Revisiting Citation Prediction with Cluster-Aware Text-Enhanced Heterogeneous Graph Neural Networks

Carl Yang
Department of Computer Science
Emory University
Atlanta, United States
j.carlyang@emory.edu

Jiawei Han
Department of Computer Science
University of Illinois, Urbana Champaign
Urbana, United States
hanj@illinois.edu

***Abstract*—Numerous papers get published all the time. However, some papers are born to be well-cited while others are not. In this work, we revisit the important problem of citation prediction, by focusing on the important yet realistic prediction on the average number of citations a paper will attract per year. The task is nonetheless challenging because many correlated factors underlie the potential impact of a paper, such as the prestige of its authors, the authority of its publishing venue, and the significance of the problems/techniques/applications it studies. To jointly model these factors, we propose to construct a heterogeneous publication network of nodes including papers, authors, venues, and terms. Moreover, we devise a novel heterogeneous graph neural network (HGN) to jointly embed all types of nodes and links, towards the modeling of research impact and its propagation. Beyond graph heterogeneity, we find it also important to consider the latent research domains, because the same nodes can have different impacts within different communities. Therefore, we further devise a novel cluster-aware (CA) module, which models all nodes and their interactions under the proper contexts of research domains. Finally, to exploit the information-rich texts associated with papers, we devise a novel text-enhancing (TE) module for automatic quality term mining. With the real-world publication data of DBLP, we construct three different networks and conduct comprehensive experiments to evaluate our proposed CATE-HGN framework, against various state-of-the-art models. Rich quantitative results and qualitative case studies demonstrate the superiority of CATE-HGN in citation prediction on publication networks, and indicate its general advantages in various relevant downstream tasks on text-rich heterogeneous networks.**

***Index Terms*—conditional network embedding, hierarchical network embedding, generative adversarial networks**

I. Introduction

420K newly published papers have been recorded in DBLP in 2020[1], and more than 1M papers are poured into the PubMed database each year[2]. With such ever-growing volume of scientific publications, researchers are overwhelmed to pick out the potentially impactful papers to read, reviewers struggle to find the likely significant papers to publish, and even the authors misestimate which of their own papers can fly. However, some papers are born to be impactful (*i.e.*, well-cited), while others are not. Is there any pattern underlying the citations of papers, and can we predict, *e.g.*, the average number of citations a paper can get each year, upon or even before its publication?

Although whether a paper will be highly cited largely depends on the importance of the research problem itself as well as the novelty and thoroughness of the solution, these factors are often reflected by the authors' experience and venues' reputation towards specific topics. Based on such insight, existing research on citation prediction has achieve the consensus on the modeling of authors, venues as well as textual contents of papers. However, in terms of methods, they heavily rely on manual feature engineering, by constructing a list of features for each paper such as number of publications of authors and venues, publication type, author's h-index, venue's impact. Then they brutally apply predictive models like linear regression and classification trees [1], [2], [3], [4]. Besides the time-consuming feature designing process and separation from predictive model training, features constructed in such ways are neither systematic nor comprehensive regarding the capturing of certain factors hidden in the complex interactions among papers, authors, venues, *etc.*, and many of the designed features are not universally available across different datasets.

Motivated by the recent success of heterogeneous network representation learning [5], [6], in this work, we aim to revisit the important yet challenging problem of citation prediction and develop a framework that is effort-less in data preprocessing yet powerful in capturing the possibly complex interactions among papers, authors, venues and textual contents. First, we leverage the ubiquitous data structure of heterogeneous networks [7] to jointly model different types of nodes and their interaction links upon data availability with minimum data preprocessing[3]. Furthermore, we leverage graph neural networks (GNN) [9], [10], [11] for citation prediction, to free human labors from exhaustive feature engineering by automatically learning the features and predicting the citations in an end-to-end fashion.

Unfortunately, existing GNNs on heterogeneous networks cannot properly handle our citation prediction problem, because they mainly focus on the embedding of certain types of

[1]https://dblp.org/xml/, https://dblp.org/statistics/publicationsperyear.html
[2]https://www.nature.com/articles/nj7612-457a

[3]To fairly compare with existing models which cannot automatically mine important terms from textual contents, we follow [8] to also extract papers' keywords, but even this process is not needed for our proposed framework.

target nodes (*e.g.*, authors), and only use the other nodes as contexts to construct different message propagation channels [12], [13], [14], [15], [16], [17], [18]. For citation prediction, however, it is critical to not only model the research impact of papers, but also that of authors, venues, and terms. Inspired by RankClus [19], we design a novel heterogeneous graph neural network (HGN) to iteratively estimate the research impacts between target and context nodes based on their complex interactions regarding different types of links through one-space semi-supervised node representation learning (Sec III-C).

Our HGN effectively estimates the general research impact of all types of nodes in the heterogeneous publication network, but in reality, each node might only be impactful (thus attract citations) in a certain research domain or community. To this end, we devise a novel cluster-aware module (CA) to jointly infer the latent research communities and domain-specific impacts of all nodes. The idea is to iteratively learn a node clustering assignment through self-training [20], [21], while predicting citations only in the masked domain-specific embedding spaces. To enhance clustering quality, we further add two regularizers for cross-layer cluster consistency and cross-cluster embedding disparity (Sec III-D).

Although we extract terms from keywords of papers to allow general heterogeneous GNNs to get direct access to the textual contents of papers, terms are different from authors and venues as they are often not directly available and accurate. To this end, we design a novel term-enhancing module (TE) to automatically mine quality terms from the raw textual contents of papers. Since keywords have varying quality, we leverage a pre-trained BERT language model [22] to bootstrap semantically relevant terms from the weak supervision of simple research domain names, and use TF-IDF [23] to reconstruct the paper-term links. We seamlessly integrate the TE and CA modules through term-enhanced clustering initialization and cluster-enhanced adaptive term mining (Sec III-E).

Through extensive experiments on three heterogeneous publication networks constructed from the DBLP data[1], we comprehensively evaluate the performance of our CATE-HGN framework and each of its novel components on the citation prediction task. Rich quantitative results and qualitative case studies demonstrate the superiority of CATE-HGN over various traditional citation prediction methods and state-of-the-art heterogeneous GNNs (Sec IV).

II. Problem Statement

We aim to predict the impact of research papers regarding how many citations they will get after being published. Since it is unrealistic to predict the exact number of citations and such numbers can constantly change over time, we simplify the problem into static regression by focusing on the average number of citations per year. We will discuss the possibility of future works on dynamic citation prediction in Section III-G.

Below we formulate the unique challenges of citation prediction.

1) The impact of research papers can depend on the prestige of authors, authority of publishing venues, and the significance of studied problems/techniques/applications (described by textual terms such as keywords). Unfortunately, none of such information is directly available.
2) Beyond the heterogeneous factors, each author/venue/term might be impactful only in certain domains, but again, such clustering of domains is also unknown.
3) In reality, while authors and venues are explicitly specified for papers, terms are not always available or accurate, which prevents direct analysis towards the actual problems, techniques or applications studied in the papers.
4) The above factors can interact in complex ways, so none of them should be considered in isolation.

III. CATE-HGN

A. Preliminaries

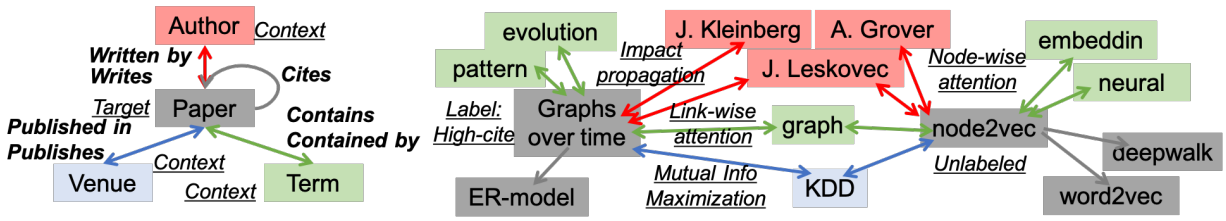
Definition 3.1: A *heterogeneous network* [7] is a network $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ with multiple types of objects and links. Within $\mathcal{G}$, $\mathcal{V}$ is the set of objects, where each object $v \in \mathcal{V}$ is associated with an object type $\phi(v)$, and $\mathcal{E}$ is the set of links, where each link $e \in \mathcal{E}$ is associated with a link type $\psi(e)$.

Heterogeneous networks have been widely used to model real-world multi-typed multi-relational nodes. Since we want to explicitly model the impact of different nodes on citations, it is natural for us to resort to this powerful and general data model. Figure 1 (a) shows the schema of the heterogeneous publication network we construct and model in this work. As we can see, in our schema, there is a single type of link between each pair of nodes, which means the node types on both ends uniquely determine the link type. This can simplify the presentation of our model, but the model is not restricted to such a setting. Moreover, beyond Definition 3.1, we also model the link weights in $\mathcal{G}$ using a tabular function $\omega \in \mathcal{E} \rightarrow \mathbb{R}$, where $\omega(e)$ is the link weight of e. We consider the two directions of each link as two different types, except for the citing links between papers, so as to avoid label leakage.

Graph neural networks: One of the most phenomenal works on graph neural network (GNN) is the original graph convolutional networks (GCN) [9]. To recapitulate its main design, we repeat the typical output of the $(l+1)$-th convolutional layer $\mathbf{H}^{(l+1)}$ of GCN as follows

$$\mathbf{H}^{(l+1)} = \sigma\Big(\tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}\mathbf{H}^{(l)}\mathbf{W}^{(l)}\Big), \tag{1}$$

where $\tilde{\mathbf{A}}$ is the adjacency matrix with self-connections of the whole graph with N nodes, $\tilde{\mathbf{D}}_{ii} = \sum_j \tilde{\mathbf{A}}_{ij}$, $\mathbf{W}^{(l)}$ is the trainable layer-wise weight matrix, and $\sigma(\cdot)$ is a nonlinear activation function such as ReLU. $\mathbf{H}^{(l)} \in \mathbb{R}^{N \times D_l}$ is the output of the l-th layer, with $\mathbf{H}^{(0)} = \mathbf{X}$, *i.e.*, the original node features. Most GNNs on heterogeneous networks are based on GCN, where different link-type-aware and meta-path-aware propagation channels (*e.g.*, with exclusive weight matrices or other embedding transformation/aggregation functions) have been incorporated into Eq. (1) to enable semantic-aware convolutions.



(a) Network schema

(b) Toy example and main techniques of HGN

Fig. 1. **The schema and toy example of real-world heterogeneous publication network.**

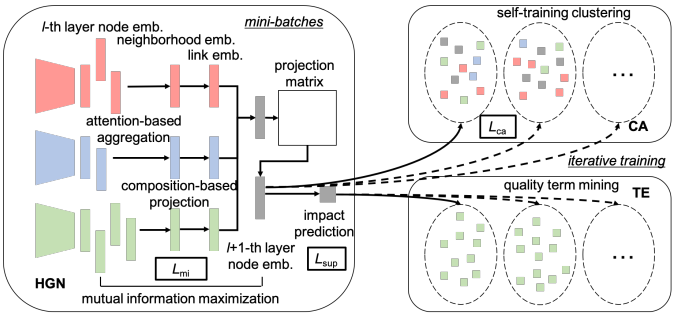


Fig. 2. **The overall architecture of CATE-HGN.**

B. Framework Overview

In this paper, we propose a novel framework called CATE-HGN for citation prediction on text-rich heterogeneous publication networks, which consists of three modules of HGN, CA and TE, as summarized in Figure 2. A novel one-space heterogeneous graph network (HGN) is designed to jointly embed heterogeneous nodes and links in a single space with composition operations (Section III-C.1); the HGN is then optimized in a semi-supervised fashion with an explicit unsupervised loss based on cross-type mutual information maximization to align and smooth the embedding space (Section III-C.2) and is equipped with three-way attentions for type-aware selective neighbor aggregation (Section III-C.3). A cluster-aware (CA) module is proposed to be iteratively trained with HGN for research domain discovery, and a text-enhancing (TE) module is devised to adaptively update the set of quality terms based on the output of HGN during the training of HGN and CA.

C. One-Space Heterogeneous Graph Network

Figure 1 (b) gives an example of the heterogeneous network constructed from real-world publication data. As formulated in Section II, we use the citation numbers of papers published before a certain year as the ground-truth, and the task is to predict the citation numbers of the papers published later. By definition, this is a graph-based semi-supervised learning problem [24], [25], which can be solved via jointly optimizing a supervised loss and an unsupervised loss as follows

$$\mathcal{L}_{hgn} = \mathcal{L}_{sup} + \lambda \mathcal{L}_{unsup}. \tag{2}$$

Recently, many GNNs have been designed for such graph-based semi-supervised learning tasks [9], [10], [11]. However, most existing GNNs on heterogeneous networks focus on the representation learning of one particular type of nodes (*i.e.*, target nodes) and only use the other types of nodes (*i.e.*, context nodes) to model the semantically different interactions among the target nodes [12], [13], [14], [15], [16], [17]. In our setting of citation prediction, it is important to also accurately embed the context nodes in the same space of target nodes, so as to jointly model their individual impacts towards paper citations (*e.g.*, the prestige of authors, the authority of venues).

1) Joint node and link embedding with compositions: Inspired by the idea of iteratively estimating the importance of target and context nodes in heterogeneous networks from RankClus [19], [8], we design a novel HGN to propagate the impact estimation step-by-step among all types of nodes according to different types of direct links. In other words, starting from the labeled papers like Graphs over time in Figure 1 (b), we do not use different meta-paths like paper-author-paper or paper-venue-paper to directly infer the citation of unlabeled papers like node2vec; instead, we firstly infer the impact of its direct neighbors, *i.e.*, the prestige of authors like Jure Leskovec, and the authority of venues like KDD, and then iteratively infer the impact of node2vec and so on.

The need of jointly embedding all nodes of different types in the same space while accounting for different types of links closely resembles that of knowledge graph embedding (KGE) [26], [27], [28]. To leverage the power of GNNs in a KGE setting while addressing the well-known over-parameterization problem in models like R-GCN [12], we borrow the ideas of entity-relation composition [26], [29]. Particularly, we rewrite the convolution function in Eq. (1) as

$$\mathbf{h}_v^{(l+1)} = \sigma\Big(\sum_{(u,e)\in\mathcal{N}(v)} \mathbf{W}_a^{(l)} \big(\varphi^{(l)}(\mathbf{h}_u^{(l)}, \mathbf{h}_e^{(l)}) \odot \mathbf{h}_v^{(l)}\big) \Big), \tag{3}$$

where $\mathcal{N}(v)$ is the set of direct neighbors of node v and the connecting link; $\mathbf{h}_u^{(l)}$ and $\mathbf{h}_e^{(l)}$ are the node embedding of u and edge embedding of e in the l-th HGN layer, respectively. $\varphi^{(l)} : \mathbb{R}^{d_l} \times \mathbb{R}^{d_l} \to \mathbb{R}^{d_l}$ is a composition operation, which can be simple non-parameterized operations such as subtraction [26], multiplication [27] and circular-correlation [28]; $\odot$ is the vector concatenation operation; $\mathbf{W}_a^{(l)} \in \mathbb{R}^{d_{l+1} \times 2d_l}$ is the learnable node embedding transformation matrix shared across all types of links in the l-th HGN layer. Similarly to the node embedding transformation in Eq. (3), for all links, we have

$$\mathbf{h}_e^{(l+1)} = \mathbf{W}_b^{(l)} \mathbf{h}_e^{(l)}, \tag{4}$$

where $\mathbf{W}_b^{(l)} \in \mathbb{R}^{d_{l+1} \times d_l}$ is the learnable link embedding transformation matrix. The sharing of composition operation φ and embedding transformation matrices $\mathbf{W}_a, \mathbf{W}_b$ leads to effective reduction of learnable parameters compared with R-GCN [12], while the joint embedding of nodes and links still allows the HGN to distinguish different types of nodes and links through the integration of the following type-aware node and link encoders

$$\begin{aligned} \mathbf{h}_v^{(0)} &= \sigma\big(\mathbf{W}_{\phi(v)}^0 \mathbf{x}_v + \mathbf{b}_{\phi(v)}^{(0)}\big), \\ \mathbf{h}_e^{(0)} &= \sigma\big(\mathbf{W}_{\psi(e)}^0 \mathbf{x}_e + \mathbf{b}_{\psi(e)}^{(0)}\big), \end{aligned} \tag{5}$$

where $\mathbf{x}_v$ and $\mathbf{x}_e$ are the node features of v and link features of e, respectively. In our setting, we use aggregated pre-trained word embeddings as the node features, and randomly generate one feature vector to share across all links of each type.

To fully leverage the ground-truth citation numbers of labeled papers (average by year), we design our supervised loss on all HGN layers over the set of labeled papers $\mathcal{Y} \subset \mathcal{V}$ as follows

$$\mathcal{L}_{sup} = \sum_{l=1}^{L} \sum_{v \in \mathcal{Y}} ||y_v - \hat{y}_v^{(l)}||^2, \text{ with } \hat{y}_v^{(l)} = \mathbf{W}_y^{(l)} \mathbf{h}_v^{(l)} + \mathbf{b}_y^{(l)}. \tag{6}$$

We theoretically analyze the ability of HGN to approximate the message passing functions of any heterogeneous higher-order meta-paths in Theorem 3.1. Our proof follows the Theorem 2.1 in [30] and is detailed in the Appendix.

Theorem 3.1 (Modeling meta-paths with a composition of R Functions): For a heterogeneous network $\mathcal{G}$ defined in Definition 3.1 with R types of relations, we assume there is an oracle function $\hat{O}$ that takes in a target node v's meta-paths information $\mathcal{M}_v \in \mathbb{R}^d$ on $\mathcal{G}$, and outputs the v's ground-truth label $y_v \in \mathbb{R}^d$. When $\mathcal{M}_v$ is absolutely continuous with respect to the Lebesgue measure, for any given approximation error ε and R functions $\{F_r | r \in [R]\}$, there exists a composition function $Comp(\cdot|\{F_r|r \in [R]\}) : \mathbb{R}^d \to \mathbb{R}^d$, which is viewed as the gradient function of an FNN $u(\cdot) : \mathbb{R}^d \to \mathbb{R}$ with ReLU activation, of depth $L = [\log_2 n]$ and width $N = 2L$, where $n = O(\frac{1}{\varepsilon^d})$. For the 1-Wasserstein distance measurement $W_1(\cdot, \cdot)$, we have

$$\mathbb{E}_{\mathcal{M}_v \sim H}[W_1(\hat{O}(\mathcal{M}_v), Comp(\mathcal{M}_v|\{F_r|r \in [R]\})] < \varepsilon.$$

2) Embedding alignment with cross-type mutual information maximization: Under the supervision of labeled papers with citation classes, the HGN embeddings of target nodes (*i.e.*, papers) can directly capture the research impacts of papers. However, unlike semi-supervised learning on homogeneous networks, the cross-type embedding transformations with $\mathbf{W}_a$ and $\mathbf{W}_b$ may project the context nodes (*i.e.*, authors, venues, terms) into arbitrary locations in the embedding spaces, which are never directly supervised by citation labels and may not directly reflect the impact of their connected research papers as we care about. Moreover, the smoothness of embeddings on heterogeneous networks is not necessarily guaranteed due to the arbitrary learnable transformations, leading to large parameter variances that compromise stable model inference. To alleviate this limitation, we get inspired from recent research on deep mutual information (MI) maximization [31], we propose to properly regularize the embedding transformations in our HGN by maximizing the MI between each convolution layer.

In our setting, let $\mathbf{h}_{\mathcal{N}(v)}^{(l)}$ denote the l-th layer embeddings of the set of heterogeneous neighbors of v, which is the input of the $(l+1)$-th HGN layer at node v; $\mathbf{h}_v^{(l+1)}$ is the output. Following [31], the MI between $\mathbf{h}_v^{(l+1)}$ and $\mathbf{h}_{\mathcal{N}(v)}^{(l)}$ can be defined as

$$I(\mathbf{h}_v^{(l+1)}; \mathbf{h}_{\mathcal{N}(v)}^{(l)}) = \int_{H^{(l+1)}} \int_{H^{(l)}} p(\mathbf{h}_v^{(l+1)}, \mathbf{h}_{\mathcal{N}(v)}^{(l)}) \log \frac{p(\mathbf{h}_v^{(l+1)}, \mathbf{h}_{\mathcal{N}(v)}^{(l)})}{p(\mathbf{h}_v^{(l+1)}) p(\mathbf{h}_{\mathcal{N}(v)}^{(l)})} d\mathbf{h}_v^{(l+1)} d\mathbf{h}_{\mathcal{N}(v)}^{(l)}, \tag{7}$$

where $p(\mathbf{h}^{(l+1)})$ denotes the empirical probability distributions of node embeddings, and $p(\mathbf{h}_v^{(l+1)}, \mathbf{h}_{\mathcal{N}(v)}^{(l)})$ the joint distribution.

In our HGN setting, Eq. (7) is not directly optimizable, due to the variable types, orders and sizes of heterogeneous neighbors in $\mathcal{N}(v)$. Fortunately, we find the existing theory on MI decomposition [32] easily extensible to our setting, which allows us to decompose Eq. 7 as follows

$$I(\mathbf{h}_v^{(l+1)}; \mathbf{h}_{\mathcal{N}(v)}^{(l)}) = \sum_{t=1}^{T} \sum_{(u,e) \in \mathcal{N}_t(v)} \omega(e) I(\mathbf{h}_v^{(l+1)}; \mathbf{h}_u^{(l)}), \tag{8}$$

where $\omega(e)$ is the link weight of e.

Eq. (8) allows us to maximize $I(\mathbf{h}_v^{(l+1)}; \mathbf{h}_{\mathcal{N}(v)}^{(l)})$ according to the individual direct links of different types on the heterogeneous network, which enhances the correlations of embeddings between target and context nodes towards the encoding of research impact regarding paper citations. However, it still does not explicitly guarantee the smoothness of node embeddings across different types, *i.e.*, the embeddings of neighboring nodes on the heterogeneous network are correlated but can still be far away due to the embedding transformations. To alleviate this, we parameterize the link weights in Eq. (8) with neighboring node embeddings and explicitly enforces the learnable link weights to be close to the real link weights. Specifically, we rewrite Eq. (8) as follows

$$\begin{aligned} I(\mathbf{h}_v^{(l+1)}; \mathbf{h}_{\mathcal{N}(v)}^{(l)}) = &\sum_{t=1}^{T} \sum_{(u,e) \in \mathcal{N}_t(v)} \hat{\omega}(e) I(\mathbf{h}_v^{(l+1)}; \mathbf{h}_u^{(l)}) \\ &+ I(\hat{\omega}(e); \omega(e)), \\ \text{with } \hat{\omega}(e) = &\ \text{sigmoid}((\mathbf{h}_v^{(l+1)})^T \mathbf{h}_u^{(l)}). \end{aligned} \tag{9}$$

As we focus more on maximizing MI instead of obtaining its specific value, we employ the Jensen-Shannon MI estimator

(JSD) [33] to maximize the first term in Eq. (9), which is parameterized as follows

$$I(\mathbf{h}_v^{(l+1)};\mathbf{h}_u^{(l)}) = -\,\mathrm{sp}(-\mathcal{D}(\mathbf{h}_v^{(l+1)},\mathbf{h}_u^{(l)})) - \mathbb{E}[\mathrm{sp}(-\mathcal{D}(\mathbf{h}_v^{(l+1)},\mathbf{h}_{u'}^{(l)}))], \quad \text{with } \mathcal{D}(\mathbf{x}_i,\mathbf{x}_j)=\sigma(\mathbf{x}_i^T\mathbf{W}_d\mathbf{x}_j), \tag{10}$$

where u' is a negative sample from $\tilde{\mathbb{P}} = \mathbb{P}$, and $\mathrm{sp}(x) = \log(1+\exp(x))$ is the soft-plus function. For the second term in Eq. (9), since both $\hat{\omega}(e)$ and $\omega(e)$ are one-dim scalars, we replace the MI with a negative L2 loss, since minimizing the L2 loss contributes to maximizing the MI. Specifically, we have

$$I(\hat{\omega}(e);\omega(e)) = -(\hat{\omega}(e)-\omega(e))^2. \tag{11}$$

Finally, we define our unsupervised MI loss across all HGN layers over all nodes and links of different types as follows

$$\mathcal{L}_{unsup} = -\sum_{l=1}^{L-1}\sum_{v} I(\mathbf{h}_v^{(l+1)};\mathbf{h}_{\mathcal{N}(v)}^{(l)}). \tag{12}$$

3) Important node and link selection with three-way attentions: For a research paper and its particular list of authors, some authors may contribute more to its attraction of citations while others contribute less; moreover, the importance of authors, venues and terms may contribute differently across different research papers.

Motivated by the wide success of attention mechanisms in GNNs [11], we further apply node-wise and link-wise attentions, which are used to enable learnable selections towards important nodes within each type of neighbors and across different types of neighbors within each layer of our HGN. Specifically, the node-wise and link-wise attentions are implemented as

$$\mathbf{h}_v^{(l+1)} = \sigma\Big(\sum_{t=1}^{T}\alpha_b^{(l)}\sum_{(u,e)\in\mathcal{N}_t(v)}\alpha_t^{(l)}\mathbf{W}_a^{(l)}\varphi^{(l)}(\mathbf{h}_u^{(l)},\mathbf{h}_e^{(l)})\Big), \tag{13}$$

where t denotes the node type, and $\mathcal{N}_t$ is the set of neighbors of node type t.

In our setting, we assume a single type of link between two specific nodes, while the framework trivially generalizes to multi-typed links between the same two nodes. α_t is the node-wise attention weight among neighbors of the same node type t, which is computed as

$$\alpha_t(v,e,u) = \frac{\exp\Big(\tau\big(\mathbf{a}_t^T[\mathbf{h}_v\odot\mathbf{h}_e\odot\mathbf{h}_u]\big)\Big)}{\sum_{(u',e')\in\mathcal{N}_t(v)}\exp\Big(\tau\big(\mathbf{a}_t^T[\mathbf{h}_v\odot\mathbf{h}_{e'}\odot\mathbf{h}_{u'}]\big)\Big)}, \tag{14}$$

where $\mathbf{a}_t \in \mathbb{R}^{3d}$ is the learnable node-wise attention parameter; following [11], τ is a LeakyReLU activation function and $\odot$ is the vector concatenation operation. Similarly, α_b in Eq. (13) is the link-wise attention weight across neighbors of different node types, which is computed as

$$\alpha_b(v,t,n) = \frac{\exp\Big(\tau\big(\mathbf{a}_b^T[\mathbf{h}_v\odot\mathbf{h}_{evt}\odot\mathbf{h}_{nvt}]\big)\Big)}{\sum_{t'=1}^{T}\exp\Big(\tau\big(\mathbf{a}_b^T[\mathbf{h}_v\odot\mathbf{h}_{evt'}\odot\mathbf{h}_{nvt'}]\big)\Big)}, \tag{15}$$

where $\mathbf{a}_t \in \mathbb{R}^{3d}$ is the learnable link-wise attention parameter; since we assume a single type of link between two specific nodes, v and t together determine the type of link between them, and thus $\mathbf{h}_{evt}$ can be replaced by the edge embedding $\mathbf{h}_e$ accordingly; $\mathbf{h}_{nvt} = \sum_{(u,e)\in\mathcal{N}_t(v)}\alpha_t^{(l)}\mathbf{W}_a^{(l)}\varphi^{(l)}(\mathbf{h}_u^{(l)},\mathbf{h}_e^{(l)})$ is the aggregated embedding of v's neighbors of type t. In both Eq. (14) and (15), we omit the layer superscripts (l) for simplicity. Following [11], we also use multi-head attentions to further improve the representation power and stability of both node-wise and link-wise attentions, with numbers of heads as D_a and D_b, respectively.

D. Cluster-Aware Module

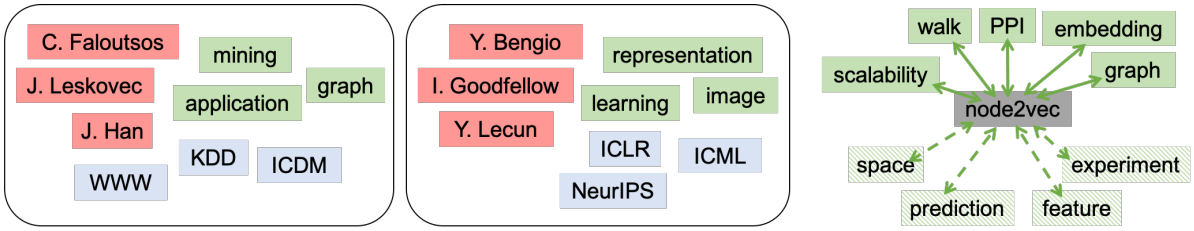
Our one-space HGN enables the embedding of all types of nodes and links in the heterogeneous network into the same space. However, as exemplified in Figure 3 (a), research impacts are often specific to research domains or communities, *i.e.*, a prestigious researcher may be more impactful in a particular domain (*e.g.*, Jiawei Han more impactful in data mining instead of machine learning), so as some venues/terms are only popular in certain domains but not the others. Thus, failure to capture the research domains or communities underlying different types of nodes can lead the lack of power in fine-grained modeling of research impacts.

To mitigate this limitation, we aim to jointly model the latent research domains and the impacts of all nodes. However, the task is non-trivial, mainly due to the lack of labeled data for research domains– aside from some best-known papers, authors, venues and terms, we hardly know the ground-truth domains of most other nodes, which makes the modeling of research domains essentially an unsupervised graph clustering (community detection) problem. To this end, we design a novel cluster-aware module on top of our HGN, which consists of self-training clustering, masked-embedding prediction, and cluster consistency-disparity regularizers.

1) Self-training clustering: Inspired by [21], we jointly learn node embedding and clustering through iterative training of HGN and CA, which only requires a single hyper-parameter K, *i.e.*, the number of clusters. Specifically, at each HGN layer l, we randomly initialize K trainable cluster centers $\{\xi_k^{(l)}\}_{k=1}^{K}$, where $\xi_k^{(l)} \in \mathbb{R}^{d_l}$. Then we compute a soft clustering assignment $\mathbf{Q}^{(l)}$ as follows

$$q_{vk}^{(l)} = \frac{(1+||\mathbf{h}_v^{(l)}-\xi_k^{(l)}||^2)^{-1}}{\sum_{k'}(1+||\mathbf{h}_v^{(l)}-\xi_{l'}^{(l)}||^2)^{-1}}, \tag{16}$$

where $q_{vk}^{(l)}$ is the probability of assigning node v to cluster k based on the distance between $\mathbf{h}_v^{(l)}$ and $\xi_k^{(l)}$. Since our HGN projects all types of nodes into the same embedding



(a) Research impacts should be modeled under the considerations of research domains

(b) Terms can have different qualities

Fig. 3. **Motivating real-world toy examples towards the CA and TE modules.**

space, the soft clustering can be done across all nodes in the heterogeneous network.

The idea of self-training is to make the model learn from the more confident results of itself [20]. Following [21], we use the auxiliary distribution $\mathbf{P}^{(l)}$ defined as follows

$$p_{vk}^{(l)} = \frac{(q_{vk}^{(l)})^2 / f_k^{(l)}}{\sum_{k'} (q_{vk'}^{(l)})^2 / f_{k'}^{(l)}}, \tag{17}$$

where $f_k^{(l)} = \sum_v q_{vk}^{(l)}$ is the total number of nodes softly assigned to cluster k. Raising $\mathbf{Q}^{(l)}$ to the second power and then dividing by the frequency per cluster allows the auxiliary distribution $\mathbf{P}^{(l)}$ to improve cluster purity and highlight confident assignments. Subsequently, the KL divergence is computed between $\mathbf{Q}^{(l)}$ and $\mathbf{P}^{(l)}$ to improve both $\mathbf{H}^{(l)}$ and $\{\xi_k^{(l)}\}_{k=1}^K$ through self-training as follows

$$\mathcal{L}_{st} = \sum_l \mathrm{KL}(\mathbf{P}^{(l)} || \mathbf{Q}^{(l)}) = \sum_l \sum_v \sum_k p_{vk}^{(l)} \log \frac{p_{vk}^{(l)}}{q_{vk}^{(l)}}. \tag{18}$$

2) Masked-embedding prediction: In our setting, node clustering is not the purpose, but a means to improve the modeling of research impacts over all nodes in the heterogeneous network. Recall the example in Figure 3 (a) about our insight: the impact of a node should be modeled within the proper contexts of research domains or communities. Therefore, we design a novel cluster-aware masking operation to the HGN node embeddings at each layer l before computing the loss in Eq. (2). Specifically, we randomly initiate K trainable embedding masks $\{\hat{\pi}_k^{(l)}\}_{k=1}^K$, where $\hat{\pi}_k^{(l)} = \sigma(\pi_k^{(l)}) \in \mathbb{R}_+^{d_l}$. Then we replace all HGN embedding $\mathbf{h}_v^{(l)}$ in Eq. (2) with the masked HGN embedding $\hat{\mathbf{h}}_v^{(l)}$ computed as

$$\hat{\mathbf{h}}_v^{(l)} = \sum_{k=1}^K q_{vk}^{(l)} \mathbf{h}_v^{(l)} \otimes \hat{\pi}_k^{(l)}, \tag{19}$$

where $q_{vk}^{(l)}$ is the soft clustering assignment we compute in Eq. (16) and $\otimes$ is the element-wise multiplication operation.

3) Cluster consistency/disparity regularizers: To further enhance the quality of self-training clustering, we design two regularizers for the consistency of clustering assignments across different HGN layers, and the disparity among different clusters in all HGN layers. Specifically, we improve cluster consistency by minimizing the KL divergence between the soft clustering assignment $\mathbf{Q}^{(l)}$ between all consecutive HGN layers as follows

$$\mathcal{L}_{con} = \sum_{l=1}^{L-1} \mathrm{KL}(\mathbf{Q}^{(l)} || \mathbf{Q}^{(l+1)}) = \sum_{l=1}^{L-1} \sum_v \sum_k q_{vk}^{(l)} \log \frac{q_{vk}^{(l)}}{q_{vk}^{(l+1)}}. \tag{20}$$

In parallel, we improve cluster disparity by minimizing the negative L2 loss between all pairs of cluster centers with

$$\mathcal{L}_{dis} = -\sum_{l=1}^{L} \sum_{k=1}^{K} \sum_{k'=1}^{K} ||\xi_k^{(l)} - \xi_{k'}^{(l)}||_2^2. \tag{21}$$

Thus the total loss of our cluster-aware module is

$$\mathcal{L}_{ca} = \lambda_{st} \mathcal{L}_{st} + \lambda_{con} \mathcal{L}_{con} + \lambda_{dis} \mathcal{L}_{dis}. \tag{22}$$

E. Text-Enhancing Module

In the heterogeneous network of publication data, the nodes of papers, authors, venues and their links can be directly extracted from meta-data. Unlike them, terms are not always specified for all papers, and their quality vary a lot even when they are available. In our setting, quality terms are rather important in the modeling of research impact, as they can describe the problems, techniques, applications and other factors studied in a research paper. Consider the example in Figure 3 (b), the terms on the upper side might have better qualities compared with the ones on the lower side, but there exists no standard to measure the exact quality of terms and all terms in Figure 3 (b) are commonly seen in the keyword lists of research papers either manually specified by experts or automatically extracted by machines. As such, the accurate modeling of quality terms in our HGN is a challenging task: without sufficient reliable supervision (*i.e.*, large amounts of ground-truth quality terms of research papers), how can we define the quality of terms and model the research impact of them towards citation prediction?

1) Quality terms should be *semantically coherent*. In each research domain, there is a limited set of popular terms, which are often mentioned together (*e.g.*, terms like graph, network, embedding and representation in the data mining domain).
2) Quality terms should be *statistically important*. In order to make a non-ignorable influence to the network model, a quality term needs to be frequent within certain papers

or domains, while not too frequent across all others (*e.g.*, terms like convolution and smoothness are important in the machine learning domain, while resnet is too specific and maximization is too general).

3) Quality terms should be *citation indicative*. In our setting, quality terms are also likely to trigger high citations of the papers that contain them.

We design a text-enhancing (TE) module with BERT, TF-IDF and impact-based voting towards mining quality terms based on the above heuristics. Due to the lack of sufficient reliable labeled data of quality terms, we propose to start from the weak supervision of research domain names like data mining, machine learning, *etc.*, which can be easily specified. From there, we propose to leverage the pre-trained BERT model [22] to bootstrap the initial set of *semantically coherent* terms relevant to each domain based on the domain names alone. Towards *statistical importance* and *citation indictiveness*, we further connect each paper with the terms it contains based on TF-IDF scoring [23], and design an impact-based voting mechanism for the papers to vote for the terms based on their research impacts. Finally, we train CA-HGN with the TE module in iterations to adaptively refine the sets of domain-specific quality terms while improving the embedding and clustering over all types of nodes.

1) Cluster-oriented term initialization: Since keywords have unreliable quality and are not always available, we dynamically maintain the set of quality terms. To start with, we manually construct a set of important research domain names[4], and leverage the pre-trained BERT masked language model (MLM) to bootstrap the set of quality terms from each domain name. Specifically, for each occurrence of a domain name, we replace it with the [MASK] token, and the BERT encoder will generate a contextualized embedding vector $\mathbf{z} \in \mathbb{R}^z$ for the masked tokens. The MLM can then output a probability distribution over the entire vocabulary Ω indicating the likelihood of a term u appearing at the masked position as follows

$$p(u|\mathbf{z}) = \text{Softmax}(\mathbf{W}_{t1}\sigma(\mathbf{W}_{t2}\mathbf{z} + \mathbf{b}_t)), \tag{23}$$

where $\mathbf{W}_{t1} \in \mathbb{R}^{z\times z}$, $\mathbf{b}_t \in \mathbb{R}^z$, and $\mathbf{W}_{t2} \in \mathbb{R}^{|\Omega|\times z}$ are the pre-trained parameters in the MLM. We use a hard threshold of κ (*e.g.*, $\kappa = 50$ in our experiments) to get the initial set of terms T_k^0 for each domain k based on the domain name.

After applying the pre-trained BERT model, we use the union of all sets of terms as the initial set of term nodes in our heterogeneous network, and then we build weighted links from papers to terms based on the classic TF-IDF scoring as follows

$$\begin{aligned}&\forall(v, e, u) \text{ where } \phi(v) = \text{paper and } \phi(u) = \text{term:}\\ &\omega(e) = \big(\frac{f(u,v)}{\sum_{u'\in\Omega} f(u',v)}\big)\big(\log\frac{|\{v \in \mathcal{V}; \phi(v) = \text{paper}\}|}{n(u)}\big),\end{aligned} \tag{24}$$

[4]The research domain names we use simply include: data, learning, vision, language, bio, robotics, network, system, security.

where $f(u, v)$ is the raw count of term u appearing in paper v, and $n(u)$ is the number of papers that include term u. In this way, the link weights directly reflect the statistical importance of terms based on their distribution over the papers– terms that are statistically important will have strong links with at least some of the papers.

To seamlessly combine TE with the CA module, for all term nodes, instead of randomly initializing the cluster centers $\{\xi_k^{(l)}\}_{k,l}$ in CA-HGN, we initialize them as $\xi_k^{(l)} = \frac{1}{|T_k^0|}\sum_{u\in T_k^0}\mathbf{h}_u^{(l)}$. In this way, we directly initialize the latent clusters in CA as research domains in TE with semantic guidance.

2) Adaptive term refinement: Like cluster modeling, term mining is not our ultimate goal in this work. We propose to adaptively refine the set of quality terms and model their research impacts through an impact-based voting mechanism. On top of the leverage of MLM and TF-IDF that guarantees the semantic coherence and statistical importance, we further consider the citation indicativeness of quality terms. Specifically, we apply the predicative model in Eq. (6) and take $\{\hat{y}_u^{(L)}\}_{\phi(u)=\text{paper}}$ at the last layer of our HGN, which gives the current estimation of the model towards the term's research impact. Now that under the help of our HGN, we can estimate the quality of a given (initial) set of potentially impactful terms retrieved with MLM and connected to the heterogeneous network with TF-IDF, we want the set of quality terms to be able to adaptively improve with lower quality ones removed and new higher quality ones inserted. To achieve this, we leverage MLM and TF-IDF again through an impact-based voting mechanism. After estimating the current sets of terms $\{T_k^t\}_{k=1}^K$, for each cluster k, we allow each term $u \in T_k^t$ to vote for its most semantically relevant terms based on the pre-trained MLM. Same as for each research domain name, we bootstrap the top κ terms $T(u)$ from each term u based on Eq. (23). Then each term u will vote for all κ terms in $T(u)$ with a weight of u's estimated impact $\hat{y}_u^{(L)}$. After gathering all weights, we rank the union of all $T(u)$'s for $u \in T_k^t$ and apply hard thresholding again on it with $|T_k^{(t)}|$ to get the new set of quality terms T_k^{t+1}. Subsequently, we apply TF-IDF in the same way as in term initialization to connect the new set of quality terms to the heterogeneous network, and further update the HGN to model their cluster membership and research impact.

As a quick summary, unlike the CA module, the TE module does not introduce additional losses, but is used to initialize the term nodes and cluster centers of CA. Then it is proceeded along the iterations between HGN and CA to further refine the term nodes and paper-term link weights which helps the clustering and embedding of all other nodes in the heterogeneous publication network.

F. Training Algorithm

Algorithm 1 outlines the training procedures of CATE-HGN. In Line 5, we apply neighborhood sampling as introduced in [10], which allows the memory fingerprint of

Algorithm 1 CATE-HGN Training

Input: A heterogeneous graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \phi, \psi, \omega\}$, partial labels on some papers $\mathcal{Y}$, hyper-parameters including: # of embedding layers L, # of clusters K, top relevant term threshold κ, batch size B, # of HGN mini-iterations I
Output: citation prediction $\hat{\mathcal{Y}}$ over all papers

1: Initialize the terms, paper-term weights and cluster centers as in Sec III-E1
2: **while** not converge **do**
3: **for** i in $1 : I$ **do**
4: Sample B papers with labels in $\mathcal{Y}$
5: Sample the 1-to-L-hop neighbors of the B papers each with size S
6: Compute $\mathcal{L}_{sup}$ according to Eq. (6) and (19)
7: For all sampled neighborhoods, compute $\mathcal{L}_{mi}$ according to Eq. (12) and Eq. (19)
8: Update all model parameters except for $\{\xi_k^{(l)}\}$ with the gradients of $\mathcal{L}_{hgn}$ in Eq. (2)
9: **end for**
10: Compute and update $\{\xi_k^{(l)}\}$ with the gradients of $\mathcal{L}_{ca}$ in Eq. (22)
11: Update the terms and paper-term weights according to Eqs. (23) and (24)
12: **end while**

HGN to be tractable on large-scale networks. In Lines 8 and 10, we update the cluster centers and all other model parameters with mini-iterations, whose design resembles the classic clustering algorithm of k-means; we find such iterative training of HGN and CA can lead to better and more stable results than jointly training without mini-iterations. In Lines 10 and 11, the order of updating cluster centers and quality terms can be reversed, but we find updating the cluster centers before quality terms can lead to faster convergence and better results, probably because the embeddings of new quality terms can severely disturb the cluster centers before being recomputed through HGN with mini-iterations. For large-scale networks, Lines 10 and 11 can also be done through sampled batches. Standard grid-search can be applied to find the optimal hyper-parameters, and our model is shown to be insensitive to most hyper-parameters (*e.g.*, in Figure 4). The output of Algorithm 1 is a trained CATE-HGN model which can be applied for the citation prediction task.

Complexity analysis. As a main contribution of the HGN module, we simplify existing heterogeneous GNNs with the cheap operation of entity-relation composition borrowed from KGE, and learn a single embedding transformation matrix to share across all link types. In this way, our HGN can still learn different projections for different types of link, while being extremely parameter- and computation-efficient.

Specifically, HGN has a constant number of parameters of $O\big(Ld\ (d + D_a + D_b + K)\big)$, where L is the number of embedding layers (*e.g.*, 1-4 in our experiments), d is the embedding dimensions (*e.g.*, a few hundred in all layers in our experiments), D_a and D_b are numbers of attention heads (*e.g.*, both 10 in our experiments), K is the number of clusters (*e.g.*, 10 in our experiments). Since all node embeddings are inductive (feature-based) and the number of link types is fixed, this number of learnable parameters does not grow with the graph size. Moreover, due to fixed-size neighborhood sampling, the maximum memory cost of CATE-HGN is also a constant number of $O(BS^Ld + BLKd)$, where B is the batch size, S is the neighborhood sample size, considering Line 10 to be done with batch size B as well. The memory cost of Line 11, *i.e.*, BERT inference, is not included, which grows linearly with the size of vocabulary $|\Omega|$.

As for time complexity, again since HGN simplifies various existing heterogeneous GNNs, the computation of Line 3-9 is much faster than most existing works. Lines 10 and 11 do incur additional computations, but they only need to be done once in a while between the mini-batches of HGN training. In particular, running Line 10 on one batch of data does not take significantly more time than running one iteration of Line 3-9, due to the similar time complexities of $O(BS^LdK)$ and $O(BS^Ld)$, respectively; the inference based on pre-trained BERT MLM is also very efficient, and the rest of the computations for term selection and paper-term weighting has a complexity of $O\big(\kappa(|\Omega| + |\mathcal{V}_p||v_p|)\big)$, where κ is the cut-off of relevant terms (*e.g.*, fixed to 100 in our experiments), $|\mathcal{V}_p|$ is the total number of paper nodes, and $|v_p|$ is the average length of papers regarding the number of terms.

G. Insights and Discussions

Through the development and experiments of CATE-HGN, we have learned the importance of powerful data models (*i.e.*, heterogeneous networks) and end-to-end deep learning frameworks (*i.e.*, GNNs) in the specific application domain of citation prediction. Our innovative model designs, especially regarding the joint discovery of latent research domains and unknown quality terms, have shown strong practical impacts in comparison with existing works that overlook such aspects.

As we have noticed recently, during the revision of this work, several papers have been published on the topic of applying GNNs for citation prediction [34], [35], [36], [37], [38], which has in fact verified the motivation and significance of our work. However, all of them have overlooked the latent research domains and unknown quality terms as we find important in this work, which have made their contributions rather orthogonal to ours. Inspired by their temporal model designs, we will also study the extension of CATE-HGN to the prediction of dynamic citations, through specific modeling of temporal factors such as the emergence of new nodes and evolution of clusters.

IV. Experiments

In this section, we use the real-world publication data from DBLP[1] to extensively evaluate our proposed CATE-HGN framework. We aim to answer the following three research questions.

- **RQ1:** What is the overall performance of CATE-HGN in comparison with different sets of baselines?
- **RQ2:** What are the impacts of the novel technical designs and some important hyper-parameters on CATE-HGN?
- **RQ3:** What interpretable results can be provided to further analyze the effectiveness of CATE-HGN?

A. Experimental Settings

1) Datasets.:

1) **DBLP-full**: The heterogeneous publication network constructed from the full DBLP dump as of 2019 joined with AMiner Citation V11[5], which includes all papers, authors, and venues. The terms are words extracted from the papers' keyword attributes. We further exclude all papers with missing attributes like authors, venue, citation count, and citation list.
2) **DBLP-single**: To highlight the effectiveness of domain-oriented clustering, we construct a heterogeneous publication network with papers only published in a list of data-related venues (*i.e.*, venues with "data" in the name) and their direct neighbors.
3) **DBLP-random**: To highlight the success of quality term mining, we construct a heterogeneous publication network with randomly generated terms and paper-term links, based on the per-paper term counts in the real data.

In all networks, we use papers published before 2014 for training, papers published in 2014 for validation, and papers published from 2015 to 2020 for testing. We use the setting in Sec II to generate labels as the per year average numbers of citations. We adopt the most widely used Root Mean Square Error (RMSE) metric for regression to measure the performance of compared algorithms.

Dataset	# papers	# authors	# venues	# terms	# links
DBLP-full	2.7M	1.6M	4.6K	188K	79.5M
DBLP-single	59K	83K	80	19K	1.7M
DBLP-random	2.7M	1.6M	4.6K	188K	79.5M

TABLE I
STATISTICS OF THE FOUR DATASETS WE CONSTRUCT AND USE.

2) Compared algorithms.:

1) **BERT** [22]: We use the pre-trained BERT model and fine-tune it with our citation prediction loss. This baseline represents the state-of-the-art model that only uses the textual contents of papers to predict citations.
2) **GAT** [11]: This baseline represents the state-of-the-art model that only uses the graph topology of a homogeneous network to predict citations.
3) **CCP** [2]: An older traditional citation prediction method based on feature engineering. We implement 9 out of 10 features as described in the paper, except for the h-index which is unavailable. We use CART [39] as the best predictive model.
4) **CPDF** [1]: The state-of-the-art citation prediction method. We implement 16 out of 17 features with CART as described in the paper, except for the paper length in pages which is unavailable.
5) **metapath2vec** [40]: A popular unsupervised heterogeneous network embedding algorithm based on given meta-paths. A three layer MLP with equal sizes is trained on the output of metapath2vec to predict paper citations.
6) **hin2vec** [41]: The state-of-the-art unsupervised heterogeneous network embedding algorithm that does not rely on given meta-paths. The same MLP as metapath2vec is trained on its output.
7) **R-GCN** [12]: A popular GNN model designed for KGE by computing exclusive transformation matrices for each type of link.
8) **HAN** [15]: A popular heterogeneous GNN with node-level and meta-path-level attentions.
9) **HetGNN** [16]: A recent heterogeneous GNN with random-walk-based neighbor sampling and LSTM-based aggregation.
10) **HGT** [13]: A recent heterogeneous GNN with edge-type-specific node attention and node-type-specific message aggregation.
11) **MAGNN** [17]: One of the state-of-the-art heterogeneous GNNs with intra-metapath and inter-metapath aggregations.
12) **HGCN** [14] : One of the state-of-the-art heterogeneous GNNs that models the compatibility among different types of links.

3) Parameter settings.: Our HGN includes several parameters that are very common in other heterogeneous GNNs, and our CA and TE modules include a few more. However, most of them can be empirically set without much tuning, and we study the impacts of some important ones in Section IV-C.

By default, we set the citation thresholds $c_1 = 1$ and $c_2 = 5$ based on moderate grid-search. We use a two-layer HGN with both layers of size 100 (*i.e.*, $L = 2$ and $d_1 = d_2 = 100$) and composition function of correlation [28]; the numbers of attention heads are both set to 10 (*i.e.*, $D_a = D_b = 10$); the number of clusters K is also set to 10 (same as the number of actual domain names we specify[4] plus one for others); the loss weights are set as $\lambda_{mi} = \lambda_{con} = \lambda_{dis} = 0.1$; relevant term threshold κ is set to 100; batch size B is set to 1024; neighborhood sample size S is set to 50; number of HGN mini-iterations I is set to 100. Without explicit specification, we always use ReLU as the activation function σ.

For all baselines except for BERT, we use the same embedding size of 100 and follow the default settings for all other parameters as specified in the original papers or published codes. For all GNNs, we aggregate and normalize the pre-trained word embedding of all words in the titles as paper features, in the venue names as venue features, in the titles of all published papers as author features, and the word itself as term features. For all algorithms that need a given set of meta-paths, we use the most fundamental ones of P-P, P-A-P, P-V-P and P-T-P with equal weights.

[5]https://lfs.aminer.cn/misc/dblp.v11.zip

Algorithms	full	single	random
BERT	17.3704	17.1951	17.3704
GAT	16.9865	16.8054	17.6341
CCP	10.4202	9.3002	10.4202
CPDF	8.0837	7.2431	8.0837
metapath2vec	14.3552	13.8781	17.5800
hin2vec	14.0003	13.5002	16.1450
R-GCN	12.6441	12.3614	13.9681
HAN	9.8362	9.6211	10.5597
HetGNN	10.1304	9.9621	10.5597
HGT	9.1169	8.7524	10.2027
MAGNN	8.1935	7.6928	9.7226
HGCN	8.3210	8.2225	10.4448
HGN	6.9833	6.6693	9.1583
CA-HGN	5.3397	5.7088	7.5101
CATE-HGN	**3.4574***	**4.8305***	**3.4574***

TABLE II
PERFORMANCE OF COMPARED ALGORITHMS ON DIFFERENT DATASETS REGARDING THE RMSE METRIC. SCORES WITH * ALL PASSED THE SIGNIFICANCE T-TEST WITH P-VALUE 0.05.

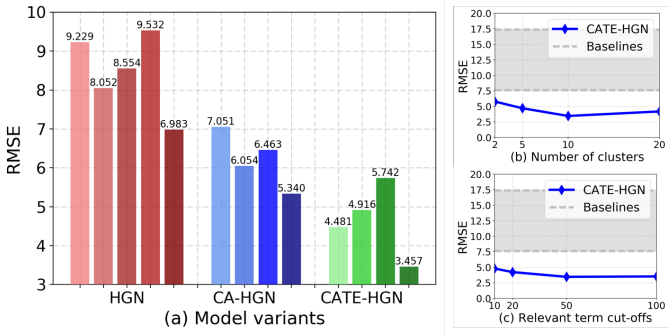


Fig. 4. **In-depth study on the performance of CATE-HGN regarding the impact of model components and hyper-parameters.**

B. Overall Performance (RQ1)

The performance of compared algorithms on different datasets regarding RMSE metric is provided in Table II. As we can observe: (1) On DBLP-full, the compared algorithms clearly form five tiers: BERT performs poorly because it only uses paper contents without link topology, whereas GAT falls short due to the ignorance of node/link types and the modeling of a homogeneous network; the traditional citation prediction methods of CCP and CPDF perform much better, although the results are not directly comparable to their original papers due to different datasets and problem settings; metapath2vec and hin2vec learn an unsupervised network representation and only get the supervision from ground-truth citations afterwards, thus only getting slightly better performance than BERT and GAT; among all heterogeneous GNNs, our HGN models significantly outperform all others due to our proper model designs, while the CA and TE modules both make clear contributions to the overall performance. (2) On DBLP-single, most algorithms including our HGN get a slight improvement, probably due to less noises, while the CA and TE modules lead to rather limited improvements, which in fact validates the conjecture that they are more useful in networks with multiple underlying domains; (3) On DBLP-random, most algorithms including our HGN and CA-HGN models get significantly worse performance, supporting our insight that the correct terms and paper-to-term links are important in the modeling of research impact; however, our CATE-HGN model is not affected at all, due to its ability of automatically mining quality terms from simple domain names.

C. Ablations and Hyper-parameters (RQ2)

Figure 4 (a) shows the results of our comprehensive analysis towards several CATE-HGN variants without certain important models components. As we can observe: (1) Besides the correlation-based node-link composition, we compare with the subtraction-based and multiplication-based compositions (the 1st and 2nd bars in the HGN group), which tend to perform worse (compared with the full HGN model– the last bar in HGN), probably because the subtraction and multiplication operations can hardly lead to smooth cross-type embeddings while correctly modeling the research impacts over different types of nodes. (2) The HGNs without our MI maximization (the 3rd bar in HGN) or attention (the 4th bar in HGN) also lead to inferior performance, with MI influencing more on the embedding smoothness and attention more on the task performance, which directly supports our designs of both components. (3) For the CA module, the removal of any of the self-training (the 1st bar in CA-HGN), consistency (the 2nd bar in CA-HGN) or disparity (the 3rd bar in CA-HGN) leads to lower performance (compared with the full CA-HGN model– the 4th bar), while self-training is the most important among all three. (4) For the TE module, the removal of any of the BERT-based quality term initialization (the 1st bar in CATE-HGN), TF-IDF based paper-term linking (the 2nd bar in CATE-HGN) and iterative training (the 3fd bard in CATE-HGN) leads to lower performance (compared with the full CATE-HGN model– the 4th bar); among them, iterative training contributes the most, while initialization contributes less, which means even without very good initializations (*i.e.*, using available keywords of the papers as all other models), the TE module can still gradually discover the real high-quality terms through training.

Figure 4 (b) and (c) show the performance of CATE-HGN with varying hyper-parameters. As we can see, neither of the cluster numbers or relevant term cut-offs have significant impact on the performance unless they are set to extremely small or large; in general, setting the cluster number in a large range around 10-20 and relevant term cut-off in a large range around 50-100 can lead to a good trade-off between model performance and efficiency.

D. Case Studies (RQ3)

In CATE-HGN, the MI-based cross-type embedding alignment allows us to directly model and predict the impact of all types of nodes with the citation regressor, and the CA

module allows us to model the research domain belongingness of all nodes. In Table III, we list the top-impactful authors, venues, and terms, whereas in Figure 5, we further visualize the quality terms mined by CATE-HGN during training. Due to space limit, we only present the results on the two domains of data and system as examples. As we can observe, CATE-HGN has a pretty accurate modeling of impactful authors, venues, and terms by domains, while the identity and quality of impactful terms are indeed adaptively improved through training. Note that, since we bootstrap the clusters from simple domain names, the modeling of domains are not exactly accurate, due to a bias towards terms that are more frequent and directly relevant to the domain names in semantics. Moreover, due to our simplification of citation prediction into a binary classification problem, the results are biased towards authors/venues/terms connected with larger numbers of moderately well-cited papers, instead of those with papers of extremely high citations or simply with a lot of papers.

V. Related Work

A. Citation prediction

The problem of citation prediction has been studied for a long time [42], [43]. However, existing methods mostly rely on a simple combination of feature engineering and predictive models. For example, the most recent work [1] extracted a total of 17 features for each paper regarding author interdisciplinarity, author influence, title words and other classical features, after which they applied a classification tree to predict the interval of a paper's citation count. [2] extracted a set of 10 features for each paper around topics, authors and venues, and then applied different regression models to directly predict a paper's citation count. With the same spirit, [3] applied a more complex predictive model of stepwise regression, whereas [4] used slightly different features for biomedical articles. The feature engineering process is ad-hoc and time-consuming, and is isolated from the direct supervision of existing citation counts. Until very recently, end-to-end deep graph models like GNNs have been studied for citation prediction [34], [35], [36], [37], [38], but only [35] has considered the heterogeneous nodes beyond papers, and none of them has considered the influence of latent domains and important terms.

Other than the prediction of citation counts, various research has been done on the modeling of scientific research impact, regarding relevant problems including the prediction of actual citations (citation links) [44], [45], influence propagations among authors and papers [46], [47], [48], detection of topic initiators [49], ranking of authors and venues [50], [19], [8], *etc.*. Although these methods cannot be directly applied for citation prediction, most of them are developed in a heterogeneous publication network setting, which reassures us that such a setting can provide necessary information for citation prediction through research impact modeling, and motivates us to develop a general framework that can be potentially useful for various tasks related to research impact modeling.

B. Heterogeneous GNNs

GNNs have been widely studied for various graph mining tasks recently [9], [10], [11], [51], [52], [53]. Here we focus the discussion on GNNs on heterogeneous networks, which we categorize into two groups: shallow and deep heterogeneous GNNs. Shallow GNNs are designed to directly capture the proximity among nodes in a network. For example, [40], [41] use heterogeneous random walks to define and capture node proximity, whereas [54], [55] use first- and second-order heterogeneous neighborhoods. These models are often trained in an unsupervised fashion, thus being generally useful for various downstream tasks, but not specifically tuned for any.

Deep GNNs are mainly designed based on the impactful model of GCN [9]. To extend the homogeneous GCN to heterogeneous networks, [12], [13], [14], [56], [57] use edge types to parameterize exclusive embedding propagation channels, whereas [15], [16], [58] use meta-paths [7]. Motivated by the success of GAT [11], many of the models also use attention mechanisms to weigh the different propagation channels [15], [13], [17]. These models can be trained towards specific tasks, but their design often focuses on correctly embedding a specific type of target nodes due to direct task need (*e.g.*, classification of authors [15], [16], [17], [14]), and they do not equivalently model all types of nodes in the same embedding space. This is not ideal in our case, since we aim to also capture the research impacts of authors/venues/terms from the supervision on papers.

VI. Conclusions

In this work, we revisit the important yet challenging task of citation prediction, under the setting of heterogeneous publication networks. Equipped with recent advanced techniques of GNN and deep learning, we design a powerful end-to-end model to dispel traditional manual feature engineering, and demonstrate its superior performance. While we focus on citation prediction, the powerful framework of CATE-HGN could be further studied in various other downstream applications over text-rich heterogeneous networks. Immediate future works include the modeling of temporal factors to break the limitation of static citation prediction, as well as incremental training of large-scale models over new nodes and evolving clusters towards a deployable real-time system.

Appendix

A. Proof for Theorem 3.1.

Theorem 3.1 (Modeling meta-paths with a composition of R Functions): For a heterogeneous network $\mathcal{G}$ defined in Definition 3.1 with R types of relations, we assume there is an oracle function $\hat{O}$ that takes in a target node v's meta-paths information $\mathcal{M}_v \in \mathbb{R}^d$ on $\mathcal{G}$, and outputs the v's ground-truth label $y_v \in \mathbb{R}^d$. When $\mathcal{M}_v$ is absolutely continuous with respect to the Lebesgue measure, for any given approximation error ε and R functions $\{F_r | r \in [R]\}$, there exists a composition function $Comp(\cdot|\{F_r|r \in [R]\}) : \mathbb{R}^d \to \mathbb{R}^d$, which is viewed as the gradient function of an FNN $u(\cdot) : \mathbb{R}^d \to \mathbb{R}$ with ReLU activation, of depth $L = [\log_2 n]$ and width

Authors		Venues		Terms	
Data	System	Data	System	Data	System
Jiawei Han	Rajkumar Buyya	VLDB	IEEE TIE	social	energy
Philip S. Yu	Luca Benini	SIGMOD	IEEE TAC	recommend	cloud
Wil M. P. van der Aalst	Ian T. Foster	TKDE	IEEE TCST	privacy	gpu
Jian Pei	Schahram Dustdar	WWW	MIS Quarterly	rank	attack
Yufei Tao	Onur Mutlu	SIGKDD	IEEE TFS	graph	center
Jennifer Widom	MengChu Zhou	SIGIR	IEEE TPDS	sparse	convex
Hector Garcia-Molina	Thomas A. Henzinger	WSDM	ISCA	empirical	vehicular
Alon Y. Halevy	Ion Stoica	ICDE	Automatica	metric	multicore
Susan T. Dumais	David E. Culler	CIKM	JCP	large	demand
Christos Faloutsos	Scott A. Mahlke	ICDM	HPCA	probabilistic	workload

TABLE III
TOP-IMPACT AUTHORS, VENUES, TERMS BY DOMAINS LEARNED BY CATE-HGN.

(a) Data-Initialization (b) Data-During training (c) Data-Convergence

(a) System-Initialization (b) System-During training (c) System-Convergence

Fig. 5. **Adaptive quality term mining results by domains learned by CATE-HGN.**

$N = 2L$, where $n = O(\frac{1}{\varepsilon^d})$. For the 1-Wasserstein distance measurement $W_1(\cdot,\cdot)$, we have

$$\mathbb{E}_{\mathcal{M}_v \sim H}[W_1(\hat{O}(\mathcal{M}_v), Comp(\mathcal{M}_v|\{F_r|r \in [R]\})] < \varepsilon.$$

Proof: Theorem 3.1 can be obtained by properly revising Theorem 2.1 in [30], which we state as follows.

Lemma 3.2 (Theorem 2.1 in [30]): Let P and Q be the target and the source distributions respectively, both defined on $\mathbb{R}^d$. Assume that $\mathbb{E}_{x\sim P}||x||^3$ is bounded and Q is absolutely continuous with respect to the Lebesgue measure. It holds that for any given approximation error ε, setting $n = O(\frac{1}{\varepsilon^d})$, there is a fully connected and feed-forward deep neural network $u(\cdot)$ of depth $L = [\log_2 n]$ and width $N = 2L$, with d inputs and a single output and with ReLU activation such that for 1-Wasserstein distance measurement, $W_1(P, \nabla u(Q)) < \varepsilon$ holds. Here, $\nabla u(\cdot)$ is the function $\mathbb{R}^d \to \mathbb{R}^d$ induced by the gradient of u while $\nabla u(Q)$ is the distribution that is generated from the distribution Q through the mapping $\nabla u(\cdot)$.

With $P = \hat{O}(\mathcal{M}_v)$ and $Q = \mathcal{M}_v$, where $\mathcal{M}_v$ is absolutely continuous with respect to the Lebesgue measure. We only need to show that $P = \hat{O}(\mathcal{M}_v)$ has a bounded $\mathbb{E}_{x\sim P}||x||^3$.

Obviously, $\hat{O}(\mathcal{M}_v)$, as a set of label vectors, is absolute a bounded 3-order moment. Thus, by setting the $Comp(\cdot|\{F_r|r \in [R]\}) : \mathbb{R}^d \to \mathbb{R}^d$ as the gradient function of the feed-forward deep neural network $u(\cdot)$ in Lemma 3.2, Theorem 3.1 is proved.

For a better empirical performance, our case adopts R type-aware FNNs for the R F_r functions in Theorem 3.1 and composing these FNNs with the construction HGN. In this way, we argue that the design of HGN can approximate any message passing functions of the higher-order meta-paths.

ACKNOWLEDGEMENT

This research was supported in part by the internal funds and GPU servers provided by the Computer Science Department of Emory University; in part by US National Science Foundation IIS-19-56151, IIS-17-41317, and IIS 17-04532. Any opinions, findings, and conclusions or recommendations expressed herein are those of the authors and do not necessarily represent the views, either expressed or implied, of the U.S. Government.

REFERENCES

[1] H. S. Bhat, L.-H. Huang, S. Rodriguez, R. Dale, and E. Heit, "Citation prediction using diverse features," in *Workshop of the IEEE International Conference on Data Mining Workshop (ICDMW)*, 2015, pp. 589–596.

[2] R. Yan, J. Tang, X. Liu, D. Shan, and X. Li, "Citation count prediction: learning to estimate future citations for literature," in *Proceedings of the ACM international Conference on Information and Knowledge Management (CIKM)*, 2011, pp. 1247–1252.

[3] T. Yu, G. Yu, P.-Y. Li, and L. Wang, "Citation impact prediction for scientific papers using stepwise regression analysis," *Scientometrics*, vol. 101, no. 2, pp. 1233–1252, 2014.

[4] L. D. Fu and C. Aliferis, "Models for predicting and explaining citation count of biomedical articles," in *AMIA Annual Symposium Proceedings*, vol. 2008. American Medical Informatics Association, 2008, p. 222.

[5] Y. Dong, Z. Hu, K. Wang, Y. Sun, and J. Tang, "Heterogeneous network representation learning." in *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, vol. 20, 2020, pp. 4861–4867.

[6] C. Yang, Y. Xiao, Y. Zhang, Y. Sun, and J. Han, "Heterogeneous network representation learning: A unified framework with survey and benchmark," *IEEE Transactions on Knowledge and Data Engineering (TKDE)*, 2020.

[7] Y. Sun and J. Han, "Mining heterogeneous information networks: principles and methodologies," *Synthesis Lectures on Data Mining and Knowledge Discovery*, vol. 3, no. 2, pp. 1–159, 2012.

[8] Y. Sun, Y. Yu, and J. Han, "Ranking-based clustering of heterogeneous information networks with star network schema," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2009, pp. 797–806.

[9] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2017.

[10] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

[11] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[12] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. v. d. Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *Proceedings of the European Semantic Web Conference (ESWC)*. Springer, 2018, pp. 593–607.

[13] Z. Hu, Y. Dong, K. Wang, and Y. Sun, "Heterogeneous graph transformer," in *Proceedings of the International World Wide Web Conference (WWW)*, 2020, pp. 2704–2710.

[14] Z. Zhu, X. Fan, X. Chu, and J. Bi, "Hgcn: A heterogeneous graph convolutional network-based deep learning model toward collective classification," in *Proceedings of ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2020, pp. 1161–1171.

[15] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. S. Yu, "Heterogeneous graph attention network," in *Proceedings of the International World Wide Web Conference (WWW)*, 2019, pp. 2022–2032.

[16] C. Zhang, D. Song, C. Huang, A. Swami, and N. V. Chawla, "Heterogeneous graph neural network," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2019, pp. 793–803.

[17] X. Fu, J. Zhang, Z. Meng, and I. King, "Magnn: Metapath aggregated graph neural network for heterogeneous graph embedding," in *Proceedings of the International World Wide Web Conference (WWW)*, 2020, pp. 2331–2341.

[18] C. Yang, A. Pal, A. Zhai, N. Pancha, J. Han, C. Rosenberg, and J. Leskovec, "Multisage: Empowering graphsage with contextualized multi-embedding on web-scale multipartite networks," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2020.

[19] Y. Sun, J. Han, P. Zhao, Z. Yin, H. Cheng, and T. Wu, "Rankclus: integrating clustering with ranking for heterogeneous information network analysis," in *Proceedings of the International Conference on Extending Database Technology (EDBT)*, 2009, pp. 565–576.

[20] K. Nigam and R. Ghani, "Analyzing the effectiveness and applicability of co-training," in *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, 2000, pp. 86–93.

[21] J. Xie, R. Girshick, and A. Farhadi, "Unsupervised deep embedding for clustering analysis," in *Proceedings of the International Conference on Machine Learning (ICML)*. PMLR, 2016, pp. 478–487.

[22] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of Annual Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019, pp. 4171–4186.

[23] K. S. Jones, "A statistical interpretation of term specificity and its application in retrieval," *Journal of documentation*, 1972.

[24] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using gaussian fields and harmonic functions," in *Proceedings of International conference on Machine learning (ICML)*, 2003, pp. 912–919.

[25] D. Zhou, O. Bousquet, T. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2003.

[26] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2013.

[27] B. Yang, W.-t. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," in *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

[28] M. Nickel, L. Rosasco, and T. Poggio, "Holographic embeddings of knowledge graphs," in *Proceedings of the AAAI Conference on Artificial Intelligence (AAA)*, 2016.

[29] S. Vashishth, S. Sanyal, V. Nitin, and P. Talukdar, "Composition-based multi-relational graph convolutional networks," in *Proceedings of the 7th International Conference on Learning Representations*, 2019.

[30] Y. Lu and J. Lu, "A universal approximation theorem of deep neural networks for expressing probability distributions," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020, pp. 3094–3105.

[31] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, "Learning deep representations by mutual information estimation and maximization," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[32] Z. Peng, W. Huang, M. Luo, Q. Zheng, Y. Rong, T. Xu, and J. Huang, "Graph representation learning via graphical mutual information maximization," in *Proceedings of the International World Wide Web Conference (WWW)*, 2020, pp. 259–270.

[33] S. Nowozin, B. Cseke, and R. Tomioka, "f-gan: Training generative neural samplers using variational divergence minimization," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2016.

[34] B. Plank and R. van Dalen, "Citetracked: A longitudinal dataset of peer reviews and citations," in *Joint Workshop on Bibliometric-enhanced Information Retrieval and Natural Language Processing for Digital Libraries at the International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIRW)*, 2019.

[35] S. Jiang, B. Koch, and Y. Sun, "Hints: Citation time series prediction for new publications via dynamic heterogeneous information network embedding," in *Proceedings of the International World Wide Web Conference (WWW)*, 2021, pp. 3158–3167.

[36] D. Cummings and M. Nassar, "Structured citation trend prediction using graph neural networks," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 3897–3901.

[37] J. Wen, L. Wu, and J. Chai, "Paper citation count prediction based on recurrent neural network with gated recurrent unit," in *Proceedings of the IEEE International Conference on Electronics Information and Emergency Communication (ICEIEC)*. IEEE, 2020, pp. 303–306.

[38] A. N. Holm, B. Plank, D. Wright, and I. Augenstein, "Longitudinal citation prediction using temporal graph neural networks," *arXiv preprint arXiv:2012.05742*, 2020.

[39] W.-Y. Loh, "Classification and regression trees," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 1, no. 1, pp. 14–23, 2011.

[40] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2017, pp. 135–144.

[41] T.-y. Fu, W.-C. Lee, and Z. Lei, "Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning," in *Proceedings of ACM International Conference on Information and Knowledge Management (CIKM)*, 2017, pp. 1797–1806.

[42] A. Ibáñez, P. Larrañaga, and C. Bielza, "Predicting citation count of bioinformatics papers within four years of publication," *Bioinformatics*, vol. 25, no. 24, pp. 3303–3309, 2009.

[43] C. Castillo, D. Donato, and A. Gionis, "Estimating number of citations using author reputation," in *Proceedings of the International Symposium on String Processing and Information Retrieval*. Springer, 2007, pp. 107–117.

[44] X. Ren, J. Liu, X. Yu, U. Khandelwal, Q. Gu, L. Wang, and J. Han, "Cluscite: Effective citation recommendation by information network-based clustering," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2014, pp. 821–830.

[45] X. Yu, Q. Gu, M. Zhou, and J. Han, "Citation prediction in heterogeneous bibliographic networks," in *Proceedings of the SIAM International Conference on Data Mining (SDM)*. SIAM, 2012, pp. 1119–1130.

[46] L. Dietz, S. Bickel, and T. Scheffer, "Unsupervised prediction of citation influences," in *Proceedings of the International Conference on Machine learning (ICML)*, 2007, pp. 233–240.

[47] D. Zhou, X. Ji, H. Zha, and C. L. Giles, "Topic evolution and social interactions: how authors effect research," in *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, 2006, pp. 248–257.

[48] L. Liu, J. Tang, J. Han, M. Jiang, and S. Yang, "Mining topic-level influence in heterogeneous networks," in *Proceedings of the ACM International Conference on Information and Knowledge Management (CIKM)*, 2010, pp. 199–208.

[49] X. Jin, S. Spangler, R. Ma, and J. Han, "Topic initiator detection on the world wide web," in *Proceedings of the International World Wide Web Conference (WWW)*, 2010, pp. 481–490.

[50] N. Pobiedina and R. Ichise, "Predicting citation counts for academic literature using graph pattern mining," in *Proceedings of the International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*. Springer, 2014, pp. 109–119.

[51] C. Yang, P. Zhuang, W. Shi, A. Luu, and P. Li, "Conditional structure generation through graph variational generative adversarial nets," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

[52] Q. Zhu, C. Yang, Y. Xu, H. Wang, C. Zhang, and J. Han, "Transfer learning of graph neural networks with ego-graph information maximization," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[53] C. Yang, M. Tang, and P. Li, "Graph auto-encoder via neighborhood wasserstein reconstruction," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2022.

[54] J. Tang, M. Qu, and Q. Mei, "Pte: Predictive text embedding through large-scale heterogeneous text networks," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2015, pp. 1165–1174.

[55] Y. Shi, Q. Zhu, F. Guo, C. Zhang, and J. Han, "Easing embedding learning by comprehensive transcription of heterogeneous information networks," in *Proceedings of the ACM International Conference on Knowledge Discovery and Data Mining (KDD)*, 2018, pp. 2190–2199.

[56] C. Yang, M. Liu, F. He, X. Zhang, J. Peng, and J. Han, "Similarity modeling on heterogeneous networks via automatic path discovery," in *Proceedings of the European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases (ECML-PKDD)*, 2018.

[57] C. Yang, J. Zhang, and J. Han, "Neural embedding propagation on heterogeneous networks," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2019.

[58] C. Yang, Y. Feng, P. Li, Y. Shi, and J. Han, "Meta-graph based hin spectral embedding: Methods, analyses, and insights," in *Proceedings of the IEEE International Conference on Data Mining (ICDM)*, 2018.