2021 IEEE International Parallel and Distributed Processing Symposium (IPDPS) | 978-1-6654-4066-0/20/$31.00 ©2021 IEEE | DOI: 10.1109/IPDPS49936.2021.00073

Facilitating Data Discovery for Large-scale Science Facilities using Knowledge Networks

Yubo Qin[1], Ivan Rodero[1], and Manish Parashar[1,2]

[1]Rutgers Discovery Informatics Institute, Rutgers University, New Brunswick, New Jersey, USA
[2]Scientific Computing Imaging Institute, University of Utah, Salt Lake City, Utah, USA
{yubo.qin; irodero; parashar}@rutgers.edu

***Abstract*—Large-scale multiuser scientific facilities, such as geographically distributed observatories, remote instruments, and experimental platforms, represent some of the largest national investments and can enable dramatic advances across many areas of science. Recent examples of such advances include the detection of gravitational waves and the imaging of a black hole's event horizon. However, as the number of such facilities and their users grow, along with the complexity, diversity, and volumes of their data products, finding and accessing relevant data is becoming increasingly challenging, limiting the potential impact of facilities. These challenges are further amplified as scientists and application workflows increasingly try to integrate facilities' data from diverse domains.**

In this paper, we leverage concepts underlying recommender systems, which are extremely effective in e-commerce, to address these data-discovery and data-access challenges for large-scale distributed scientific facilities. We first analyze data from facilities and identify and model user-query patterns in terms of facility location and spatial localities, domain-specific data models, and user associations. We then use this analysis to generate a knowledge graph and develop the collaborative knowledge-aware graph attention network (CKAT) recommendation model, which leverages graph neural networks (GNNs) to explicitly encode the collaborative signals through propagation and combine them with knowledge associations. Moreover, we integrate a knowledge-aware neural attention mechanism to enable the CKAT to pay more attention to key information while reducing irrelevant noise, thereby increasing the accuracy of the recommendations. We apply the proposed model on two real-world facility datasets and empirically demonstrate that the CKAT can effectively facilitate data discovery, significantly outperforming several compelling state-of-the-art baseline models.

I. Introduction

Large-scale science facilities (LFs), such as multiuser scientific observatories, instruments, and experimental platforms, provide a broad community of researchers and educators with open access to shared-use infrastructure and data products generated from geo-distributed instruments and equipment [1]. These facilities have become key enablers of a range of scientific discoveries, including the recent detection of gravitational waves [2] and the imaging of a black hole's event horizon [3].

The availability of these LFs is changing how scientists access experimental and observational data and data products, as well as the nature of their applications. The latter are increasingly taking the form of application workflows with integrated data pipelines, and they require parallel and distributed processing. An example is the earthquake early warning system [4] workflow, which leverages machine learning techniques to gather and locally process high-precision GPS and seismometer data from distributed diverse sources in a timely manner and then integrate the results with more traditional modeling and analysis. Correspondingly, the underlying cyberinfrastructure is evolving to support these data-driven distributed workflows. For example, the NSF-funded Virtual Data Collaboratory [5] is designed to support data-driven end-to-end workflows that combine data from multiple data sources at runtime. However, discovering and effectively using data and data products from multiple geo-distributed sources and across multiple domains remains challenging [6], [7].

Specifically, as the number of LFs grow along with the complexity, diversity, and volumes of the data they produce, ensuring all users can find and access relevant data is becoming increasingly challenging. As of October 2020, there are 33 major facilities in-operation funded by the US National Science Foundation [8], and these are complemented by similar LF facilities supported by other US agencies and by other countries. These LFs are producing – or will produce – a massive amount of diverse data and data products to serve users from different science domains. For example, the Ocean Observatories Initiative (OOI) [9], [10] has deployed dozens of stable platforms and mobile assets carrying hundreds of instruments and providing thousands of scientific and engineering data products. Furthermore, as applications target broader science questions, they are increasingly seeking to integrate data from multiple facilities as part of end-to-end workflows. For example, although the OOI has primarily targeted the oceanography community, several interdisciplinary projects, such as studying whole earth systems, are using its data and data products.

In this work, we leverage the concepts underlying recommender systems to facilitate the discovery of and access to data (and data products) from large facilities. We analyze two existing facilities: the OOI and the Geodetic Facility for the Advancement of Geoscience (GAGE) [11]. Through this analysis, we observe three key affinities – based on *instrument locality*, *data-domain model*, and *user association* – that can characterize user-data-query behaviors, and we exploit knowledge graph (KG) techniques to combine these affinities into a collaborative knowledge graph (CKG).

Inspired by the recent developments in GNN-based recom-

978-1-6654-4066-0/21/$31.00 ©2021 IEEE
DOI 10.1109/IPDPS49936.2021.00073

mendation models [12]–[15], we propose the *collaborative knowledge-aware graph attention network* (CKAT) recommendation model, which can explicitly encode collaborative signals in user–data–item interactions and auxiliary knowledge associations (e.g., from published data models) into the CKG. To alleviate noise issues during the attentive embedding propagation (see Section V), we integrate the knowledge-aware neural attention mechanism that optimizes the model's ability to focus on learning key information. Furthermore, we filter out irrelevant information when we create the CKG. We evaluate the proposed CKAT empirically using data-query traces from the OOI and GAGE facilities. The experimental evaluation results show the value of recommender systems in addressing data discovery and accessing challenges for LFs, and they show that the CKAT improves accuracy by 6.12% and 7.26% for the OOI and GAGE, respectively, as compared to the state-of-the-art models. The code is publicly accessible at https://github.com/qybo1234/CKAT_rec_model.

This paper makes the following key contributions:

- We analyze data-query traces from current facilities and develop a model for the observed data-query affinities base on *instrument locality*, *data-domain model*, and *user association*.
- We propose an approach for knowledge extraction that aims to collect knowledge from both data-query traces and the data models published by the LFs as well as to integrate this knowledge into a CKG.
- We design the CKAT recommendation model, which utilizes the extracted knowledge about both the data models and the user–data interactions.
- We evaluate the effectiveness and performance of the CKAT model, the impact of each of its components, and the impact of various knowledge-source combinations on the quality of recommendations.

The rest of this paper is organized as follows. Section II discusses recent developments in recommender systems and their use for science data. Section III presents an analysis of OOI and GAGE data-query behavior and the modeling of user-query patterns using correlations and affinities. Section IV describes the construction of the CKG. Section V presents the design of the CKAT recommendation model. Section VI presents a performance evaluation of the CKAT. Section VII concludes the paper and outlines future work.

II. Background and Related Work

A. Recommendation systems for science data objects

While recommendation systems are being used quite extensively by enterprise applications such as e-commerce, their use for scientific applications has been more limited. This is, in part, due to the lack of linked data and enriched metadata, such as information about data usage. Recent studies [16], [17] have focused on extracting meaningful knowledge from literature to collect linked data. Weston et al. [16] have applied natural language processing (NLP) techniques to extract information from materials-science literature. Mukund et al. [18] proposed

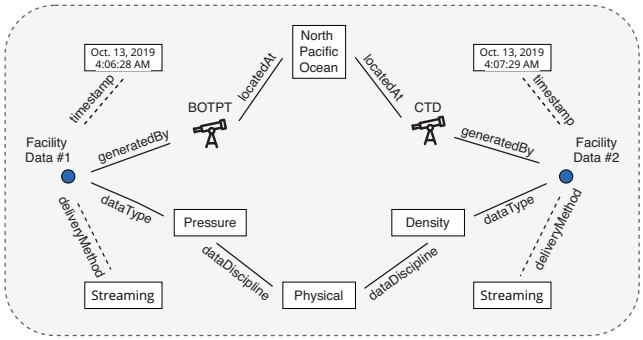


Fig. 1: An illustrative example showing the Ocean Observatories Initiative (OOI) knowledge graph. The blue dots represent two OOI data objects, and other entities are their attributes. They are connected through the paths (in solid lines) along with their attributes.

an NLP-based method to discover knowledge from the LIGO logbook and enable recommendations for astronomical observatories. Barros et al. [19] developed a hybrid recommender model for chemical compounds.

However, recommendation models that can support the discovery and access of data objects from different data sources has not been explored to best of our knowledge. We believe that this is the first study that models large-scale data facilities' user-query patterns and leverages knowledge graph techniques to support the discovery of facilities' data.

B. Knowledge graphs

A knowledge graph (KG) is a heterogeneous graph that contains a structured representation of facts, where nodes function as entities, and edges correspond to relationships. Many recent studies [12]–[14], [20]–[23] have leveraged KGs to carry auxiliary information to alleviate the cold-start and data-sparsity challenges.

Large facilities have lots of structured information for the instruments they deploy, including location and domain data models. We can represent this information in the form of a KG. Figure 1 depicts an illustrative KG example based on the OOI facility. It shows two OOI data objects (blue dots), their attributes (data discipline, data type, instrument, and location), and their relationships. The KG explicitly presents the connectivity between these two data objects along different paths based on their attributes. Specifically, *(Object #1* $\xrightarrow{dataType}$ *Pressure* $\xrightarrow{dataDiscipline}$ *Physical* $\xrightarrow{-dataDiscipline}$ *Density* $\xrightarrow{-dataType}$ *Object #2)*. KGs can thus be used to represent a facility's structured information as well as the relationships between its data objects.

C. Graph neural networks and recommendation systems

Recently, GNN-based KG recommendation models [12], [13], [20], [22] have achieved large performance improvements as compared to popular KG-based recommendation models. GNN-based methods can capture both the semantic representation of entities and relationships, and the collaborative signal (a.k.a. high-order connectivity information)

among them. Specifically, in a KG, first-order connectivity is the direct connection between items, representing a pre-existing feature. In contrast, high-order connectivity reflects the long-distance connections between items. For example, Figure 2a shows three subgraphs, the users and their location C (User–City), the interactions between users U and items I (User–Item), and interactions between items and their attributes A (Item–Attribute). Each subgraph itself represents first-order connectivity. When aligning them, we can combine the three subgraphs into a single collaborative graph, as shown in Figure 2b. A color depicts the relationship r between the entities. Thus, high-order connectivity is constructed via paths between two in-directly connected entities. For example, high-order connectivities from user U_1 to item I_2 are shown as follows:

- $\mathrm{U}_1 \xrightarrow{r_2} \mathrm{I}_1 \xrightarrow{r_3} \mathrm{A}_2 \xrightarrow{-r_3} \mathrm{I}_2$
- $\mathrm{U}_1 \xrightarrow{r_1} \mathrm{C}_1 \xrightarrow{-r_1} \mathrm{U}_2 \xrightarrow{r_2} \mathrm{I}_2$

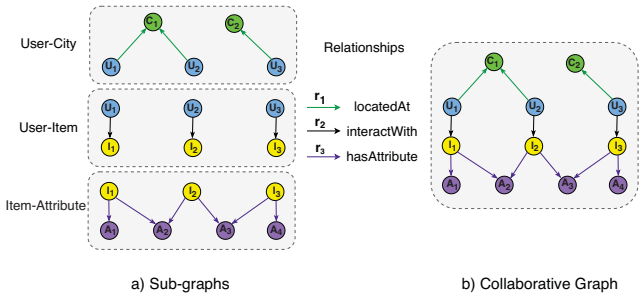


Fig. 2: A demonstration of high-order connectivity and the construction of a collaborative graph from subgraphs through entity alignment (please see in color).

Capturing high-order connectivity is essential for learning the facility's KG. The facility's data objects are connected via their attributes and other information in the KG. Thus two related data objects may be far from each other in the graph. However, standard KG-based methods, such as embedding-based methods [24], either do not consider or give insufficient attention to such long-distance correlation [14], [15]. Hence, to deliver high-recommendation performance, it is crucial to capture such high-order connectivity.

A GNN-based method can capture high-order connectivity because of its information-propagation mechanism. It generates an entity representation by aggregating messages from all neighbors and recursively performing such propagation to update the entity's embedding from its high-hop neighbors. However, GNN-based methods have the drawback that all neighbors are treated equally in GNNs. An entity may connect with multiple types of neighbors via various relations due to the heterogeneity of KGs. This inevitably introduces noise, regardless of the specific user–item interaction, thus limiting performance. To address this issue, recent work [12], [13] has integrated the attentive mechanism, which enables it to pay more attention to key information while reducing irrelevant noise. We leverage this optimization in our model.

III. An Analysis Large-Facility Data Usage

A. Exploring facility data-query behaviors

Large-scale facilities are designed for specific research domains. As a result, their instruments' location, data, and data-products' attributes are known. Furthermore, this structured information (i.e., metadata) can be collected from a facility's website and technical documentation.

Typical user queries for facility data are focused on science questions, and as a result, the queried data objects are associated with specific disciplines. For example, in oceanography, seawater *conductivity*, *temperature*, and *depth* are used to calculate seawater *salinity* and *density*. As a result, user queries for data objects are aligned with these domain-specific relationships, which in turn are part of the facilities' data model (e.g., see [25]). Another factor influencing user queries is geographical locality. For example, users are often interested in phenomena in a specific region and query data object in that region. Consequently, understanding domain knowledge (and associated data models) as well as the spatial distribution of instruments can help anticipate user queries.

B. Analyzing query patterns for OOI and GAGE

In this research we study user query behaviors for the OOI and GAGE large facilities. OOI [9], [10] is a networked ocean research observatory that deploys hundreds of instruments distributed across eight research arrays and across four oceans. The Geodetic Facility for the Advancement of Geoscience (GAGE) [11] is a nonprofit university-governed consortium that facilitates geoscience research and education using geodesy. It deploys more than 2,600 permanent GPS/GNSS stations in 90 countries, 75.9% of which are in the United States.

Our study is conducted using one-year-long user-query traces from OOI and GAGE with multi-million activity records (138 and 77 million records for the OOI and GAGE, respectively). Each trace record contains the user public IP address and its queried data object information. Although a public IP may represents multiple users from the same subnetwork, such as researchers from an institute, in this study, we regard this information as the user identity because we do not have access to additional user identification information due to privacy concerns. Moreover, we leverage the public IP to trace the user geographical location at city granularity. Some of the IPs can be further traced using additional information about the organization, such as for example, information about Rutgers University that we have. We use this additional tracing for our user similarity model.

Furthermore, we collect facility instrument metadata from the facilities' websites, including instrument name, coordinates, data type, and research discipline. Specifically, the OOI trace involves 36 instruments that are distributed at 55 sites across 8 research arrays; and the GAGE trace contains queries for 12 types of data from 2,106 permanent GPS/GNSS stations in the United States, which are distributed across 338 cities and 48 states. Furthermore, we represent this facility data using

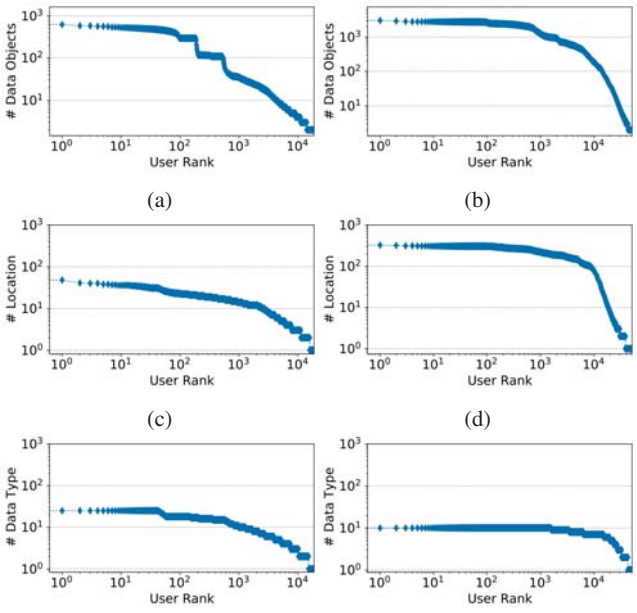


Fig. 3: Distribution curves of the OOI (left column) and GAGE (right column) user data queries characterized by number of data objects (a,b), number of instrument locations (c,d), and number of data types (e,f). For example, (a,b) show how many data objects has a user queried. The X-axis is the user ID.

the attributes *location* and *data type*, and plot the distribution of OOI and GAGE user queries in Figure 3.

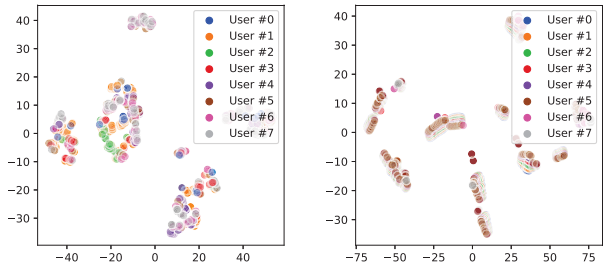
Inspired by the collaborative filtering technique [26], we model user-query patterns from *user* and *data item* perspectives as follows:

1) User similarity: In our analysis, we assume that users from the same research group are likely to query similar data because they may work on similar projects. Based on this assumption, we extract the eight users who have the most frequent data queries for OOI from the Rutgers University, and eight users for GAGE from the University of Washington. We then plot t-SNE figures of these data queries in terms of the instrument location and associated data attributes (e.g. data type), as shown in Figure 4.

The t-SNE [27] is a technique for visualizing high-dimensional data in a low-dimensional space (2D in this case) while preserving their local structure. Specifically, the distance between points in the figure represents their proximity in the high-dimension space. Therefore, adjacent points in the plot indicate that these data objects are similar.

As shown in Figure 4, the points cluster with overlaps across users, which indicates that queried data objects by users are similar. We have the same observation for other organizations. It shows that users from the same research group (or same organization) tend to have similar data-query patterns.

2) Instrument locality and data-domain affinity: Our analysis indicates that users typically focus on querying data from a specific region and related to a specific domain. If this



(a) Rutgers University. (b) University of Washington.

Fig. 4: Eight most frequent data-query t-SNE plots for OOI (a) and GAGE (b). Each dot in the plot is a user-queried data object, and the distance between dots represents data-object similarity.

observation is broadly accurate, leveraging it can improve recommendation accuracy. Analyzing the traces we find that, on average, users make 43.1% and 36.3% of their queries for data objects from instruments located in one region, and 51.6% and 68.8% of their queries are to the same data type for the OOI and GAGE, respectively. Considering that OOI and GAGE instruments are distributed across tens to hundreds of locations and each instruments provide tens of distinct data types, these results indicate a strong affinity of user queries to specific locations, instruments and data types.

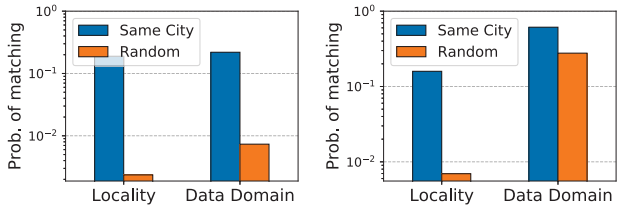
Often location information may only be available at larger granularity, e.g., a city. To verify whether users at city-proximity granularity have similar query patterns, we select two groups of 10,000 user pairs from OOI and GAGE, respectively. Within the first group, the two users in a pair are from the same city. By contrast, in the other group, users in a pair are randomly sampled. Then, we calculate the probability that two users in a group share the same query pattern in terms of instrument locality and data domain.

The results are presented in Figure 5 and show that users from the same location have a 79.8x and 29.8x (OOI) and a 22.87x and 2.21x (GAGE) greater likelihood to query data that was generated from the same region and belonged to the same data domain than the randomly selected users. These results illustrate that users from nearby locations (i.e., the same city or town, depending on the granularity of information available) have similar data-query patterns, which can be exploited for data recommendation.

IV. Collaborative Knowledge Graph Construction

We build our model using the three types of information discussed above: user-query traces, which provide user–item interactions; metadata and auxiliary information obtained from facility websites; and user-location information. Given our observation that users from the same location have similar data-query patterns, we group users by their location.

Following the formulations of state-of-the-art recommender models, we assume that there is a set of M users $\mathcal{U} = \{u_1, u_2, \ldots, u_M\}$ and a set of N items $\mathcal{V} = \{v_1, v_2, \ldots, v_N\}$



(a) OOI. (b) GAGE.

Fig. 5: Probability of two users having a similar data-query pattern in terms of instrument locality and data domain, in two cases: (1) they are from the same city; (2) they are randomly sampled.

in the OOI and GAGE traces. Using this notation, we can transform the three types of information into individual graphs, and then combine them into a CKG using entity alignment.

User-item bipartite graph: We represent user queries as a user-item bipartite graph $\mathcal{G}_1$, which is defined as $\{(u, y_{uv}, v)|u \in \mathcal{U}, v \in \mathcal{V}\}$, where $\mathcal{U}$ and $\mathcal{V}$ denote the user and item sets, and a link $y_{uv} = 1$ indicates that the user u has queried data item v; otherwise $y_{uv} = 0$.

Item-attribute graph: We typically have additional information describing the facility data, such as coordinates, sensor type, etc. We organize these data attributes in the form of a KG, $\mathcal{G}_2 = \{(h, r, t)|h, t \in \mathcal{E}, r \in \mathcal{R}\}$, where each knowledge triple (h, r, t) denotes that there is a relationship r between the head entity h and the tail entity t, and $\mathcal{E}$ and $\mathcal{R}$ are the sets of entities and relations in the KG $\mathcal{G}_2$. For example, the triple *(Bottom Pressure and Tilt Meter*, *Measure*, *Pressure)* states the fact that the instrument *Bottom Pressure and Tilt Meter* can measure the *pressure* data. Note that $\mathcal{R}$ contains relations in both in the canonical direction *(e.g., Measure)* and in the inverse direction *(e.g., MeasuredBy)*.

User-user bipartite graph: We represent the user-user associations using the graph $\mathcal{G}_3 = \{(u_i, y_{uu}, u_j)|(u_i, u_j) \in \mathcal{U}\}$, where $\mathcal{U}$ denotes the users and a link $y_{uu} = 1$ indicates that the user u_i and user u_j are in the same location; otherwise $y_{uu} = 0$.

Collaborative Knowledge Graph (CKG): We combine the three subgraphs into a CKG $\mathcal{G}$ using entity alignment. First, we integrate the *user-item* ($\mathcal{G}_1$) and *user-user* ($\mathcal{G}_3$) bipartite graphs together into $\mathcal{G}$ by aligning the user u. We represent each interaction as a triple, *(u, interact, v)* and *(u, interact, u)*, where $y_{uv} = 1$ and $y_{uu} = 1$ are represented as an additional relationship *Interact* between user u and item v, and between users. Then, in order to combine the *item-attribute* KG ($\mathcal{G}_2$), we employ a set of item–entity alignments $\mathcal{A} = \{(v, e)|v \in \mathcal{V}, e \in \mathcal{E}\}$, where (v, e) indicates that item v can be aligned with entity e in the KG $\mathcal{G}_2$. By aligning entity e in $\mathcal{G}_2$ to the item v in $\mathcal{G}$ according to $\mathcal{A}$, we can integrate the *item-attribute* graph $\mathcal{G}_2$ into the CKG $\mathcal{G} = \{(h, r, t)|h, t \in \mathcal{E}', r \in \mathcal{R}'\}$, where $\mathcal{E}' = \mathcal{E} \cup \mathcal{U} \cup \mathcal{V}$ and $\mathcal{R}' = \mathcal{R} \cup \{Interact\}$.

The CKG is flexible allowing the addition of new entities, such as data objects and knowledge sources. Using entity alignment, KGs from multiple facilities can be consolidated. This can potentially enable recommendations across multiple facilities. However, we do not explore this aspect in the paper.

Recommendation task formulation: We formulate the recommendation task as follows:

- **Input**: the collaborative knowledge graph $\mathcal{G}$ that includes the user–item bipartite graph $\mathcal{G}_1$, knowledge graph $\mathcal{G}_2$, and the user–user bipartite graph $\mathcal{G}_3$.
- **Output**: a prediction function that predicts the probability $\hat{y}_{uv}$ that user u will query item v.

V. Design of the Recommendation Model

Inspired by existing studies [12], [13], [20], [22], we propose a recommendation model called the collaborative knowledge-graph attention network (CKAT). Figure 6a presents its architecture, which consists of three components: (1) an embedding layer, which initializes and parameterizes each node on the CKG using a vector representation; (2) a knowledge-aware attentive embedding propagation layer that refines each node's representation by aggregating messages from its neighborhoods in the CKG and applies an knowledge-aware attention mechanism to learn the varying importance of each neighbor during a propagation; and (3) a prediction layer, which outputs the user–item pair prediction score through estimating the likelihood of an interaction based on the final representation. This three components are discussed below.

A. Embedding layer

The embedding layer aims to learn the structured representation of the KG. Translation-based methods [28], [29] are widely used for embedding graphs; here, we apply the TransR [29] method. Given a triple (h, r, t) as an example, its embeddings are $\mathbf{e}_h, \mathbf{e}_t \in \mathbb{R}^d$ and $\mathbf{e}_r \in \mathbb{R}^k$. TransR learns and embeds the entities and relationships by optimizing the translation principle $\mathbf{e}_h^r + \mathbf{e}_r \approx \mathbf{e}_t^r$, where $\mathbf{e}_h^r$, $\mathbf{e}_t^r$ are the projected representations of $\mathbf{e}_h$ and $\mathbf{e}_t$ in the relationship r's space. Its score function is formulated as follows:

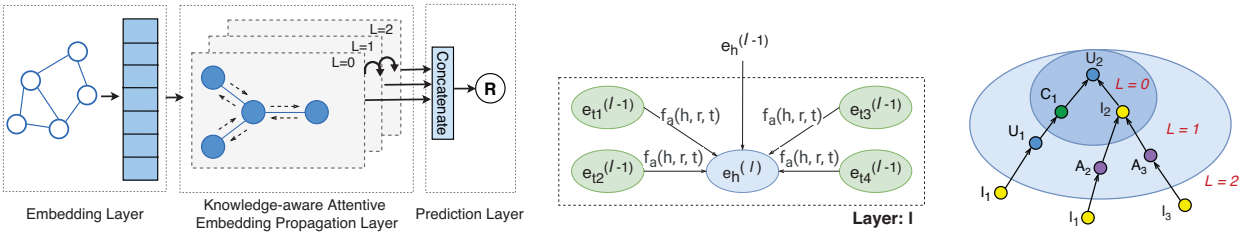
$$f_r(h, r, t) = ||\mathbf{W}_r\mathbf{e}_h + \mathbf{e}_r - \mathbf{W}_r\mathbf{e}_t||_2^2 \tag{1}$$

in which $\mathbf{W}_r \in \mathbb{R}^{k \times d}$ is the transformation matrix for relationship r. It projects entities from the d-dimension entity space into the k-dimension relationship space. A lower score of $f_r(h, r, t)$ indicates that the triple is more likely to be true, and vice versa.

Following [29], we use the following margin-based score function as the objective for training TransR:

$$\mathcal{L}_1 = \sum_{(h,r,t) \in \mathcal{S}} \sum_{(h,r,t') \in \mathcal{S}'} \max\left(0, f_r(h, r, t) + \gamma - f_r(h, r, t')\right) \tag{2}$$

where $\mathcal{S}$ is the set of correct triples in the CKG, and $\mathcal{S}'$ is the set of broken triplets that is constructed by replacing one entity in a valid triple randomly; $\max(\cdot)$ is the maximum function, and γ is the margin.



(a) CKAT recommendation model architecture.

(b) Knowledge-aware attentive embedding propagation layer.

(c) Information propagation example.

Fig. 6: The CKAT recommendation model.

B. Knowledge-aware attentive embedding propagation layer

Based on the graph convolution network architecture [30] and leveraging graph attention networks [31], we build this layer to recursively propagate embeddings along with high-order connectivity. We also generate different attentive weights for cascaded propagation to reveal the importance of such connectivity. Here, we start with the description of a single layer, as shown in Figure 6b, and then offer a discussion of how to stack them across multiple layers.

Information propagation: An entity in the graph has direct or high-order connections with its neighbors. To demonstrate the information propagation among them, we employ user U_2 in Figure 6c as an example. There are three propagation paths. On one of them, item I_2 takes attributes A_2 and A_3 as inputs to enrich its features and then contributes user U_2's preferences, which can be simulated by propagating information from A_2 to U_2. Based on this intuition, we use $\mathcal{N}_h = \{(h,r,t)|(h,r,t) \in \mathcal{E}\}$ to denote the set of triplets in which h is the head entity and formalize the information being propagated from its neighbors to h as follows:

$$\mathbf{e}_{\mathcal{N}_h} = \sum_{(h,r,t)\in\mathcal{N}_h} f_a(h,r,t)\mathbf{e}_t \tag{3}$$

where $f_a(h,r,t)$ is the attention component that controls the decay factor on each propagation on edge (h,r,t), indicating the contributions of t to h conditioned to relationship r.

Knowledge-aware attention: We implement $f_a(h,r,t)$ via the relational attention mechanism, which is formulated as follows:

$$f_a(h,r,t) = (\mathbf{W}_r\mathbf{e}_t)^\top \tanh\left(\mathbf{W}_r\mathbf{e}_h + \mathbf{e}_r\right) \tag{4}$$

where tanh is used as a nonlinear activation function. For simplicity, here we consider only the inner product to obtain the attention weights, which reflects the affinity between two entities e_h and e_t in relationship r's space. Hereafter, we employ the softmax function to normalize the attention weights across all neighbors, which is formulated as follows:

$$f_a(h,r,t) = \frac{\exp\left(f_a(h,r,t)\right)}{\sum_{(h,r',t')\in\mathcal{N}_h}\exp\left(f_a(h,r',t')\right)} \tag{5}$$

where the final attention scores can distinguish varying importance scores of neighbors.

Information aggregation: In this phase, we utilize the entity representation $\mathbf{e}_h$ and the information being propagated from its neighbors $\mathbf{e}_{\mathcal{N}_h}$ to update the representation of entity h, as $\mathbf{e}_h^{(1)} = agg(\mathbf{e}_h, \mathbf{e}_{\mathcal{N}_h})$. In this study, we implement the aggregation function $agg(\cdot)$ using the following two methods:

The *concatenate aggregation* method concatenates two representations, followed by a nonlinear transformation:

$$agg_{concat} = \text{LeakyReLU}(\mathbf{W}(\mathbf{e}_h||\mathbf{e}_{\mathcal{N}_h})) \tag{6}$$

where $||$ is the concatenation operation, $\mathbf{W} \in \mathbb{R}^{d'\times d}$ are the trainable weight matrices for distilling useful information for propagation, and d' is the transformation size.

The *sum aggregation* method sums two representations and applies a nonlinear transformation, as follows:

$$agg_{sum} = \text{LeakyReLU}(\mathbf{W}(\mathbf{e}_h + \mathbf{e}_{\mathcal{N}_h})) \tag{7}$$

High-order propagation: Building on previous efforts [12], [13], [20], we can stack more information-propagation layers to exploit the high-order connectivity inherent in the collaborative KG. For example, Figure 6b illustrates the propagation in the l-th steps that recursively update the representation of entity h by the previous representations of itself and its neighbors with the following formulation:

$$\mathbf{e}_h^{(l)} = agg(\mathbf{e}_h^{(l-1)}, \mathbf{e}_{\mathcal{N}_h}^{(l-1)}) \tag{8}$$

$$\mathbf{e}_{\mathcal{N}_h}^{(l-1)} = \sum_{(h,r,t)\in\mathcal{N}_h} f_a(h,r,t)\mathbf{e}_t^{(l-1)} \tag{9}$$

$\mathbf{e}_t^{(l-1)}$ is the representation of entity t generated from the previous information-propagation steps, memorizing the information from its $(l-1)$-hop neighbors, where same to $\mathbf{e}_h^{(l-1)}$. This propagation allows an entity to contribute to another entity's representation up to l-hops away. As a result, each entity's representation captures and embeds the connectivity from its high-order neighbors. Such as a path in Figure 6c, $I_1 \rightarrow U_1 \rightarrow C_1 \rightarrow U_2$ indicates that the information from I_1 is propagated and embedded in $\mathbf{e}_{u2}^{(3)}$.

C. Model prediction

Assuming the number of information propagation layers is L, at the end, we produce multiple representations of each node, such as user node $\{\mathbf{e}_u^{(1)}, \cdots, \mathbf{e}_u^{(L)}\}$. Because each

representation emphasizes different orders of neighbors, we concatenate them into a single vector, as follows:

$$\mathbf{e}_u^* = \mathbf{e}_u^{(0)}||\cdots||\mathbf{e}_u^{(L)},\ \mathbf{e}_v^* = \mathbf{e}_v^{(0)}||\cdots||\mathbf{e}_v^{(L)} \tag{10}$$

where $||$ is the concatenation operation, u is users, and v is items.

Finally, we calculate the inner product of user and item representations, so as to predict their matching score:

$$\hat{y}(u,v) = \mathbf{e}_u^{*\top}\mathbf{e}_v^* \tag{11}$$

D. Optimization

Following mainstream optimization methods [12], [13], [32], we adopt Bayesian personalized ranking (BPR) [33] to optimize the model parameters. BPR assumes that users prefer items they have interacted with before, which indicates that the observed interactions should be assigned higher prediction values than unobserved ones:

$$\mathcal{L}_2 = \sum_{(u,I,j)\in\mathcal{O}} -\ln\sigma(\hat{y}(u,i) - \hat{y}(u,j)) \tag{12}$$

where $\mathcal{O} = \{(u,i,j)|(u,i)\in\mathcal{R}^+,(u,j)\in\mathcal{R}^-\}$ denotes the training set, $\mathcal{R}^+$ indicates the observed (positive) interactions between user u and item j, and $\mathcal{R}^-$ is the sampled unobserved (negative) interaction set; $\sigma(\cdot)$ is the sigmoid function.

Finally, we represent the objective function as follows:

$$\mathcal{L}_{\text{CKAT}} = \mathcal{L}_1 + \mathcal{L}_2 + \lambda||\Theta||_2^2 \tag{13}$$

where Θ indicates the parameters used in the model and σ is the decay factor on Θ to prevent overfitting.

VI. Experimental Evaluation

In this section, we use two user-query traces from OOI and GAGE to evaluate the performance of the CKAT recommendation model presented in the previous sections by answering the following research questions:

- **RQ1**: How does the CKAT model perform on the facility data as compared to the state-of-the-art knowledge-aware models?
- **RQ2**: How do the CKG model and its components impact the recommendation results?
- **RQ3**: How do the attention mechanism and other hyper-parameter settings (i.e., depth of the knowledge-aware propagation layer, aggregation selection) impact the recommendation results?

A. Dataset description

We first preprocess the query traces to extract key information, building on the mechanisms used by existing efforts for benchmark datasets, e.g., MovieLens [34]. We then construct the CKG from the three subgraphs: The user–item graph (UIG) is constructed on the basis of user and data item interactions, extracted from the OOI and GAGE traces. The user–user graph (UUG) contains user association information obtained by clustering users based on their proximity (i.e., the same organization, physical location, etc.). The item–attribute graph (IAG) contains two attributes, instrument location (LOC) and data-domain knowledge (DKG), which are obtained from the OOI and GAGE websites. Furthermore, we combine other attributes available at the facility websites such as instrument metadata (MD) including instrument names and associated groups. As some of this information is not directly relevant to user data-query patterns (see Section III-B2), we regard it as noise when evaluating the impact on recommendation performance. Table I lists the basic CKG information.

	OOI	GAGE
# entities	1,342	4,754
# relationships	8	7
# KG triplets	5,554	20,314
# link-avg	6	10

TABLE I: Statistics for the OOI and GAGE collaborative knowledge graphs (CKG). The term "link-avg" refers to the average links per item.

For each dataset, we randomly select 80% of each user's query history for the training set and treat the remaining percentage as the test set. For each observed user–item interaction, we consider it as a positive instance and then conduct the negative sampling strategy to pair it with one negative item that the user did not consume before.

B. Metrics

We use the top-K measurement [35] to evaluate the effectiveness of the recommendations. Furthermore, we adopt two widely used evaluation protocols: recall@K and ndcg@K. By default, we set K = 20.

C. Baseline models

To demonstrate the effectiveness of our approach, we compare our proposed CKAT model against the following state-of-the-art baseline models: collaborative-filtering-based (BPRMF), supervised-learning (FM and NFM), regularization-based (CKE, CFKG), and graph-convolutional-network-based (RippleNet, KGCN) models.

- **BPRMF** [33] is a collaborative-filtering-based method using pairwise matrix factorization for item recommendation from implicit feedback, optimized by the Bayesian personalized ranking loss.
- **FM** [36] is a factorization-based method that uses second-order feature interactions between inputs. Here, we convert the user IDs, data objects, and CKG entities as the input features.
- **NFM** [37] is a factorization-based method that subsumes FM under a neural-network framework. As suggested by He at al. [37], we employ one hidden layer on input features.
- **CKE** [24] is a regularization-based method that applies TransR [38] for semantic embeddings.
- **CFKG** [39] is a regularization-based method that applies TransE [28] to embed the unified graph, including heterogeneous multitype user behaviors and knowledge of the items.

- **RippleNet** [22] is a propagation-based model that refines an entity's representation through sampling ripple sets from its neighbors.
- **KGCN** [20] is a propagation-based model that extends nonspectral graph convolutional network approaches [40] to aggregate and incorporate neighborhood information with bias when calculating the entity representation in the KG.

D. Parameter settings

We implement the CKAT model in Tensorflow. The embedding size is fixed at 64 for all models except RippleNet, for which it is set to 16 due to RippleNet's computational complexity. We optimize all models with the Adam optimizer [41], where the batch size is fixed at 512. Furthermore, we use the default Xavier initializer [42] to initialize the model parameters. We apply a grid search for hyperparameters: the learning rate is tuned to values in $\{0.05, 0.01, 0.005, 0.001\}$, the coefficient for L_2 normalization is searched within the set $\{10^{-5}, 10^{-4}, \cdots, 10^{1}, 10^{2}\}$, and the dropout ratio is tuned to value in $\{0.0, 0.1, \cdots, 0.8\}$ for NFM and CKAT. We set the depth of CKAT, L, as 3 with hidden dimension 64, 32, 16, respectively. We set the RippleNet n_hop=2, which is its propagation-layer number. By default, we use the *concatenate* aggregator, three propagation layers, and CKG.

E. Performance comparison (*RQ1*)

Table II summarizes the performance results for all the models using the OOI and GAGE datasets and using the CKG as auxiliary information. CKAT consistently yields the best performance in all cases. We have the following observations from the experimental results:

- Compared to the baseline models, CKAT improves performance for both, OOI and GAGE datasets. Specifically, CKAT improves *recall* by over 6.1237% and 5.7399%, and *ndcg* by over 7.2624% and 6.0496%, for OOI and GAGE, respectively.
- The performance of the propagation-based methods, RippleNet and KGCN, is comparable to that of the CKAT, because they can capture high-order connectivity in the KG. Moreover, the results also justify the effectiveness of the knowledge-aware attention mechanism. It allows the CKAT to distinguish different entity relationships, reduce noise, and thus focus on learning key information.
- The CF-based method (BPRMF) is outperformed by most of the KG-based methods, especially the propagation-based methods. This demonstrates the effectiveness of using KG as auxiliary information for improving recommendation performance.
- FM and NFM outperform the regularization-based methods (CKE and CFKG), which indicates the importance of capturing high-order connectivity. FM and NFM exploit the embeddings of its neighbor entities, which can serve as the second-order connectivity. However, CKE and CFKG model connectivity only on triples' granularity, which is equivalent to only the first-order relationship in the KG. Moreover, it is worth noting that BPRMF performs better than CKE and CFKG in the GAGE case. The ability to model high-order connectivity is essential for recommending facility data.
- Overall, the propagation-based methods perform the best because they can fully exploit the high-order information on the KG. Moreover, the knowledge-aware attention mechanism allows the CKAT to improve performance by paying more attention to key information during the propagation process.

	OOI		GAGE	
	recall@20	ndcg@20	recall@20	ndcg@20
BPRMF	0.1935	0.1693	0.2742	0.2115
FM	0.2353	0.2228	0.3174	0.2356
NFM	0.2339	0.2211	0.3289	0.2471
CKE	0.2102	0.2197	0.2675	0.2106
CFKG	0.2283	0.2241	0.2572	0.2096
RippleNet	0.2833	0.2394	0.3584	0.2981
KGCN	**0.3020**	**0.2414**	**0.3767**	**0.3106**
CKAT	**0.3217**	**0.2561**	**0.4062**	**0.3306**
% Impro.	6.1237	5.7399	7.2624	6.0496

TABLE II: Overall performance comparison.

F. Evaluation of knowledge-source combinations (*RQ2*)

The CKG is constructed from three subgraphs – user–item graph (UIG), user–user graph (UUG), and item–attribute graph (IAG). The IAG includes knowledge such as instrument location (LOC) and data-domain knowledge (DKG). In this experiment, we evaluate the impact of such knowledge sources by evaluating different knowledge combinations. Moreover, as irrelevant knowledge sources would negatively impact the propagation-based method (they are the noise when the model is learning the entity representation through its neighbors), in this experiment we use the additional instrument metadata (MD) as noise to demonstrate the importance of selecting knowledge.

Table III presents the results. When combing the UIG with one more sources of knowledge (i.e., LOC, DKG, UUG), the performance varies. UIG+DKG has better performance for OOI, whereas UIG+LOC is better for GAGE. It reveals the characteristics of the different facility user communities. In this case, OOI users would query data with a stronger focus on the domain model, whereas GAGE users would tend to follow the instrument–locality correlation.

When all the knowledge is stacked together *UIG+UUG+LOC+DKG*, we achieve the best performance for both traces. This indicates that they are the most relevant information to characterize the facility user query patterns.

Furthermore, when adding MD (i.e., noise) to the best knowledge combination, the performance decreases. This shows that collecting the right knowledge is essential to providing optimal recommendation performance. Because the facility provides abundant structured meta-data information, a careful selection of information is needed.

To fine-tune the CKG for each facility, we can try different knowledge combinations, as in the process demonstrated

	OOI		GAGE	
	recall@20	ndcg@20	recall@20	ndcg@20
UIG+LOC	0.2675	0.2322	0.3848	0.3191
UIG+DKG	0.2844	0.2424	0.3643	0.3148
UIG+UUG	0.2756	0.2364	0.3543	0.3048
UIG+LOC+DKG	0.3074	0.2527	0.3943	0.3148
UIG+UUG+LOC+DKG	**0.3217**	**0.2561**	**0.4062**	**0.3306**
UIG+UUG+LOC+DKG+MD	0.3197	0.2511	0.4011	0.3276

TABLE III: Results for different knowledge graph inputs. UIG is the user–item graph, and UUG is the user–user graph. Here we further extract the instrument location (LOC) and data-domain knowledge (DKG) from the item-attribute graph (IAG) and evaluate it separately. Additional instrument metadata (MD) is considered noise information.

	OOI		GAGE	
	recall@20	ndcg@20	recall@20	ndcg@20
w/ Att + agg_{concat}	0.3217	0.2561	0.4062	0.3306
w/ Att + agg_{sum}	0.3120	0.2409	0.3894	0.3123
w/o Att + agg_{concat}	0.2994	0.2331	0.3755	0.3147

TABLE IV: Effect of attention mechanism (Att), *concatenate* and *sum* aggregators on recommendation performance. The first row represents the default CKAT setup.

above. However, when the facility adds new instruments or data objects, the fine-tuning process needs to be repeated. This is a limitation that we will address in the future.

G. Impact study of each component of the CKAT (RQ3)

CKAT exploits the neural attention mechanism to assign different weights to different entities in order to reduce noise and focus more attention on key information during the propagation process. To analyze its impact on the recommendation results, we keep the best practice model parameter settings and use CKG as the input, and then compare recommendations with and without the attention mechanism. Results presented in Table IV demonstrate that the CKAT with the attention mechanism performs better than CKAT without it.

Moreover, we evaluate CKAT under two aggregator settings, *concatenate* and *sum*. As Table IV shows, agg_{concat} performs better than agg_{sum} for both OOI and GAGE. One possible reason is that agg_{concat} can retain more hidden information in embeddings, which improves entity representation learning.

As capturing higher-order connectivity is a key advantage of the CKAT model, we investigate the efficiency of using multiple embedding propagation layers. In this experiment, we consider 1, 2 and 3 layers, and CKAT-1 refers to the model using one layer.

The results in Table V show that increasing the depth of CKAT can boost its performance. CKAT-3 and CKAT-2 consistently achieve an improvement over CKAT-1 across the board. This implies that CKAT can effectively capture high-order relationships between entities carried by the second- and third-order connectivity. Additionally, we observe a larger improvement from CKAT-2 to CKAT-3 for GAGE as compared to OOI. Since the size of the CKG for GAGE is larger than that for OOI, it may need to stack more layers when expanding the CKG to exploit it fully.

	OOI		GAGE	
	recall@20	ndcg@20	recall@20	ndcg@20
CKAT-1	0.3108	0.2471	0.3736	0.3118
CKAT-2	0.3209	0.2478	0.3821	0.3215
CKAT-3	0.3217	0.2561	0.3919	0.3278

TABLE V: Impact of using different number of embedding propagation layers, *L*.

VII. Conclusion

In this paper, we explored the use of recommendation systems to address the data-discovery and data-access challenges faced by large-scale scientific facilities, such as instruments, experimental platforms, and observatories. We first analyzed user-query traces from two existing facilities, OOI and GAGE, and analyzed the access patterns observed in terms of facility-instrument locality, domain-specific data model, and user association. Based on this analysis, we combined key information that characterizes the data-query patterns into a collaborative knowledge graph (CKG). We then constructed the collaborative knowledge-aware graph attention network (CKAT) recommendation model, which leverages the graph neural network (GNN) to explicitly encode the collaborative signals through propagation and combine it with knowledge associations. To reduce the irrelevant knowledge in the KG, which brings noise to the entity representation learning process, we integrated a knowledge-aware neural attention mechanism into CKAT. This enabled CKAT to pay more attention to key information. The empirical evaluation presented in the paper demonstrates that CKAT can effectively facilitate data discovery and access and that it significantly outperforms several compelling state-of-the-art baseline models.

The overall approach presented in this paper has broad applications, such as enabling the "intelligent" discovery and anticipatory delivery of data and data products from large facilities. Furthermore, the CKG can integrate knowledge from many sources and can quickly grow in scale. As a result, the parallelization of the CKAT model and the use of (in situ) purposeful accelerators are important areas for future work.

Acknowledgments

This research is supported in part by NSF via grants numbers OAC 1835692, OAC 1826997, and OAC 1640834, and was conducted as part of the Rutgers Discovery Informatics Institute (RDI^2). This material is based in part on services provided by the GAGE Facility, operated by UNAVCO, Inc., with support from NSF and the National Aeronautics and Space Administration under NSF Cooperative Agreement EAR-1724794 and NSF grant OAC 1835791.

References

[1] I. Rodero and M. Parashar, "Data cyberinfrastructure for end-to-end science," *Computing in Science Engineering*, vol. 22, no. 5, pp. 60–71, 2020.

[2] B. P. Abbott, R. Abbott, T. Abbott, M. Abernathy, F. Acernese, K. Ackley, C. Adams, T. Adams, P. Addesso, R. Adhikari *et al.*, "Observation of gravitational waves from a binary black hole merger," *Physical review letters*, vol. 116, no. 6, p. 061102, 2016.

[3] K. Akiyama, A. Alberdi, W. Alef, K. Asada, R. Azulay, A.-K. Baczko, D. Ball, M. Baloković, J. Barrett, D. Bintley *et al.*, "First m87 event horizon telescope results. iv. imaging the central supermassive black hole," *The Astrophysical Journal Letters*, vol. 875, no. 1, p. L4, 2019.

[4] K. Fauvel, D. Balouek-Thomert, D. Melgar, P. Silva, A. Simonet, G. Antoniu, A. Costan, V. Masson, M. Parashar, I. Rodero *et al.*, "A distributed multi-sensor machine learning approach to earthquake early warning," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 403–411.

[5] M. Parashar, V. Honavar, A. Simonet, I. Rodero, F. Ghahramani, G. Agnew, and R. Jantz, "The virtual data collaboratory," *Computing in Science & Engineering*, 2019.

[6] Y. Qin, A. Simonet, P. E. Davis, A. Nouri, Z. Wang, M. Parashar, and I. Rodero, "Towards a smart, internet-scale cache service for data intensive scientific applications," in *Proceedings of the 10th Workshop on Scientific Cloud Computing*, 2019, pp. 11–18.

[7] I. Rodero, Y. Qin, J. Valls, A. Simonet, J. Villalobos, M. Parashar, C. Youn, C. Wang, K. Thareja, P. Ruth *et al.*, "Enabling data streaming-based science gateways through federated cyberinfrastructure," *Gateways 2019*, 2019.

[8] [Online]. Available: https://www.nsf.gov/bfa/lfo/docs/major-facilities-list.pdf

[9] L. M. Smith, J. A. Barth, D. S. Kelley, A. Plueddemann, I. Rodero, G. A. Ulses, M. F. Vardaro, and R. Weller, "The ocean observatories initiative," *Oceanography*, vol. 31, no. 1, pp. 16–35, 2018.

[10] I. Rodero and M. Parashar, "Architecting the cyberinfrastructure for National Science Foundation Ocean Observatories Initiative (OOI)," *7th International Workshop on Marine Technology: MARTECH 2016*, pp. 99–101, 2016.

[11] [Online]. Available: http://www.unavco.org/

[12] X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua, "Kgat: Knowledge graph attention network for recommendation," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 2019, pp. 950–958.

[13] Z. Wang, G. Lin, H. Tan, Q. Chen, and X. Liu, "Ckan: Collaborative knowledge-aware attentive network for recommender systems," in *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2020, pp. 219–228.

[14] Q. Guo, F. Zhuang, C. Qin, H. Zhu, X. Xie, H. Xiong, and Q. He, "A survey on knowledge graph-based recommender systems," *arXiv preprint arXiv:2003.00911*, 2020.

[15] S. Ji, S. Pan, E. Cambria, P. Marttinen, and P. S. Yu, "A survey on knowledge graphs: Representation, acquisition and applications," *arXiv preprint arXiv:2002.00388*, 2020.

[16] L. Weston, V. Tshitoyan, J. Dagdelen, O. Kononova, A. Trewartha, K. A. Persson, G. Ceder, and A. Jain, "Named entity recognition and normalization applied to large-scale information extraction from the materials science literature," *Journal of chemical information and modeling*, vol. 59, no. 9, pp. 3692–3702, 2019.

[17] A. M. Hiszpanski, B. Gallagher, K. Chellappan, P. Li, S. Liu, H. Kim, J. Han, B. Kailkhura, D. J. Buttler, and T. Y.-J. Han, "Nanomaterial synthesis insights from machine learning of scientific articles by extracting, structuring, and visualizing knowledge," *Journal of Chemical Information and Modeling*, 2020.

[18] N. Mukund, S. Thakur, S. Abraham, A. Aniyan, S. Mitra, N. S. Philip, K. Vaghmare, and D. Acharjya, "An information retrieval and recommendation system for astronomical observatories," *The Astrophysical Journal Supplement Series*, vol. 235, no. 1, p. 22, 2018.

[19] M. Barros, A. Moitinho, and F. M. Couto, "Using research literature to generate datasets of implicit feedback for recommending scientific items," *IEEE Access*, vol. 7, pp. 176 668–176 680, 2019.

[20] H. Wang, M. Zhao, X. Xie, W. Li, and M. Guo, "Knowledge graph convolutional networks for recommender systems," in *The world wide web conference*, 2019, pp. 3307–3313.

[21] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *arXiv preprint arXiv:1901.00596*, 2019.

[22] H. Wang, F. Zhang, J. Wang, M. Zhao, W. Li, X. Xie, and M. Guo, "Ripplenet: Propagating user preferences on the knowledge graph for recommender systems," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, 2018, pp. 417–426.

[23] H. Wang, F. Zhang, X. Xie, and M. Guo, "Dkn: Deep knowledge-aware network for news recommendation," in *Proceedings of the 2018 world wide web conference*, 2018, pp. 1835–1844.

[24] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W.-Y. Ma, "Collaborative knowledge base embedding for recommender systems," in *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016, pp. 353–362.

[25] [Online]. Available: https://oceanobservatories.org/instrument-class/ctd/

[26] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proceedings of the 10th international conference on World Wide Web*, 2001, pp. 285–295.

[27] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.

[28] 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*, 2013, pp. 2787–2795.

[29] H. Lin, Y. Liu, W. Wang, Y. Yue, and Z. Lin, "Learning entity and relation embeddings for knowledge resolution," *Procedia Computer Science*, vol. 108, pp. 345–354, 2017.

[30] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[31] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[32] Z. Tao, Y. Wei, X. Wang, X. He, X. Huang, and T.-S. Chua, "Mgat: Multimodal graph attention network for recommendation," *Information Processing & Management*, vol. 57, no. 5, p. 102277, 2020.

[33] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "Bpr: Bayesian personalized ranking from implicit feedback," *arXiv preprint arXiv:1205.2618*, 2012.

[34] [Online]. Available: https://grouplens.org/datasets/movielens/

[35] R. Fagin, R. Kumar, and D. Sivakumar, "Comparing top k lists," *SIAM Journal on discrete mathematics*, vol. 17, no. 1, pp. 134–160, 2003.

[36] S. Rendle, Z. Gantner, C. Freudenthaler, and L. Schmidt-Thieme, "Fast context-aware recommendations with factorization machines," in *Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval*, 2011, pp. 635–644.

[37] X. He and T.-S. Chua, "Neural factorization machines for sparse predictive analytics," in *Proceedings of the 40th International ACM SIGIR conference on Research and Development in Information Retrieval*, 2017, pp. 355–364.

[38] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," in *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[39] Q. Ai, V. Azizi, X. Chen, and Y. Zhang, "Learning heterogeneous knowledge base embeddings for explainable recommendation," *Algorithms*, vol. 11, no. 9, p. 137, 2018.

[40] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016.

[41] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[42] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, 2010, pp. 249–256.