}_{t'}$ and may represent that a movie is *directed-by* a director. Heterogenous graphs are typically used to represent knowledge graphs [46]. Each node n_t is also associated with an $F \times 1$ feature vector $\mathbf{x}_{n_t}$. This feature may be a natural language embedding of the title of a movie. The nodal features are collected in a $N \times F$ matrix $\mathbf{X}$. Note that certain node types may not have features and for these we use an embedding layer to represent their features.

Unsupervised learning. Given $\mathcal{G}$ and $\mathbf{X}$, the goal of representation learning is to estimate a function g such that $\mathbf{H} := g(\mathbf{X}, \mathcal{G})$, where $\mathbf{H} \in \mathbb{R}^{N \times D}$ represents the node embeddings and D is the size of the embedding space.Note that in estimating g, no labeled information is available.

Universal representation learning. The universal representations $\mathbf{H}$ should perform well on different downstream tasks. Different node classification and link prediction tasks may arise by considering different number of training nodes and links and different label types, e.g., occupation label or education level label. Consider I downstream task, for the universal representations $\mathbf{H}$ it holds that

$$\mathcal{L}^{(i)}(f^{(i)}(\mathbf{H}), \mathcal{T}^{(i)}) \leq \epsilon, \ i = 1, \ldots, I, \tag{1}$$

where $\mathcal{L}^{(i)}$, $f^{(i)}$, and $\mathcal{T}^{(i)}$ represent the loss function, learned classifier, and training set (node labels or links) for task i, respectively and ϵ is the largest error for all tasks. The goal of unsupervised universal representation learning is to learn $\mathbf{H}$ such that ϵ is small. While learning $\mathbf{H}$, PanRep does not have knowledge of $\{\mathcal{L}^{(i)}, f^{(i)}, \mathcal{T}^{(i)}\}_i$. Nevertheless, by utilizing the novel decoder scheme PanRep achieves superior performance even compared to supervised approaches across tasks.

4 PANREP

Our universal representation learning framework aims at embedding nodes in a low-dimensional space such that the representations are discriminative for node classification and link prediction. Methods for learning over graphs typically rely on modeling homophily of nodes that postulates neighboring vertices to have similar attributes [19, 29, 40, 51] or structural similarity among nodes [36], where vertices involved in similar graph structural patterns possess related attributes [15]. Motivated by these methods we identify related properties encoded in the graph data. Clustering nodes based on their attributes provides a strong signal for node homophily [30]. Network motifs reveal the local structure information for nodes in the graph [5]. Metapaths encode the heterogeneous graph neighborhood and indicate the local connectivity [14]. Finally, maximizing the mutual information among embeddings declusters node representations and provides further discriminative information [44].

Although the PanRep framework can utilize any GNN model as an encoder [49], in this paper PanRep uses a relational (R)GCN encoder [37]. RGCNs extend the graph convolution operation [28] to heterogenous graphs. An RGCN model is comprised by a sequence of RGCN layers. Essentially, the output of the RGCN layer for node n is a nonlinear combination of the hidden representations of neighboring nodes weighted based on the relation type.

4.1 Universal supervision signals

In order to capture the aforementioned properties we develop four novel universal decoders.

Cluster and recover supervision. Node attributes may reveal interesting properties of nodes, such as clusters of customers based on their buying power and age. This is important in recommendation systems, where traditional matrix factorization approaches [30] rely on revealing clusters of similar buyers. To capitalize such information we propose to supervise the universal embeddings by such cluster representations. Specifically, we cluster the node attributes via K-means [27] and then design a model that decodes $\mathbf{H}$ to recover the original clusters. The CR-decoder is modeled as a two layer MLP and is supervised by

$$\mathcal{L}_{\text{CR}} := -\sum_{n=1}^{N}\sum_{k=1}^{K} C_{nk} \ln \hat{C}_{nk}, \tag{2}$$

where the cluster assignment C_{nk} is 1 if node n belongs in class k and the predicted cluster assignment $\hat{C}_{nk}$ is the output of the CR-decoder. Such a supervision signal will enrich the universal embeddings $\mathbf{H}$ with information based on the clustering of local node features.

Motif supervision. Network motifs are sub-graphs where the nodes have specific connectivity patterns. Typical size-3 motifs for example, are the triangle and the star motifs. Each of these subgraphs is identified by a particular pattern of interactions among nodes, and reveals important properties for the participating nodes. In gene regulatory networks for example, motifs are associated with certain biological properties [6]. The work in [5] develops efficient parallel implementations for extracting network motifs. We aspire to capture structural similarity among nodes by predicting their motif information. The motivation is that nodes which might be distant in the graph may have similar structural properties as described by their motifs.

Using the method in [5] we extract a frequency vector $\boldsymbol{\mu}_n$ per node that shows how many times n participates to graph motifs up to size 4. This information reveals the structural role of nodes such as star-center, star-edge nodes, or bridge nodes [23, 36]. The motif decoder predicts this vector for all nodes using the universal representation $\mathbf{H}$. This allows for information sharing among nodes which are far away in the graph but have similar motif frequency vectors. The novel motif decoder is modeled as a two-layer MLP and is supervised by the following loss function

$$\mathcal{L}_{\text{MOT}} := \sum_{n=1}^{N} \|\boldsymbol{\mu}_n - \hat{\boldsymbol{\mu}}_n\|_2^2 \tag{3}$$

where $\hat{\boldsymbol{\mu}}_n$ is the output of the Mot-decoder for the nth node. Using the Mot-decoder PanRep enhances the universal embeddings by structural information encoded in the node motifs.

Metapath RW supervision. Metapaths are sequences of edges of possibly different type that connect nodes in a KG [14]. A metapath random walk (MRW) is a specialized RW that follows different edge-types; see e.g., [14].

We aspire to capture local connectivity patterns by promoting nodes participating in a MRW to have similar embeddings. Consider all node pairs for nodes $(n_t, n'_{t'})$ participating in a MRW, the following criterion maximizes the similarity among these nodes as follows

$$\mathcal{L}_{\text{MRW}} := \log(1 + \exp(-y \times \mathbf{h}_{n_t}^{\top} \text{diag}(\mathbf{r}_{t,t'}) \mathbf{h}_{n'_{t'}})), \tag{4}$$

where $\mathbf{h}_{n_t}$ and $\mathbf{h}_{n'_{t'}}$ are the universal embeddings for nodes n_t and $n'_{t'}$, respectively, $\mathbf{r}_{t,t'}$ is an embedding parametrized on the pair of node-types and y is 1 if n_t and $n'_{t'}$ co-occur in the MRW and -1 otherwise. Negative examples are generated by randomly selecting tail nodes for a fixed head node with ratio 5 negatives per positive example. Link prediction is indeed a special case of the MRW supervision that considers MRWs of length 1. However, metapaths convey more information than regular links since the

former can be defined to promote certain prior knowledge. For example, in predicting the movie genre in IMDB the metapath configured by the edge types (played by, played in) among node types (movie, actor, movie) will potentially connect movies with same genre and hence it is desirable. The embedding per node-type pair $\mathbf{r}_{t,t'}$ allows the MRW-decoder to weight the similarity among node embeddings from different node types accordingly. The length of the MRW controls the radius of the graph neighborhood considered in equation (4) and it can vary from local to intermediate.

Heterogenous information maximization. The aforementioned supervision signals capture clustering affinity, structural similarity and local and intermediate neighborhood of the nodes. Nevertheless, further information can be extracted by the representations by maximizing the mutual information among node representations. Such an approach for homogeneous graphs is detailed in [44], where the mutual information between node representations and the global graph summaries is maximized [24].

Towards further refining the universal embeddings, we propose an adaptation of [44] for heterogeneous graphs. We consider a global summary vector per t as $\mathbf{s}_t := \sum_{n_t=1}^{N_t} \mathbf{h}_{n_t}$ that captures the average tth node representation. We aspire to maximize the mutual information among $\mathbf{s}_t$ and the corresponding nodes in $\mathcal{V}_t$. The proposed HIM decoder is supervised by the following contrastive loss function

$$\mathcal{L}_{\text{HIM}} := \sum_{t=1}^{T} \left(\sum_{n_t=1}^{N_t} \log\left(\sigma(\mathbf{h}_{n_t}^\top \mathbf{W}\mathbf{s}_t)\right) + \log\left(1 - \sigma(\tilde{\mathbf{h}}_{n_t}^\top \mathbf{W}\mathbf{s}_t)\right)\right) \quad (5)$$

where the bilinear scoring function [50] $\sigma(\tilde{\mathbf{h}}_{n_t}^\top \mathbf{W}\mathbf{s}_t)$ captures how close is h_{n_t} to the global summary, $\mathbf{W}$ is a learnable matrix and $\tilde{\mathbf{h}}_{n_t}$ represents the negative example used to facilitate training. Designing negative examples is a cornerstone property for training contrastive models [44]. We generate the negative examples in (5) by shuffling node attributes among nodes of the same type. The HIM decoder maximizes the mutual information across nodes and complements the former decoders.

Putting everything together. PanRep's overall loss function is the linear unweighted combination of (2)-(5) and can be considered in the framework of deep multitask learning [52], since the GNN encoder is shared across the multiple supervision tasks and PanRep makes multiple inferences in one forward pass. A future direction of PanRep is to introduce adaptive weights per decoder to control its learning rate [11].

4.2 PanRep-FT

In certain cases a very small subset of labels may be known a priori for the downstream task. In such cases it is beneficial to fine-tune PanRep's model to obtain refined node representations. In this context, PanRep can be considered as pretrained model and a downstream task specific loss may be applied to supervise PanRep. BERT models in natural language processing have reported state of the art results by considering such a pretrain and fine-tune framework [12]. PanRep-FT can be considered a counterpart of BERT for extracting information from heterogenous graph data. PanRep-FT combines the benefit of universal unsupervised learning and task specific information and achieves greater generalization capacity especially when labeled data are scarce [16].

5 EXPERIMENTS

The proposed universal represention learning techniques are compared with state-of-the-art methods. For node classification the following contemporary methods are considered RGCN [37], HAN [47], MAGNN [17], node2vec [21], meta2vec [14] and an adaptations of the work in [44] for heterogenous graphs termed HIM. For link prediction the baseline models is RGCN [37] with DistMult supervision [50] that uses the same encoder as PanRep. The Mot-decoder and RC-decoder employ a 2-layer MLP. For the MRW-decoder we use length-2 MRWs. The parameters for all methods considered are optimized using the performance on the validation set. The method in this paper are implemented using the efficient deep graph learning (DGL) [45].

Datasets. We consider a subset of IMDB dataset [1] containing 11,616 nodes with 3 node-types and 17,106 edges from 6 edge-types. Each movie is associated with a label representing its genre and with a feature vector capturing its keywords. We also use a subset of the OAG dataset [2] with 23,696 with 4 node types (authors, affiliations, papers, venues) and 90,183 edges from 14 edge-types. In OAG we did not use MOT supervision since [5] is not applicable. Each paper is associated with a label denoting the scientific area and with an embedding of the papers' text. Finally, we utilize the drug-repurposing knowledge graph (DRKG) constructed in [26]. DRKG contains 5,874,261 biological interactions belonging to 107 edge-types among 97,238 biological entities from 13 entity-types. For further details on the datasets and configuration of methods see the supplementary material.

5.1 Node classification

We split the labeled nodes in 10% training, 5% validation, and 85% testing sets. In this experiment we compare supervised and unsupervised methods for classification. First, the methods learn embeddings for the labeled nodes with or without labeled supervision. We then obtain the embeddings corresponding to the 85% testing nodes as calculated from the unsupervised and supervised methods and further split these nodes to training and testing sets and train a linear SVM. This evaluation setting allows us to directly compare the different supervised and unsupervised approaches.

We report the Macro and Micro F1 accuracy for different training percentages of the 85% nodes fed to the SVM classifier in Table 1. It is observed that PanRep outperforms significantly other unsupervised approaches as well as some supervised approaches. In certain splits, PanRep outperforms its supervised counterpart RGCN that uses node labels for supervision. Metapath2vec [14] reports competitive performance for OAG in Macro-F1 score but unperformed in Micro-F1. This result is also in par with the Table **??** where the strongest signal for PanRep is given by the MRW decoder. PanRep-FT outperforms significantly RGCN that uses the same encoder, which is a testament to the power of pretraining models. Finally, PanRep-FT matches and outperforms in certain splits the state-of-the-art MAGNN that uses a more expressive encoder. PanRep's

Table 1: Node classification results.

	Train %		Unsupervised				Semi-supervised			
			node2vec	meta2vec	HIM	PanRep	HAN	MAGNN	RGCN	PanRep-FT
IMDB	Mac-F1	40%	50.63	47.57	55.21	**56.04**	56.15	**60.27**	58.48	59.49
		60%	51.65	48.17	57.66	**58.51**	57.29	**60.66**	58.42	59.86
		80%	51.49	49.99	57.89	**60.23**	58.51	61.44	58.76	**61.49**
	Mic-F1	40%	51.77	48.17	55.11	**55.92**	57.32	**60.50**	58.64	59.67
		60%	52.79	49.87	56.57	**58.41**	58.42	**60.88**	58.55	59.75
		80%	52.72	50.50	57.79	**60.14**	59.24	61.53	58.89	**61.59**
OAG	Mac-F1	40%	56.37	**65.75**	50.54	57.76	63.99	63.31	64.68	**64.72**
		60%	57.01	**66.09**	51.98	59.72	64.25	63.42	65.96	**66.99**
		80%	58.05	**65.75**	53.25	63.03	64.37	63.89	67.67	**67.90**
	Mic-F1	40%	70.17	74.54	71.91	**75.50**	73.95	72.74	**81.92**	80.36
		60%	70.95	74.96	73.89	**77.39**	75.32	72.75	81.39	**81.78**
		80%	72.24	74.73	75.31	**79.76**	75.24	73.43	82.38	**83.17**

Table 2: Drug inhibits gene scores for Covid-19.

PanRep-FT		RGCN	
Drug name	# hits	Drug name	# hits
Losartan	232	Chloroquine	69
Chloroquine	198	Colchicine	41
Deferoxamine	104	Tetrandrine	40
Ribavirin	101	Oseltamivir	37
Methylprednisolone	44	Azithromycin	36

We retain the top-5 drugs based on their number of hits for each method. Note that a random classifier will result to approximately 5.3 per drug. This suggests that the reported predictions are significantly better than random.

universal decoders enhance the embeddings with additional discriminative power that results to improved performance in the downstream tasks.

5.2 Link prediction

Our universal embedding framework is further evaluated for link prediction using the IMDB and OAG datasets. The MRW decoder is used to evaluate the performance of PanRep in link prediction. Figure 2 reports the MRR, and Hit-10 scores of the baseline methods along with the PanRep and PanRep-FT methods. We report the performance of the methods for different percentages of links used for training. Observe that PanRep-FT consistently outperforms the competing methods and the performance gain increases as the percentage of training links decreases. This corroborates the advantage of pretraining GNNs for link prediction. Note that PanRep reports similar performance with RGCN that is trained solely in link prediction. This result confirms the success of the universal embeddings in link prediction.

5.3 Drug repurposing

Drug-repurposing aims at discovering the most effective existing drugs to treat a certain disease. Using the Drug Repurposing Knowledge Graph (DRKG) [26], we compare the drug repurposing results in Covid-19 among PanRep-FT that is finetuned in link prediction and the baseline RGCN [37]. We employ $L = 1$ hidden layer with $D = 600$ and train for 800 epochs both networks. Drug-repurposing can be formulated as predicting direct links in the DRKG. Here, we attempt to predict whether a drug inhibits a certain gene, which is related to the target disease. We identify 442 genes that are related with the Covid-19 disease [20, 54]. We select 8,104 FDA-approved drugs in the DRKG as candidates; see also [26]. To validate our predictions we use 32 Covid-19 clinical trial drugs from [3].

For each gene node we calculate with RGCN and PanRep-FT an inhibit link score associated with every drug. Next, we score all 'drug-inhibits-gene' triples and rank them per target gene. We obtain in this way 442 ranked lists of drugs, one per gene node. Finally, to assess whether our prediction is in par with the drugs used for treatment, we check the overlap among the top 100 predicted drugs and the drugs used in clinical trials per gene. Table 2 lists the clinical drugs included in the top-100 predicted drugs across all the genes with their corresponding number of hits for the RGCN and PanRep-FT. It can be observed, that several of the widely used drugs in clinical trials appear high on the predicted list in both prediction. Furthermore, PanRep-FT reports a higher hit rate than RGCN, which corroborates the benefit of using the universal pretraining decoders. The universal representation endows PanRep with increased generalization power that allows for accurate link prediction performance when training data are extremely scarce as is the case of Covid-19. While this study, does not recommend specific drugs, it demonstrates a powerful deep learning methodology to prioritize existing drugs for further investigation, which holds the potential of accelerating therapeutic development for Covid-19.

6 CONCLUSION

This paper develops a novel framework for unsupervised learning of universal node representations on heterogenous graphs termed. To further facilitate cases where limited labels are available we implement PanRep-FT. Experiments in node classification and link prediction corroborate the competitive performance of the learned universal node representations compared to unsupervised and supervised methods. Experiments on the DRKG showcase the advantage of the universal embeddings in drug repurposing.

REFERENCES

[1] www.imdb.com, 2020.
[2] www.openacademic.ai/oag/, 2020.
[3] http://www.covid19-trails.com/, 2020.
[4] Amr Ahmed, Nino Shervashidze, Shravan Narayanamurthy, Vanja Josifovski, and Alexander J Smola. Distributed large-scale natural graph factorization. In *Proceedings of the 22nd international conference on World Wide Web*, pages 37–48, 2013.
[5] Nesreen K Ahmed, Jennifer Neville, Ryan A Rossi, Nick G Duffield, and Theodore L Willke. Graphlet decomposition: Framework, algorithms, and applications. *Knowledge and Information Systems*, 50(3):689–722, 2017.
[6] M Madan Babu, Nicholas M Luscombe, L Aravind, Mark Gerstein, and Sarah A Teichmann. Structure and evolution of transcriptional regulatory networks. *Current opinion in structural biology*, 14(3):283–291, 2004.
[7] M. Belkin, I. Matveeva, and P. Niyogi. Regularization and semi-supervised learning on large graphs. In *Proc. Annual Conf. Learning Theory*, volume 3120, pages 624–638, Banff, Canada, Jul. 2004. Springer.

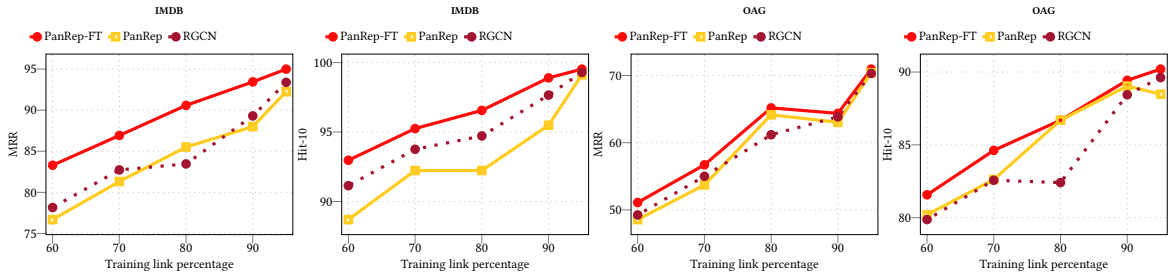

Figure 2: MRR and Hit-10 for link prediction across different percentages of testing links.

[8] Mikhail Belkin and Partha Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. In *Advances in neural information processing systems*, pages 585–591, 2002.

[9] Yoshua Bengio, Aaron C Courville, and Pascal Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR, abs/1206.5538*, 1:2012, 2012.

[10] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM international on conference on information and knowledge management*, pages 891–900, 2015.

[11] Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. *arXiv preprint arXiv:1711.02257*, 2017.

[12] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[13] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014.

[14] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 135–144, 2017.

[15] Claire Donnat, Marinka Zitnik, David Hallac, and Jure Leskovec. Learning structural node embeddings via diffusion wavelets. In *Proc. Intl. Conf. on Knowledge Disc. and Data Mining (KDD)*, 2018.

[16] Dumitru Erhan, Yoshua Bengio, Aaron Courville, Pierre-Antoine Manzagol, Pascal Vincent, and Samy Bengio. Why does unsupervised pre-training help deep learning? *Journal of Machine Learning Research*, 11(Feb):625–660, 2010.

[17] Xinyu Fu, Jiani Zhang, Ziqiao Meng, and Irwin King. Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding. In *Proceedings of The Web Conference 2020*, pages 2331–2341, 2020.

[18] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.

[19] David F Gleich. Pagerank beyond the web. *SIAM Review*, 57(3):321–363, 2015.

[20] David E Gordon, Gwendolyn M Jang, Mehdi Bouhaddou, Jiewei Xu, Kirsten Obernier, Matthew J O'meara, Jeffrey Z Guo, Danielle L Swaney, Tia A Tummino, Ruth Huttenhain, et al. A sars-cov-2-human protein-protein interaction map reveals drug targets and potential drug-repurposing. *Nature*, 2020.

[21] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 855–864, 2016.

[22] William L Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*, 2017.

[23] Keith Henderson, Brian Gallagher, Tina Eliassi-Rad, Hanghang Tong, Sugato Basu, Leman Akoglu, Danai Koutra, Christos Faloutsos, and Lei Li. Rolx: structural role extraction & mining in large graphs. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1231–1239, 2012.

[24] R Devon Hjelm, Alex Fedorov, Samuel Lavoie-Marchildon, Karan Grewal, Phil Bachman, Adam Trischler, and Yoshua Bengio. Learning deep representations by mutual information estimation and maximization. *arXiv preprint arXiv:1808.06670*, 2018.

[25] Weihua Hu, Bowen Liu, Joseph Gomes, Marinka Zitnik, Percy Liang, Vijay Pande, and Jure Leskovec. Strategies for pre-training graph neural networks. In *International Conference on Learning Representations*, 2019.

[26] Vassilis N. Ioannidis, Xiang Song, Saurav Manchanda, Mufei Li, Xiaoqin Pan, Da Zheng, Xia Ning, Xiangxiang Zeng, and George Karypis. Drkg - drug repurposing knowledge graph for covid-19. https://github.com/gnn4dr/DRKG/, 2020.

[27] Tapas Kanungo, David M Mount, Nathan S Netanyahu, Christine D Piatko, Ruth Silverman, and Angela Y Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE transactions on pattern analysis and machine intelligence*, 24(7):881–892, 2002.

[28] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proc. Int. Conf. on Learn. Represantions*, Toulon, France, Apr. 2017.

[29] Kyle Kloster and David F Gleich. Heat kernel based community detection. In *Proc. Intl. Conf. on Knowledge Disc. and Data Mining (KDD)*, pages 1386–1395, 2014.

[30] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[31] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. Albert: A lite bert for self-supervised learning of language representations, 2019.

[32] Mingdong Ou, Peng Cui, Jian Pei, Ziwei Zhang, and Wenwu Zhu. Asymmetric transitivity preserving graph embedding. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1105–1114, 2016.

[33] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 701–710, 2014.

[34] Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*, 2018.

[35] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[36] Ryan A Rossi and Nesreen K Ahmed. Role discovery in networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(4):1112–1131, 2014.

[37] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer, 2018.

[38] Jingbo Shang, Meng Qu, Jialu Liu, Lance M Kaplan, Jiawei Han, and Jian Peng. Meta-path guided embedding for similarity search in large-scale heterogeneous information networks. *arXiv preprint arXiv:1610.09769*, 2016.

[39] Chuan Shi, Binbin Hu, Wayne Xin Zhao, and S Yu Philip. Heterogeneous information network embedding for recommendation. *IEEE Transactions on Knowledge and Data Engineering*, 31(2):357–370, 2018.

[40] A. J. Smola and R. I. Kondor. Kernels and regularization on graphs. In *Learning Theory and Kernel Machines*, pages 144–158. Springer, 2003.

[41] Johan AK Suykens and Joos Vandewalle. Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300, 1999.

[42] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web*, pages 1067–1077, 2015.

[43] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. In *Proc. Int. Conf. on Learn. Represantions*, 2018.

[44] Petar Veličković, William Fedus, William L Hamilton, Pietro Liò, Yoshua Bengio, and R Devon Hjelm. Deep graph infomax. *arXiv preprint arXiv:1809.10341*, 2018.

[45] Minjie Wang, Lingfan Yu, Da Zheng, Quan Gan, Yu Gai, Zihao Ye, Mufei Li, Jinjing Zhou, Qi Huang, Chao Ma, et al. Deep graph library: Towards efficient and scalable deep learning on graphs. *arXiv preprint arXiv:1909.01315*, 2019.

[46] Quan Wang, Zhendong Mao, Bin Wang, and Li Guo. Knowledge graph embedding: A survey of approaches and applications. *IEEE Transactions on Knowledge and Data Engineering*, 29(12):2724–2743, 2017.

[47] Xiao Wang, Houye Ji, Chuan Shi, Bai Wang, Yanfang Ye, Peng Cui, and Philip S Yu. Heterogeneous graph attention network. In *The World Wide Web Conference*, pages 2022–2032, 2019.
[48] Xindong Wu, Xingquan Zhu, Gong-Qing Wu, and Wei Ding. Data mining with big data. *IEEE transactions on knowledge and data engineering*, 26(1):97–107, 2013.
[49] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.
[50] Bishan Yang, Wen-tau Yih, Xiaodong He, Jianfeng Gao, and Li Deng. Embedding entities and relations for learning and inference in knowledge bases. *arXiv preprint arXiv:1412.6575*, 2014.
[51] Guangchao Yuan, Pradeep K Murukannaiah, Zhe Zhang, and Munindar P Singh. Exploiting sentiment homophily for link prediction. In *Proceedings of the 8th ACM Conference on Recommender systems*, pages 17–24, 2014.
[52] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European conference on computer vision*, pages 94–108. Springer, 2014.
[53] Da Zheng, Xiang Song, Chao Ma, Zeyuan Tan, Zihao Ye, Jin Dong, Hao Xiong, Zheng Zhang, and George Karypis. Dgl-ke: Training knowledge graph embeddings at scale. *arXiv preprint arXiv:2004.08532*, 2020.
[54] Yadi Zhou, Yuan Hou, Jiayu Shen, Yin Huang, William Martin, and Feixiong Cheng. Network-based drug repurposing for novel coronavirus 2019-ncov/sars-cov-2. *Cell discovery*, 6(1):1–18, 2020.

A IMPLEMENTATION FRAMEWORK

The methods presented in this paper are implemented in the efficient deep graph learning (DGL)[2] library [45]. PanRep is implemented using the mini-batch training framework that facilitates training for very large graphs even with limited computational resources [3]. The competing methods RGCN, MAGNN and HAN are also implemented using the DGL. PanRep experiments are executed on an AWS P3.8xlarge[4] instances with 8 GPUs each having 16GB of memory.

B METHODS

Different competing methods include RGCN [37], HAN [47], MAGNN [17], node2vec [21], meta2vec [14] and an adaptation of the work in [44] for heterogenous graphs termed HIM. For link prediction the baseline model is RGCN [37] with DistMult supervision [50] that uses the same encoder as PanRep. The Mot-decoder and RC-decoder employ a 2-layer MLP. For the MRW-decoder we use length-2 MRWs. The parameters for all methods considered are optimized using the performance on the validation set. For the majority of the experiments PanRep uses a hidden dimension of 300, 1 hidden layer, 800 epochs of model training, 100 epochs for finetuning, and a learning rate of 0.001. For link prediction finetuning PanRep uses a DistMult model [50] whereas for node classification it uses a logistic loss.

C DATASETS

C.1 DRKG

The *Drug Repurposing Knowledge Graph* (DRKG) contains 97055 entities belonging to 13 entity-types [26]. The type-wise distribution of the entities is shown in Table 4. DRKG contains a total of 5869294 triplets belonging to 107 edge-types. Table 5 shows the number of triplets between different entity-type pairs for DRKG and various data sources. The DRKG is publicly available.[5]

C.2 IMDB and OAG

IDMB [1] is a movie database including information about the cast, production crew, and plot summaries. A subset of IMDb is used after data preprocessing in Table 3. Movies are labeled as one of three classes (Action, Comedy, and Drama) based on their genre information. Each movie is also described by a bag-of-words representation of its plot keywords.

OAG [2] is bibliography website. We preprocess the data and retain the subgraph in Table 3. The papers are divided into 6 research areas. Each paper is described by a BERT embedding of the paper's title.

[2] https://www.dgl.ai/
[3] https://github.com/dmlc/dgl/blob/master/examples/pytorch/rgcn-hetero
[4] https://aws.amazon.com/ec2/instance-types/p3/
[5] https://github.com/gnn4dr/DRKG/

Table 3: Statistics of datasets.

Dataset	Node	Edge
IMDb	# movie (M): 4,278 # director (D): 2,081 # actor (A): 5,257	# M-directed by-D: 4,278 , D-directed-M: 4,278 # M-played by-A: 12,828, A-played-M: 12,828
OAG	# author (A): 13,720 # paper (P): 7,326 # affiliation (Af): 2,290 # venue (V): 782	# P-in journal-V: 3941, V-journal has-P: 3941 # P-conference-V: 3368, V-conference has-P: 3368 # P-cites-P: 3327, P-cited by-P: 3327 # A-writes as last-P: 4522, P-written by last-A: 4522 # A-writes as other-P: 7769, P-written by other-A: 7769 # A-writes as first-P: 4795, P-written by first-A: 4795 # A-affiliated with-Af: 17035, Af-affiliated with-A: 17035

Table 4: Number of nodes per node type in the DRKG and the data sources.

Entity type	Drugbank	GNBR	Hetionet	STRING	IntAct	DGIdb	Bibliography	Total Entities
Anatomy	-	-	400	-	-	-	-	400
Atc	4,048	-	-	-	-	-	-	4,048
Biological Process	-	-	11,381	-	-	-	-	11,381
Cellular Component	-	-	1,391	-	-	-	-	1,391
Compound	9,708	11,961	1,538	-	153	6,348	6,250	24,313
Disease	1,182	4,746	257	-	-	-	33	5,103
Gene	4,973	27,111	19,145	18,316	16,321	2,551	3,181	39,220
Molecular Function	-	-	2,884	-	-	-	-	2,884
Pathway	-	-	1,822	-	-	-	-	1,822
Pharmacologic Class	-	-	345	-	-	-	-	345
Side Effect	-	-	5,701	-	-	-	-	5,701
Symptom	-	-	415	-	-	-	-	415
Tax	-	215	-	-	-	-	-	215
Total	19,911	44,033	45,279	18,316	16,474	8,899	9,464	97,238

Table 5: Number of interactions in the DRKG and the data sources.

Entity-type pair	Drugbank	GNBR	Hetionet	STRING	IntAct	DGIdb	Bibliography	Total interactions
('Gene', 'Gene')	-	667,22	474,526	1,496,708	254,346	-	58,629	2,350,931
('Compound', 'Gene')	24,801	80,803	51,429	-	1,805	26,290	25,666	210,794
('Disease', 'Gene')	-	95,399	27,977	-	-	-	461	123,837
('Atc', 'Compound')	15,750	-	-	-	-	-	-	15,750
('Compound', 'Compound')	1,379,271	-	6,486	-	-	-	-	1,385,757
('Compound', 'Disease')	4,968	77,782	1,145	-	-	-	-	83,895
('Gene', 'Tax')	-	14,663	-	-	-	-	-	14,663
('Biological Process', 'Gene')	-	-	559,504	-	-	-	-	559,504
('Disease', 'Symptom')	-	-	3,357	-	-	-	-	3,357
('Anatomy', 'Disease')	-	-	3,602	-	-	-	-	3,602
('Disease', 'Disease')	-	-	543	-	-	-	-	543
('Anatomy', 'Gene')	-	-	726,495	-	-	-	-	726,495
('Gene', 'Molecular Function')	-	-	97,222	-	-	-	-	97,222
('Compound', 'Pharmacologic Class')	-	-	1,029	-	-	-	-	1,029
('Cellular Component', 'Gene')	-	-	73,566	-	-	-	-	73,566
('Gene', 'Pathway')	-	-	84,372	-	-	-	-	84,372
('Compound', 'Side Effect')	-	-	138,944	-	-	-	-	138,944
Total	1,424,790	335,369	2,250,197	1,496,708	256,151	26,290	84,756	5,874,261