A Heterogeneous Dynamical Graph Neural Networks Approach to Quantify Scientific Impact

Xovee Xu*, Fan Zhou*‡, Ce Li*, Goce Trajcevski† and Ting Zhong*
*School of Information and Software Engineering, University of Electronic Science and Technology of China, China
†Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA
‡Corresponding author: fan.zhou@uestc.edu.cn

***Abstract*—Quantifying and predicting the long-term impact of scientific writings or individual scholars has important implications for many policy decisions, such as funding proposal evaluation and identifying emerging research fields. In this work, we propose an approach based on heterogeneous dynamical graph neural network (HDGNN) to explicitly model and predict the cumulative impact of scientific papers and authors. HDGNN extends heterogeneous GNNs by incorporating temporally evolving characteristics and capturing both structural properties of attributed graph and the growing sequence of citation behavior. HDGNN is significantly different from previous models in its capability of modeling the node impact in a dynamic manner while taking into account the complex relations among nodes. Extensive experiments demonstrate its superior performance of predicting the impact of both papers and authors.**

***Index Terms*—heterogeneous graph neural networks, scientific impact prediction, information diffusion.**

I. Introduction

The pace of growth of the body of scientific research has been rapidly increasing in recent years. For example, the number of records in DBLP[1] has increased from 2,486,800 in 2013, to 4,893,893 in 2019; according to the AI index report 2019[2], the number of peer-reviewed AI publications has increased by 300% between 1998-2018. Quantifying the impact of the publications, as well as individual scholars/authors, is an important task in many domains of societal and scientific relevance, e.g., funding agencies and research institutes need to deeply understand the current research development – discovering frontier ideas, identifying breakthrough topics and productive scholars, seeking well-fitted scientists for defined projects, hiring high-quality faculties – for improved policy and decision making [1]. The availability of various scientific databases, such as Web of Science, Google Scholar, DBLP, IEEE *Xplore*, ACM DL, etc., provides an unprecedented opportunity to explore the career of scientists and the dynamic evolving process of paper dissemination. However, scientific impacts of scholars and papers can be affected by a variety of factors, e.g., a productive researcher may publish a number of papers every year, but the impact of her/his publications may vary significantly [2]. Also, some scientific findings may receive a burst of attention immediately, while others may take decades since their original publication date [3].

[1] https://dblp.uni-trier.de/statistics/recordsindblp
[2] https://hai.stanford.edu/ai-index/2019

A. Existing Works

Quantifying and foreseeing the (impact of) scientific diffusion have been scrutinized by generations of researchers since [4]. Earlier efforts [5]–[7] primarily focused on extracting indicative features and discovering latent mechanisms that drive the accumulation of citations. Features of scholars such as the number of publications and citations, were used to forecast the future h-index in [6]. Factors such as topical authority and publication venue that may increase citations were used to predict the scientific impact in [8]. Despite certain merits, the previous works are limited on the impact predictability due to the confluence of different and sometimes controversial factors [3], [9], and the difficulty of generalizing the knowledge from one discipline to another. Meanwhile, some implicit but important factors are not fully leveraged in a scientific way, such as academic authority that amplifies author/paper exposure and facilitates grants funding. Another line of work predicts the propagation of scientific impact from the perspective of stochastic information dynamics, relying on various pattern-recognition based models (e.g., Poisson and Hawkes processes) [10], [11]. These methods are theoretically solid and demonstrated their advancement, particularly for interpretability, but they require longer sequences of observations and are unable to fully leverage the complex interactions among authors and papers for impact prediction.

Recent applications of deep neural networks on graph-based data have inspired numerous models for capturing temporal and sequential process of information diffusion. DeepCas [12] is a graph-embedding based prediction model, it learns the representation of cascade graphs with DeepWalk [13] and the diffusion process via recurrent neural networks (RNNs) [14] and attention mechanism [15]. CasCN [16] exploits the structure of each information cascade by a dynamic graph convolutional network (GCN) [17], and predicts the size of cascades while taking the directionality of cascades and time decay effects into consideration. However, these approaches deal with representation learning of homogeneous graphs, which limits their capability of exploiting the information associated with multiple node attributes and complex relations among heterogeneous nodes. Thus, incorporating meaningful relations among nodes and edges into the information diffusion remains one of the unaddressed issues in existing methods, which also motivates our work.

B. Present Work

In this paper, we propose a heterogeneous dynamical graph neural networks (HDGNN) to study the dynamic evolving process of scientific impact while capturing rich semantics embedded in bibliographic graphs. HDGNN bridges the gap between dynamical GNNs [18], [19] and heterogeneous information network (HIN) embedding [20]–[22], which have largely been studied independently in the prior works. HDGNN learns academic graph representation with a heterogeneous GNN that aggregates neighboring features of nodes with a weighted contextualized node selection strategy and temporal-attentive representation network, while preserving the unevenly distributed scientific impact of nodes. It also captures the dynamic evolution of nodes and the temporal dependencies among papers/authors, by encoding temporal cascading information into node representations which, in turn, sheds light on the underlying mechanism that accumulates the impact for both authors and papers.

Our main contributions are four-fold: (1) We study the scientific impact prediction problem from the view of heterogeneous graph learning compared to prevalent homogeneous graph/cascade models [12], [16], which allows us to capture more complex and rich interactions among different types of nodes and edges; (2) We present a novel scientific impact diffusion learning model with a newly designed weighted heterogeneous node sampling scheme, combined with the multi-head attention mechanism, for aggregating node features and heterogeneous neighbors; (3) We extend the GNN models with temporal horizon, enabling us to address the dynamic prediction problem – compared to existing graph embedding works that mainly focus on representation learning and/or *static* tasks such as link prediction and node classification; (4) We conduct extensive evaluations on a large-scale academic dataset consisting of 1M nodes and 54M edges. The experimental results show that the proposed model achieves up to 17.24% improvement over the state-of-the-art baselines. To facilitate reproducibility of the results report in this paper, we make the source code of HDGNN publicly available at https://github.com/Xovee/hdgnn.

II. Preliminaries

We now introduce the necessary background and formally define our problem settings. Consider *papers* and *authors* as two independent sets of entities, denoted as P and A, respectively. For each paper p, $p.t$ indicates the time since its first publication. For each author a, $a.t$ represents how many years this author has been publishing papers. Let t_r be the reference time and t_p be the prediction time, c_p^t be the number of citations of paper p at time $p.t$, and c_a^t the number of citations of author a at time $a.t$. The scientific impact predictions for papers and authors can be defined as regression problems as follows:

*Problem 1 (**Scientific impact prediction for papers**):* Given N papers $\{p_i\}_{i\in N}$, for each paper p_i and its associated observations at time $p.t_r$, we aim to predict its total number of citations $c_{p_i}^{t_p}$ at prediction time $p.t_p$, i.e., how many times this paper has been cited since publication.

*Problem 2 (**Scientific impact prediction for authors**):* Given M authors $\{a_i\}_{i\in M}$, for each author a_i and its associated observations at time $a.t_r$, we aim to predict its total number of citations $c_{a_i}^{t_p}$ at prediction time $a.t_p$, i.e., how many times this author has been cited since her first publication.

Given the above we build an academic heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A}, \mathcal{R}, \mathbf{C}_\mathcal{V})$, where $\mathcal{V}$ denotes the set of nodes and $\mathcal{E}$ is the set of weighted and directed edges indicating node relations. For each node in $\mathcal{V}$, it is associated with a node type in $\mathcal{A}$ (we consider $|\mathcal{A}| = 3$ types of nodes: paper, author, and venue). Edges in $\mathcal{E}$ are described by 7 different types defined in $\mathcal{R}$: author *writes* paper, author *collaborates with* author, author *publishes in* venue, author *cites* paper, paper *is published in* venue, paper *cites* paper, and paper *cites* author(s). Additionally, node features are represented by $\mathbf{C}_\mathcal{V}$, including content of papers, profiles of authors, etc.

Suppose we have N papers and M authors. During an observation window each paper or author can be cited by other papers/authors. Then, the sequence of citations can be represented as $[(p_j, t_j)]_j (t_j \leq t_r)$, i.e., Given the exact number of citations c^{t_p} at prediction time t_p for a particular paper/author, the scientific impact prediction problem can be solved by optimizing the mean square error loss between predicted number of citations $\hat{c}^{t_p}$ and true number c^{t_p}.

III. Methodology: HDGNN

We now present our proposed model HDGNN, which consists of two main building blocks: (i) heterogeneous representation learning via graph neural networks; and (ii) temporal paper sequence modeling and author aggregating via recurrent neural networks. For simplicity, here we use *scientific impact prediction for papers* as an illustrative scenario, with a note that the results can be easily generalized to the scenario of *scientific impact prediction for authors* (we show the prediction results for both settings in Section IV).

A. Heterogeneous Graph Representation

The first part of HDGNN is to learn representation of nodes in $\mathcal{G}$. Specifically, for a paper node p, author node a, and venue node v – given heterogeneous neighbors in a non-Euclidean graph structure – we learn a low-dimensional node embedding $E(p/a/v)$ via a mapping function $f : p/a/v \to E(p/a/v) \in \mathbb{R}^{d_E}$ and the embeddings preserve neighboring proximity. Towards that, we borrow the idea of random walk with restart [23] and a deep neural network [24] architecture from [20] to model the heterogeneous graph learning.

1) Heterogeneous neighboring node sampling: Given a node n (paper/author/venue) in graph $\mathcal{G}$, the distribution of its neighboring nodes may be highly skewed, i.e., some nodes connect to a large number of other nodes (e.g., those highly cited papers/authors) while most of them only have a few neighbors, greatly following the heavy-tailed distribution of citations [1]. High impact journals, productive authors, or influential papers, often have higher degrees compared to

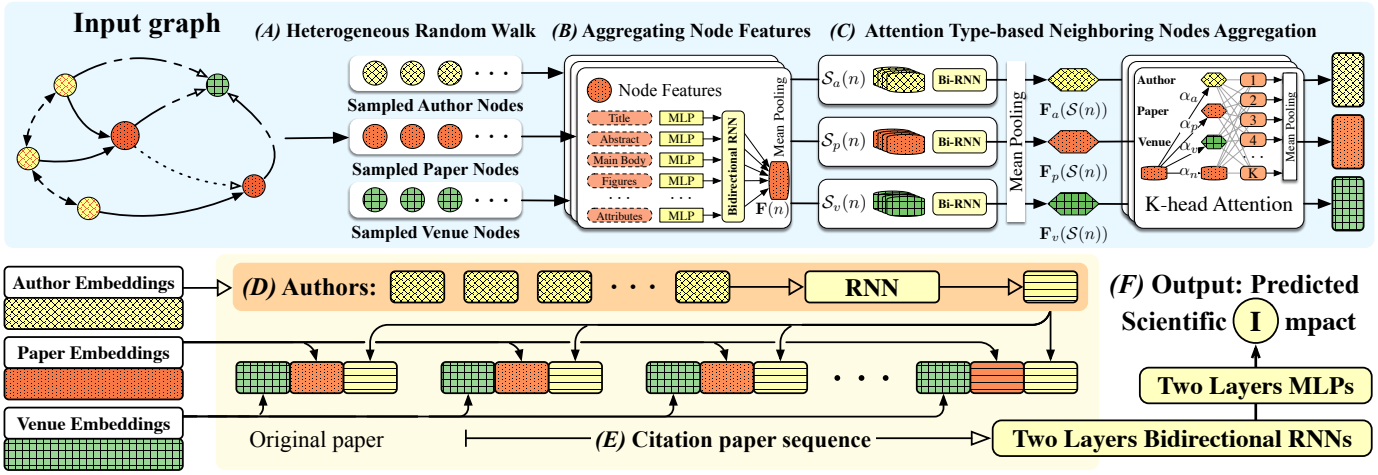


Fig. 1. The overall architecture of HDGNN: ***(A)***: Walk heterogeneous nodes by using a weighted contextualized node selection strategy based on random walk with restart; ***(B)*** and ***(C)***: Aggregating node features and heterogeneous neighbors of nodes with bidirectional RNNs and multi-head attention mechanism; ***(D)***: Multiple author aggregation; ***(E)***: Sequential citation aggregation; ***(F)***: Scientific impact predictor with RNNs and MLPs.

other majorities. To accommodate this factor into our model, we design a weighted contextualized node selection strategy based on random walk, which is more suitable for capturing scientific impact and imbalanced distribution of nodes in the heterogeneous academic graph. Specifically, for each current step, a given node n either returns to the previous node with probability q, or jumps to the next neighbor node with probability $1-q$. Let $\mathcal{N}(n)$ be the set of n's neighbors. Then the node n has a probability $1-q$ to select one of its neighbors $\mathcal{N}(n)$ based on node types $\mathcal{A}$, edge types $\mathcal{R}$, and node/edge characteristics. Specifically, the probability to walk to the next node m from n is:

$$\Pr(m|\mathcal{N}(n),\mathcal{G}) = \begin{cases} (1-q)\alpha D^{\alpha}(m), & \text{if } \mathcal{A}_m \text{ is paper} \\ (1-q)\beta D^{\beta}(m), & \text{if } \mathcal{A}_m \text{ is author} \\ (1-q)\gamma D^{\gamma}(m), & \text{if } \mathcal{A}_m \text{ is venue} \end{cases} \tag{1}$$

where $D^{\alpha}(*), D^{\beta}(*), D^{\gamma}(*)$ are influence functions measuring node influence from various factors, e.g., arrival time, node degrees, pagerank scores, and similarities, according to the node type $\mathcal{A}_m$ or edge type $\mathcal{R}_e$.

Through running random walks iteratively, we can sample a fixed number of nodes for each node type in $\mathcal{A}$, resulting in three sets denoted as: $\mathcal{S}(p)$, $\mathcal{S}(a)$, and $\mathcal{S}(v)$. Note that we consider edge directions, weights, and node degrees when sampling representative heterogeneous neighbors.

2) Aggregating node features: After sampling the neighbors for each node, we utilize bidirectional Gated Recurrent Units (Bi-GRUs) [14] to model the dependencies among the nodes' content features. Assuming that there are k content features for one specific type of nodes, the feature aggregation can be formalized as:

$$\mathbf{F}(n) = \frac{1}{k}\sum_{i=1}^{k}\left(\overrightarrow{\text{GRU}}(\mathbf{h}_n^i)\,||\,\overleftarrow{\text{GRU}}(\mathbf{h}_n^i)\right), \tag{2}$$

$$\mathbf{h}_n^i = \text{MLP}(\mathbf{C}_n^i), \text{ for } i = 1,2,\ldots,k \tag{3}$$

where $\mathbf{F}(n) \in \mathbb{R}^{d_n}$ is the aggregated embedding of node n computed by mean pooling; $||$ denotes the concatenation operation; $\mathbf{C}_n$ are k heterogeneous node content features and $\mathbf{h}_n^i \in \mathbb{R}^{d_{\mathbf{h}}}$ is the output of the Multi-Layer Perceptron (MLP). In practical applications, various content features can be used here to enhance the model learning ability – e.g., meta-data and the text of papers (title, abstract, main body), illustrations (figure, table), past publications of authors/venues, metadata of authors/venues (profile, honor, research area).

3) Aggregating heterogeneous neighbors: After aggregating node content features, for each node n in the graph $\mathcal{G}$ we have its corresponding aggregated features $\mathbf{F}(n)$. Then we are ready to use a type-based RNN to aggregate embeddings of the neighbors in $\mathcal{S}(n)$. For each node type in $\mathcal{A}$ (in our case the paper/author/venue), $\mathcal{S}_{p/a/v}(n)$ is the homogeneous type-specific neighboring set of node n and $\text{RNN}_{p/a/v}$ is a type-specific aggregator. More specifically, HDGNN utilizes another Bi-GRU for modeling n's neighbors:

$$\mathbf{F}_{\mathcal{A}_n}(\mathcal{S}_{\mathcal{A}_n}(n)) = \frac{\sum_{i=1}\left(\overrightarrow{\text{GRU}}(\mathbf{F}(i))||\overleftarrow{\text{GRU}}(\mathbf{F}(i))\right)}{|\mathcal{S}_{\mathcal{A}_n}(n)|}, \quad \text{for } i = 1,2,\ldots,|\mathcal{S}_{\mathcal{A}_n}(n)| \tag{4}$$

where $\mathbf{F}_{\mathcal{A}_n}(\mathcal{S}(n)) \in \mathbb{R}^{d_s}$ is the output embedding from the homogeneous neighboring set $\mathcal{S}_{\mathcal{A}_n}(n)$, and d_s is the dimension of aggregated neighboring embeddings of node n.

In HDGNN we use deterministic neural networks, bidirectional RNN, and mean pooling as aggregators of node's content along with node's neighbors. Alternatively, other types of aggregators, e.g., last hidden state of RNNs, CNNs, max or sum pooling, can be used (cf. [20], [25]).

4) Multi-head attention for type-based neighbors: With each type-based neighboring aggregators in hand, we are able

to combine them using multi-head attention mechanism [15]:

$$\alpha_i = \frac{\exp\left(\text{LeakyReLU}(u^T[\mathbf{F}(n)||\mathbf{F}_i^{\mathcal{S}})]\right)}{\sum_{j\in\mathcal{S}'(n)}\exp\left(\text{LeakyReLU}(u^T[\mathbf{F}(n)||\mathbf{F}_j^{\mathcal{S}}])\right)}, \quad (5)$$

$$\mathcal{S}'(n) = \mathbf{F}(n) \cup \{\mathbf{F}_j^{\mathcal{S}}\}_{j\in\mathcal{S}(n)}, \quad (6)$$

$$E(n) = \frac{1}{K}\sum_{i=K}\sum_{\mathbf{F}_i(n)\in\mathcal{S}'(n)} \alpha_i \mathbf{F}_i(n) \quad (7)$$

where $E(n) \in \mathbb{R}^{d_E}$ is the learned embedding of node n, LeakyReLU is the activation function, $||$ denotes the concatenation operation, u is the attention parameter, and K is the number of attention heads. Here, $\mathbf{F}(n)$ and $\mathbf{F}_j^{\mathcal{S}} = \mathbf{F}_{\mathcal{A}_j}(\mathcal{S}_{\mathcal{A}_j})$ are computed by Eq. (2) and Eq. (4), respectively.

B. Citation Cascading and Author Aggregation

The second part of HDGNN is to model the cascading behavior of papers/authors. Here we consider each paper p as an independent entity. Recall that $p.t_0$ is the publication time, $p.t_r$ is the reference time, $\{(p_j, p.t_j)\}_j$ is the set of citation papers of p published at time $p.t_j$ during the observation window $[p.t_0, p.t_r](p.t_j \leq p.t_r)$. Since we already obtained the embeddings of papers $E(p)$, authors $E(a)$, and venues $E(v)$ (cf. Eq. (7)), we now separately model authors of a paper and the paper itself by feeding them into RNNs.

1) Multi-author aggregation layer: Note that each citing paper p_j of the original paper p, may contain multiple authors (in our dataset the mean number of authors per paper is 3.438 and the max number is 25). We sequentially pipeline the author embeddings into a GRU and then use the last hidden state $\mathbf{h}_{p_j}^a$ as the representation of p_j's authors.

2) Sequential citation aggregation layer: After author aggregation, for each paper p_j, we have its own embedding $E(p_j)$, the corresponding venue embedding $E(v_j)$, and the aggregated author embeddings $E(\mathbf{a}_j) = \mathbf{h}_{p_j}^a$. We then use a two-layer Bi-GRU to sequentially aggregate the citing papers ordered by their publishing time t_j, where each citing paper p_j is modeled as the combination of paper, authors, and venue. The rationalè is that we expect to capture temporal dependencies among citing papers, which, as we will show in the experiments, is superior to other aggregators such as sum or max pooling [25]. The overall architecture of the citation aggregation is:

$$\mathbf{E}(p_j) = (E(p_j)||E(\mathbf{a}_j)||E(v_j)), \quad (8)$$

$$\mathbf{h}_j^1 = (\overrightarrow{\text{GRU}}(\mathbf{E}(p_j))||\overleftarrow{\text{GRU}}(\mathbf{E}(p_j))), \quad (9)$$

$$\mathbf{h}_j^2 = (\overrightarrow{\text{GRU}}(\mathbf{h}_i^1)||\overleftarrow{\text{GRU}}(\mathbf{h}_i^1)) \quad (10)$$

where $\mathbf{h}_j^2 \in \mathbb{R}^{d_{\mathbf{h}^2}}$ is the j-th hidden state of the second layer of Bi-GRU. Here we concatenate the last hidden state of Bi-GRU as the final output representation of paper, and then make use of it to predict the scientific impact of p.

3) Output and model training: The output of HDGNN is the predicted citation number $\hat{c}_p^{t_p}$ of a paper p. We use two-layer of MLPs. The training losses of graph representation and impact prediction are respectively defined as:

$$\mathcal{L}_1(\Theta_1) = \arg\max_{\Theta_1} \prod_{n\in\mathcal{V}}\prod_{\mathcal{A}_n}\prod_{n_c\in\mathcal{N}_c} \Pr(n_c|n;\Theta_1), \quad (11)$$

$$\mathcal{L}_2(\Theta_2) = \frac{1}{N_T}\sum_{i=1}^{N_T}(\hat{c}_{p_i}^{t_p} - c_{p_i}^{t_p})^2, \quad (12)$$

where $\mathcal{N}_c(n)$ is the type-based neighboring set of node n, $\Pr(n_c|n;\Theta_1)$ is the conditional probability, N_T is the number of training samples, and $\hat{c}_{p_i}^{t_p}$ is the predicted number of citations of paper p_i at time t_p.

As for scientific impact prediction for scholars, the general training process is similar, except that Eq. (12) is alternatively defined as: $\mathcal{L}_2(\Theta_2) = \frac{1}{M_T}\sum_{i=1}^{M_T}(\hat{c}_{a_i}^{t_p} - c_{a_i}^{t_p})^2$,where $\hat{c}_{a_i}^{t_p}$ is the predicted number of citations for author a_i.

IV. Experiments

We evaluate HDGNN and several baselines on two scientific impact predictions – paper and author citations, respectively.

A. Dataset

The evaluations were performed on American Physical Society dataset (https://journals.aps.org/datasets). The APS dataset contains over 422K academic papers on 17 venues and 54M citations among papers between 1893 and 2017. The constructed heterogeneous graph of APS contains 616,316 papers, 430,950 authors, and 17 venues. For edges we have: author *writes* paper (2.9M), author *collaborates with* author (2.8M), author *publishes in* venue (0.6M), author *cites* paper (20.5M), paper *cites* paper (7.3M), paper *cites* author (4.8M), paper *is published in* venue (0.6M). For papers (authors) in the dataset, we select 20 years as the prediction time $p.t_p$ ($a.t_p$). Thus, we only consider the papers published before 1997, to ensure that each paper has at least 20 years to grow its citations. In the same way, selected authors are required to start their research career no later than 1997. We set reference time to 2 years and we note that papers/authors whose citations are less than 10 during the observation window are filtered out. The settings of prediction for authors are as the same as that for papers. After the preprocessing, we have a total of 11,475 papers and 14,318 authors. We use 50% of them for training, 25% for validation and the rest 25% for testing. Fig. 2 shows the statistics of APS dataset.

B. Baselines

The baselines used for comparison include feature-oriented and graph embedding models, as well as the state-of-the-art information cascade popularity prediction models.

- **Uniform** – for all papers/authors, we always predict their impact as a fixed number, uniformly searched from the minimum $\log c_{p/a}^{t_p}$ to maximum $\log c_{p/a}^{t_p}$ with a step of 0.001.
- **Features** – are fed into a linear regression model: observed citations $c_{p/a}^{t_r}$, mean arrival time, and degrees of

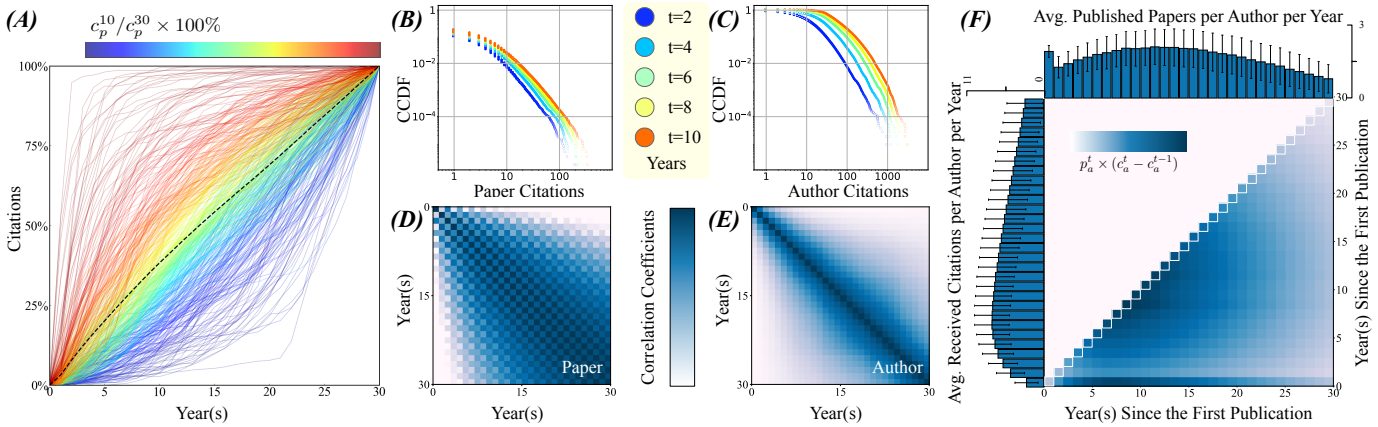


Fig. 2. Dataset statistics: ***(A)***: 509 papers with more than 200 citations after 30 years since publication before 1987. Lines represent normalized citation growth trends, line colors indicate citation rank of papers at the tenth year, i.e., $c_a^{10}/c_a^{30} \times 100\%$; dashed line denote the mean values. ***(B)*** and ***(C)***: Complementary cumulative distribution function (CCDF) of paper citations and author citations, respectively. ***(D)*** and ***(E)***: Pearson correlation coefficients of paper citations and author citations over 30 years, respectively. The (i, j) block in heatmap represents the correlation between i-th year's and j-th year's cumulative citations for all papers/authors. ***(F)***: The value of the i-th diagonal block of the heatmap is $p_a^t \times (c_a^t - c_a^{t-1})$, i.e., the average number of papers each author published at the i-th year multiplied by the average number of citations each author received at the i-th year; Top histogram: average published papers per author per year, Left histogram: average citations authors received per year (errorbars are standard deviations).

TABLE I
PERFORMANCE COMPARISON: PREDICTION FOR PAPERS/AUTHORS.

Model	Papers		Authors	
	MSLE	ACC	MSLE	ACC
Uniform	0.588	49.70%	1.102	36.37%
Features-c^{t_r}	0.401	58.35%	0.939	39.16%
Features	0.361	58.77%	0.832	40.19%
DeepCas	0.349	58.22%	0.787	41.45%
DeepHawkes	0.328	59.91%	0.725	42.38%
CasCN	0.310	61.71%	0.692	44.03%
HDGNN	**0.268**	**69.77%**	**0.590**	**51.62%**
(improves)	↑13.55%	↑13.06%	↑14.74%	↑17.24%

nodes. We use observed citation $c_{p/a}^{t_r}$ (i.e., Feature-c^{t_r}) as a simple baseline.

- **DeepCas** [12] – is a deep learning based prediction model utilizing DeepWalk for graph embedding and RNNs for cascade modeling and predicting.
- **DeepHawkes** [11] – makes use of Hawkes point process and neural networks for microblog/paper cascade prediction.
- **CasCN** [16] – utilizes GCN [17] and LSTM to model the structural and temporal information of cascades.

C. Variants of HDGNN

In order to compare other graph representation frameworks with our proposed HDGNN, we select following 10 models to replace the first part of HDGNN as variants, including homogeneous or heterogeneous methods, skip-gram based or matrix factorization based methods: **DeepWalk** [13], **LINE** [26], **metapath2vec** [27], **ProNE** [28], together with graph neural network **GraphSAGE** [25] and **HetGNN** [20]. Besides, we substitute the RNN aggregator with max pooling or sum pooling as two additional variants, denoted as **HDGNN-MaxPooling** and **HDGNN-SumPooling**. To evaluate the impact of author/venue embeddings, we separately remove the author part or venue part in Eq. (8) as **HDGNN-NoAuthor** and **HDGNN-NoVenue**.

D. Metrics

We use two widely used evaluation metrics [10], [11], [29], i.e., mean square logarithmic error and accuracy:

- MSLE: $\frac{1}{N_t}\sum_{i=1}^{N_t}(\log \hat{c}_{p_i/a_i}^{t_p} - \log c_{p_i/a_i}^{t_p})^2$
- ACC: $\frac{1}{N_t}\sum_{i=1}^{N_t}\mathbb{1}(0.5 * c_{p_i/a_i}^{t_p} \leq \hat{c}_{p_i/a_i}^{t_p} \leq 1.5 * c_{p_i/a_i}^{t_p})$

where $\mathbb{1}(\cdot)$ is the indicator function, N_t is the test sample size.

E. Experimental settings

For all graph representation baselines, we set the embedding dimension to 128. Random walk restart probability q is 0.5, walk length is 30, and number of walks for each node equals to 5. For type specific parameters $D^{\alpha}(*), D^{\beta}(*), D^{\gamma}(*)$, we use node in-degree and edge weights as a proxy of node influence. For HDGNN and its variants, the learning rate is chosen from $\{1, 10^{-1}, \ldots, 10^{-5}\}$, and the node embedding size is 128. The length of citation sequence of all methods (whether RNN, LSTM or GRU) is set to 100 – i.e., the max number of citation sequences. For papers/authors whose length is more than 100, we only select their first 100 citations (as for author sequence, the length of RNN is set to 6). The units are set to 128 and 64 in two-layer Bi-RNNs, and to 64 and 32 in two-layer MLPs. For feature aggregation RNNs, we use paper title embeddings pre-trained via BERT [30], and node embeddings pre-trained via DeepWalk. All the other hyper-parameters of baselines are set to their default values. Performance results are reported with early stopping on validation loss of 10 epoch patience.

F. Prediction Performance

We show the performance of all the models in Table I, and we observe that:

(1) HDGNN outperforms all the other methods in both paper and author impact prediction. This result demonstrates the effectiveness of learning interactions among heterogeneous nodes with the proposed heterogeneous information aggregation, which can be further verified by the fact that both feature-based models and homogeneous cascade prediction methods do not show comparable performance. Previous popularity prediction methods, e.g., DeepCas, DeepHawkes and CasCN, do not distinguish the type of nodes and therefore fail to model their complex and meaningful interactions.

(2) Author impact prediction is much harder than that of papers. As shown in ***(B)-(E)*** in Fig. 2, the citation number of authors is higher than that of papers by orders of magnitude, as well as the coefficients of correlation between observed and future citations. In fact, in settings of two year observation, the proportion of average observed citations c_p^2 to c_p^{20} is about 9.1% for authors. In contrast, the proportion for papers is 34.6% (cf. ***(A)*** in Fig. 2), which explains why prediction for authors' impact is more difficult – i.e., largely due to insufficient observations and enormous variability in scholars' productivity [9] (cf. ***(F)*** in Fig. 2). In addition, paper citation is strongly correlated to the factors such as the citations a paper has gained and the importance of publication venue (e.g., journal impact factor), which can be easily modeled in the graph with node attributes. In contrast, scholars' impact is far more unstable due to implicit factors such as funding scheme, tenure, gender issues – all of which need to be quantified with external high-resolution data repositories.

G. Qualitative Analysis

Fig. 3 shows the prediction results on 8 representative journals – the lower the MSLE and/or the higher the ACC, the better performance. The performance of HDGNN varies significantly on different publication venues – this is natural since venue is a strong indicator for future impact accumulation. In addition, we found that the prediction accuracy is affected by the citation distribution of papers in a journal. For example, the standard deviation of 20 year citations of papers (i.e., c_p^{20}) on Rev. Mode. Phys. is very high (255.02), whereas the value on Phys. Rev. is significantly less (43.24). This discrepancy also reveals why prediction of papers on Rev. Mode. Phys. is more difficult.

Fig. 4 plots the latent space learned in HDGNN, where we can observe clear clustering phenomena of author/paper embeddings from ***(A)*** and ***(C)***. It appears that papers published in the same journal tend to cluster together, which also indicates publication venue is an important indicator for scientific impact prediction. In addition, we also visualize a "crowd effect" of high impact papers/authors, as shown in ***(B)*** and ***(D)***. This also implies strong correlations among the latent representations of high impact scholars and papers. Another interesting result can be visualized is the gradually decaying color of the paper/author citations, implying that heavy-tailed

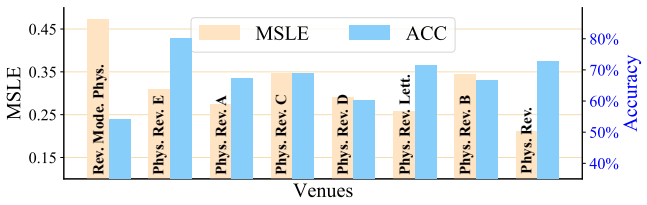
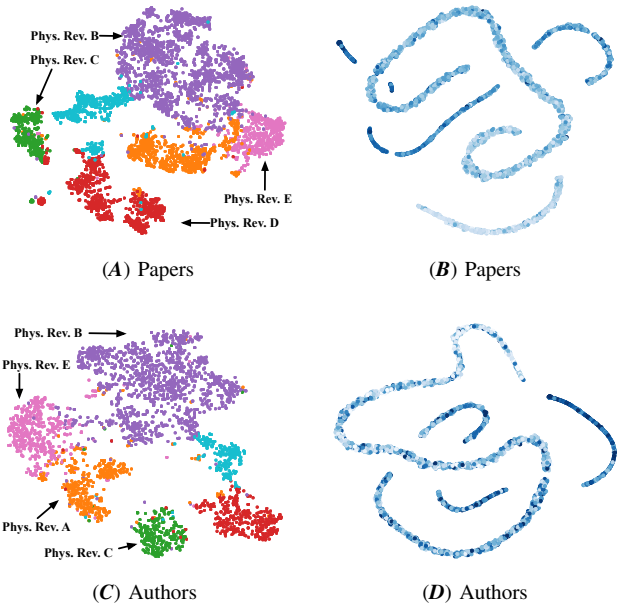


Fig. 3. Performance of HDGNN on 8 representative venues.



(***A***) Papers (***B***) Papers

(***C***) Authors (***D***) Authors

Fig. 4. Plot latent space of 6 venues after mapping to 2D with t-SNE (best viewed in color). (***A***) and (***C***) show paper/author embeddings retrieved from the heterogeneous graph representation (i.e., $E(n)$, cf. Section III-A) – colors are specified by venues; (***B***) and (***D***) plot paper/author citation embeddings from the prediction layer (i.e., $\mathbf{h}^2$, cf. Section III-B) – colors are specified by magnitude of citations.

distribution of scientific impact is successfully (to some extent at least) encoded in our model. It could also explain why our dynamic heterogeneous neighboring aggregation with a weighted contextualized node selection strategy substantially outperforms other homogeneous and heterogeneous graph embeddings.

H. Ablation Study

We now investigate the effect of important modules in HDGNN. Firstly, as shown in Table II, the information aggregation mechanism used in HDGNN is better than other graph embedding models including two heterogeneous embedding methods, i.e., metapath2vec and HetGNN – because of the more complex relations considered in our model and the benefit of considering temporal dependencies between citation sequences and/or author sequences. For example, HDGNN models 7 types of relations among nodes, whereas HetGNN, in contrast, only considers 3 edge types.

Additionally, the venue plays a vital role in predicting the impact of an author or a paper. This is demonstrated by the significant performance degradation after removing

TABLE II
ABLATION STUDY: PREDICTION FOR PAPERS/AUTHORS.

Model	Papers		Authors	
	MSLE	ACC	MSLE	ACC
DeepWalk	0.288	68.89%	0.627	49.09%
LINE	0.281	68.70%	0.614	47.88%
metapath2vec	0.294	66.38%	0.642	49.72%
GraphSAGE	0.309	64.97%	0.675	48.46%
HetGNN	0.292	65.79%	0.607	51.06%
ProNE	0.297	66.60%	0.635	47.65%
HDGNN-MaxPooling	0.358	64.22%	0.810	42.71%
HDGNN-SumPooling	0.280	69.90%	0.749	44.10%
HDGNN-NoAuthor	0.279	69.05%	0.605	50.00%
HDGNN-NoVenue	0.290	67.10%	0.651	48.26%
HDGNN	0.268	69.77%	0.590	51.62%

venue embeddings in Eq. (8). Authorship, surprisingly, is less important than the journal that a paper published in, though masking the authorship information may slightly degrade the prediction performance. As for aggregation choices, both max pooling and sum pooling are inferior to the RNN aggregator used in HDGNN, due to their lack of sequential dependencies.

V. CONCLUSION

We introduced HDGNN approach for effectively quantifying and predicting the scientific impact of scholars and research publications, by bridging the dynamic processes of impact evolution and complex nodes interactions. We presented an efficient network sampling method with the consideration of rich node relations and a temporally attentive neighbor aggregation network to model the complex and accumulating dynamic processes of scientific impact. Evaluations on a real-world scientific dataset demonstrated the superior performance HDGNN in comparison to several state-of-the-art baselines. Future work will investigate the impact of cross-institutional collaboration on citations.

ACKNOWLEDGEMENT

This work was supported by the National Natural Science of China under Grant No.61602097 and No.61472064, and NSF grants CNS 1646107.

REFERENCES

[1] S. Fortunato, C. T. Bergstrom, K. Börner, J. A. Evans, D. Helbing, S. Milojević, A. M. Petersen, F. Radicchi, R. Sinatra, B. Uzzi *et al.*, "Science of science," *Science*, vol. 359, no. 6379, 2018.

[2] R. Sinatra, D. Wang, P. Deville, C. Song, and A.-L. Barabási, "Quantifying the evolution of individual scientific impact," *Science*, vol. 354, no. 6312, p. aaf5239, 2016.

[3] Q. Ke, E. Ferrara, F. Radicchi, and A. Flammini, "Defining and identifying sleeping beauties in science," *PNAS*, vol. 112, no. 24, pp. 7426–7431, 2015.

[4] D. J. D. S. Price, "Networks of scientific papers," *Science*, 1965.

[5] R. Yan, J. Tang, X. Liu, D. Shan, and X. Li, "Citation count prediction: learning to estimate future citations for literature," in *CIKM*, 2011.

[6] D. E. Acuna, S. Allesina, and K. P. Kording, "Predicting scientific success," *Nature*, vol. 489, no. 7415, pp. 201–202, 2012.

[7] D. Wang, C. Song, and A.-L. Barabási, "Quantifying long-term scientific impact," *Science*, vol. 342, no. 6154, pp. 127–132, 2013.

[8] Y. Dong, R. A. Johnson, and N. V. Chawla, "Can scientific impact be predicted?" *IEEE Trans. Big Data*, vol. 2, no. 1, pp. 18–30, 2016.

[9] A. Clauset, D. B. Larremore, and R. Sinatra, "Data-driven predictions in the science of science," *Science*, vol. 355, no. 6324, 2017.

[10] H. Shen, D. Wang, C. Song, and A.-L. Barabási, "Modeling and predicting popularity dynamics via reinforced poisson processes," in *AAAI*, 2014, pp. 291–297.

[11] Q. Cao, H. Shen, K. Cen, W. Ouyang, and X. Cheng, "Deephawkes: Bridging the gap between prediction and understanding of information cascades," in *CIKM*, 2017, pp. 1149–1158.

[12] C. Li, J. Ma, X. Guo, and Q. Mei, "Deepcas: An end-to-end predictor of information cascades," in *WWW*, 2017, pp. 577–586.

[13] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *KDD*, 2014, pp. 701–710.

[14] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv:1412.3555*, pp. 1–9, 2014.

[15] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, "Graph attention networks," in *ICLR*, 2018, pp. 1–12.

[16] X. Chen, F. Zhou, K. Zhang, G. Trajcevski, T. Zhong, and F. Zhang, "Information diffusion prediction via recurrent cascades convolution," in *ICDE*, 2019, pp. 770–781.

[17] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017, pp. 1–14.

[18] R. Trivedi, M. Farajtabar, P. Biswal, and H. Zha, "Dyrep: Learning representations over dynamic graphs," in *ICLR*, 2019, pp. 1–25.

[19] F. Manessi, A. Rozza, and M. Manzo, "Dynamic graph convolutional networks," *Pattern Recognition*, vol. 97, pp. 1–18, 2020.

[20] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *KDD*, 2019, pp. 793–803.

[21] Y. Lu, C. Shi, L. Hu, and Z. Liu, "Relation structure-aware heterogeneous information network embedding," in *AAAI*, 2019, pp. 4456–4463.

[22] Y. Shi, Q. Zhu, F. Guo, C. Zhang, and J. Han, "Easing embedding learning by comprehensive transcription of heterogeneous information networks," in *KDD*, 2018, pp. 2190–2199.

[23] H. Tong, C. Faloutsos, and J.-Y. Pan, "Fast random walk with restart and its applications," in *ICDM*, 2006, pp. 613–622.

[24] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.

[25] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *NIPS*, 2017, pp. 1024–1034.

[26] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *WWW*, 2015, pp. 1067–1077.

[27] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *KDD*, 2017.

[28] J. Zhang, Y. Dong, Y. Wang, J. Tang, and M. Ding, "Prone: fast and scalable network representation learning," in *IJCAI*, 2019.

[29] Q. Zhao, M. A. Erdogdu, H. Y. He, A. Rajaraman, and J. Leskovec, "Seismic: A self-exciting point process model for predicting tweet popularity," in *KDD*, 2015, pp. 1513–1522.

[30] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL-HLT*, 2019, pp. 4171–4186.