Transparent Learner Knowledge State Modeling using Personal Knowledge Graphs and Graph Neural Networks

RAWAA ALATRASH, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Germany
MOHAMED AMINE CHATTI, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Germany
QURAT UL AIN, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Germany
SHOEB JOARDER, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Germany

Learner modeling is pivotal in different applications of adaptive and personalized systems in the educational domain, such as recommender systems and intelligent tutoring systems. However, how learner models are inferred and used in the these systems are often not transparent to learners. In many cases, learner models are presented as a black-box, where learners have no means to control or modify their models. To address these issues, in this paper, we present an innovative approach to learner modeling, particularly focusing on modeling learners' knowledge states. To this end, we combine Personal Knowledge Graphs (PKGs), Graph Convolutional Networks (GCNs), and transformer sentence encoders (SBERT) to construct a transparent learner model. Specifically, we explicitly involve learners in modeling their knowledge state by enabling them to mark concepts as 'Did Not Understand' (DNU) in the MOOC platform CourseMapper. This results in the construction of a user-controllable and scrutable PKG for the learner, thus increasing the transparency of the learner modeling process. Furthermore, we leverage GCNs and SBERT to model the learner knowledge state based on an enhanced representation of their DNUs. In this way, we provide a simple yet effective method for learner modeling which can be used to improve performance in downstream tasks, such as adaptive systems, recommendation, and personalized search.

CCS Concepts: • **Applied computing → E-learning**; • **Information systems → Personalization**.

Additional Key Words and Phrases: User modeling, Transparent Learner Model, Graph Neural Network, Representation learning, Knowledge Graphs, Graph Convolutional Networks, Sentence Encoders

ACM Reference Format:
Rawaa Alatrash, Mohamed Amine Chatti, Qurat Ul Ain, and Shoeb Joarder. 2024. Transparent Learner Knowledge State Modeling using Personal Knowledge Graphs and Graph Neural Networks. In *Adjunct Proceedings of the 32nd ACM Conference on User Modeling, Adaptation and Personalization (UMAP Adjunct '24), July 1–4, 2024, Cagliari, Italy*. ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3631700.3665230

Authors' addresses: Rawaa Alatrash, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Duisburg, Germany, rawaa.alatrash@stud.uni-due.de; Mohamed Amine Chatti, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Duisburg, Germany, mohamed.chatti@uni-due.de; Qurat Ul Ain, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Duisburg, Germany, qurat.ain@stud.uni-due.de; Shoeb Joarder, Social Computing Group, Faculty of Computer Science, University of Duisburg-Essen, Duisburg, Germany, shoeb.joarder@uni-due.de.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.
Manuscript submitted to ACM

1 INTRODUCTION

Learner modeling is a fundamental concept in online learning environments to provide learners with personalized services. It involves creating a representation of a learner's characteristics, preferences, and behaviors, called a learner model. learner models typically capture learners' abilities and behaviors by analyzing their actions and interactions with the learning system [19]. Among a learner's various attributes, their knowledge state stands out as a crucial element within the learner model [14]. Most existing studies model learner knowledge states implicitly based on learners' interactions with the system. However, these systems often lack transparency and user control [3]. Several studies integrated Open Learner Models (OLMs) in online learning environments to improve transparency by providing learners with insights into their generated learner models [13]. The used OLMs, however, commonly operate as a black box and give learners no insight into how they internally work to estimate their knowledge states. Moreover, these OLMs do not allow learners to scrutinize (i.e., correct or modify) their models to align with their interests [6].

Learners' interactions with an educational system can be represented as a graph. Recent studies have demonstrated the capability of Knowledge Graphs (KGs) to model the user's interactions with the system to generate a user profile [47]. However, existing systems often lack user-specific data, limiting their ability to provide truly personalized content [36]. This is where Personal Knowledge Graphs (PKGs) play an important role by focusing on user-relevant entities from the KG and structuring this data around a specific user. Although PKGs are primarily used in the health domain, their implementation in education remains limited. Since users, items, and preferences can be modeled as graphs, Graph Neural Networks (GNNs), particularly Graph Convolutional Networks (GCNs), offer a powerful technique to encode collaborative information to get the representation of user models [32]. Recognizing the power of GNNs in learning from graph structures, the application of GNNs for user modeling has recently gained prominence, highlighting the significance of leveraging their capabilities in capturing complex relationships and patterns in user-item interactions [16–18, 21, 30, 32, 42, 45, 48]. Previous research has leveraged KGs and GNNs to predict user interests by approaching user modeling as a link prediction task [16, 21, 42, 47] or a node classification task [17, 18, 45] in the e-commerce domain. However, applying KGs and GNNs for user modeling in the educational domain is under-explored.

In this paper, we tackle the issue of lack of transparency, user control, and scrutability in learner modeling by harnessing the capabilities of PKGs. We leverage PKGs to capture learners' interactions with the MOOC platform CourseMapper [5]. This enables us to achieve scrutability by empowering learners to construct and control their models and enhance transparency by providing learners with a clearer understanding of how their learner models are constructed. Furthermore, we combine PKGs and GNNs to model the learner's knowledge state. Specifically, we utilize the power of GNNs to enhance representations of entities in the PKG by leveraging structure and semantic relations among PKG entities. Then, we model the learner knowledge state based on the concepts they do not understand (referred to as DNU concepts) in a learning material in CourseMapper.

2 RELATED WORK

2.1 Learner Knowledge State Modeling

Integrating learner models within online learning environments is crucial in understanding their behaviors and characteristics. Learner models encapsulate an abstract representation of learners' abilities and behaviors through the analysis of their actions and interactions within the learning system [19]. These learner models represent attributes of learners such as, domain knowledge, cognitive skills, interests, behaviors, as well as their meta-cognitive abilities and affective states, which typically are inferred based on their interactions with the system [15]. In an educational context, among the various attributes of a learner, their knowledge state emerges as one of the most pivotal elements of information within the learner model [14]. Current learner modeling adopted in online learning systems mostly relies on an 'overlaying learner model' approach, wherein the learner model estimates learners' knowledge state by scrutinizing their interactions and activities within the system [14]. To this end, researchers have used different methods, such as the Elo rating system [2, 28], the performance factor analysis (PFA) framework [38], the Bayesian knowledge tracing (BKT) model [37], or item response theory (IRT) [29]. In general, using these methods, learner models are generated implicitly based on learner's performance and interactions with the system. However, due to the complexity of these methods, how learner models are inferred is often not transparent to learners, as well as the learners are not involved in the process of generating learner models which leads to the lack of learners' trust and transparency with the system [3, 28]. To overcome this problem, our paper introduces a more transparent approach for modeling the learners' knowledge state by involving the learners in the process of generating the learner model, thus introducing more transparency in the learning system.

2.2 Transparent Learner Modeling

Addressing learners' individual needs and preferences remains a significant challenge in massive open online courses (MOOCs) and other online learning environments. Recent advancements in educational data mining and learning analytics are tackling this challenge by developing accurate and transparent learner models within these platforms. To achieve this, Open Learner Models (OLMs) come into play. While learner modeling is the process of collecting, organizing, and inferring the learner model information, OLMs represent a learner model that is externalized and accessible to either learners or instructors [15]. Integration of OLMs into educational systems aims to offer learners insights into their learner models and consequently enhance the transparency of these systems [2, 3, 10, 11]. While OLMs facilitate learners in understanding the model information utilized for personalization and identifying erroneous assumptions made by the system [39], they typically lack the capability for users to scrutinize their models closely, correct inaccuracies when they disagree with (parts of) it, or modify them to align with their preferences. Scrutable learner models allow learners to directly change the system's assessment and representation of their model at will [19]. Numerous researchers within the user modeling and recommender systems communities have highlighted the significance of enabling scrutability, emphasizing the need for users to offer explicit feedback on their generated user models [9, 12, 19, 22–24, 40]. Considering the need for more scrutable learner models, our paper extends beyond the mere provision of open learner models and introduces scrutable learner models by empowering learners to directly manipulate the system's representation of their models as desired [19]. In this way, we aim to enhance the accuracy of the learner model by allowing students to modify their learner model representations, which they may perceive as inaccurate, as well as introducing interactivity and controllability to the system to enhance user satisfaction.

Scrutable learner models are typically inferred based on students' interactions with the educational system [15]. Similarly, current user modeling approaches in practical applications (e.g., social media, e-commerce, information retrieval, recommender systems) emphasize implicit user modeling, which involves analyzing users' actions and interactions to create user models [1]. Considering that graphs provide an intuitive and efficient way to model and analyze these behaviors, graph-based methods, such as Knowledge Graphs (KGs) and Graph Neural Networks (GNNs), have recently emerged as a natural means enabling the development of an effective user model [32]. A KG is a type of graph that consists of nodes representing entities and edges that depict the relationships between these entities. Incorporating user information into the KG results in creating more accurate relations between users and items, as well as user preferences [49]. While scrutable learner models are often presented through visualizations [15, 24], in this paper, we took benefit from the KG's capability to capture structural and semantical information from learners' interactions with the learning system and leverage KGs for scrutable learning modeling.

2.3 Personal Knowledge Graphs

Nowadays, Knowledge Graphs (KGs) are frequently used for user modeling and recommendation generation. However, they lack user-specific information, which limits their effectiveness in delivering truly personalized content tailored to individual user preferences [36]. Personal Knowledge Graphs (PKGs) address this limitation by capturing entities from the KG only relevant to a specific user and representing this information in a structured way [8, 34, 36]. While PKGs have applications in the health domain [36], their use in education remains limited. Different from existing studies which adopted PKGs in education [20, 27], our work focuses on empowering learners to explicitly identify concepts they "Did Not Understand" (DNU concepts). These DNU concepts are incorporated into their PKG, which represents their learner model. In this way, we give control to the learners to generate their own scrutable learner model, thus leading to improved accuracy and transparency.

2.4 GNN-based Learner Modeling

In the domain of user modeling, KGs can capture explicit interactions between users and items. The KG's structural data can be used to describe user attributes, such as interests, preferences, age, and gender. Recently, Graph Neural Networks (GNNs) have emerged as a powerful tool that utilizes KG structure to learn and enhance the representation of various entities in the KG. GNNs excel at capturing complex user interactions by representing users as nodes in a graph, where edges depict the connections with different items based on user behavior. These connections allow GNNs to (1) pass messages between two connected nodes, and (2) learn and enhance the representations of nodes by aggregating information from multi-hop neighboring nodes [7, 25, 43, 44]. Several studies used KGs and GNNs to predict user interests and preferences based on their interactions [21, 41, 42, 47]. These studies approached user modeling primarily as a link prediction task. Other recent studies approached user modeling primarily as a node classification task using KGs and GNNs. These studies mainly aimed to categorize user attributes (such as gender and age) using textual or behavioral data in e-commerce domain [17, 18, 45, 46]. Existing studies primarily focus on GNN-based user modeling in the e-commerce domain, leveraging user interaction data (views, purchases). These approaches view user modeling as a link prediction task to predict user interests or as a node classification task to infer user demographics (gender, age). Our work is different from existing studies in that we focus on user modeling in the educational domain. In particular, we leverage GNNs to enrich the representation of the different KG items and model the learner's knowledge state as a weighted aggregation of the enhanced representation of their DNU concepts.

3 METHODOLOGY

In our approach, we combine PKGs, GCNs, and SBERT for transparent learner modeling in the MOOC platform CourseMapper. The learner modeling process includes an offline and an online phase.

3.1 Offline Phase

The offline phase includes two main steps: (1) PKG construction and (2) representation learning of PKG items using GCN (see Figure 1).

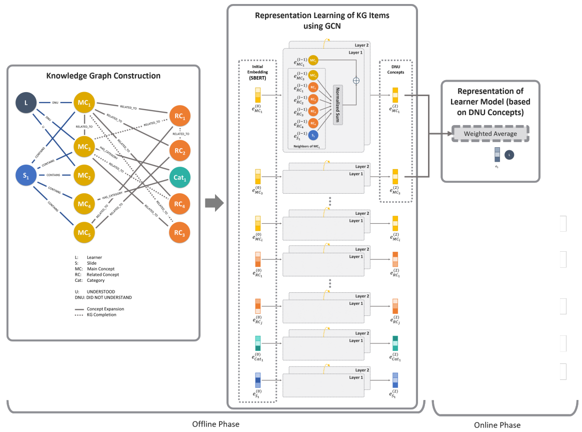

Fig. 1. The conceptual architecture of our proposed learner modeling approach

3.1.1 PKG Construction: A learner's PKG contains *Learner*, *Learning Material (LM)*, *Slide (S)*, *Main Concept (MC)*, *Related Concept (RC)*, and *Category (Cat)* as nodes, and *HAS_READ*, *CONTAINS*, *HAS_CATEGORY*, *RELATED_TO*, *CONSISTS_OF*, *UNDERSTOOD (U)*, and *DID_NOT_UNDERSTAND (DNU)* as relationships between the nodes. When a learner reads a slide in CourseMapper, a triplet *(Learner, HAS_READ, Slide)* is added to their PKG. Moreover, CourseMapper integrates a "Did Not Understand" (DNU) button at the bottom of each uploaded PDF learning material, typically lecture slides. Clicking this button reveals a slide-level KG showing the top five main concepts extracted from the slide content. Learners can interact by marking these concepts as 'Understood (U)', 'Did Not Understand (DNU)', or 'New'. This will create new relationships between the marked concepts and the learner node in the PKG. The KG generation for a new learning material in CourseMapper involves three steps: (1) KG construction, (2) concept expansion, and (3) KG completion.

KG Construction: Each uploaded learning material in CourseMapper comprises multiple slides. We construct an initial KG for each slide (Slide-level KG) through a four-step process: (1) Text extraction, (2) keyphrase extraction, (3) concept identification, and (4) concept filtering. The first step involves extracting text content from the uploaded PDF using PDFMiner [35]. Next, the SIFRank-SqueezeBERT algorithm [4] processes the text to identify the top-15 keyphrases representing the slide content. These keyphrases are annotated using the DBpedia Spotlight service [31] to identify words that correspond to entities or concepts in the DBpedia knowledge base, forming the main concepts (MCs) for the slide. After MC extraction, a filtering and ranking process refines the concepts for each slide-level KG. To achieve this, we utilize SBERT [33] to generate initial representations for each entity in the KG, capturing semantic relationships. By comparing cosine similarity between these embeddings, we filter and rank MCs based on their relevance to the slide and learning material. Combining semantic similarity scores yields an overall importance score for each MC. The top five MCs with the highest scores are retained for each slide, emphasizing crucial concepts. This process repeats for each slide in the learning material, forming slide-level KGs that are then integrated to create the KG for the entire learning material (LM-level KG).

Concept Expansion: The KG structure is enriched through concept expansion using DBpedia SPARQL queries to retrieve related concepts and categories associated with each MC [4]. However, a critical challenge emerges from the large volume of related concepts and categories within DBpedia. As we delve deeper into the knowledge base, these associated entities tend to become increasingly abstract and potentially irrelevant to the specific learning context. Additionally, incorporating all retrieved concepts without filtering would lead to an unwieldy and potentially unmanageable KG size. After acquiring related concepts and categories, SBERT is used to generate their embeddings. Related concept embeddings are computed based on the abstracts of their Wikipedia articles, while category embeddings are derived from their names. These embeddings are then used to calculate the cosine similarity between each MC and its related concepts/categories. This similarity score reflects their level of relatedness. Additionally, the cosine similarity between each related concept/category and the learning material is calculated. These two similarity scores are then summed up to create an overall score for each related concept/category. This score is used to rank the retrieved related concepts/categories based on their relevance. Finally, the top 20 most relevant related concepts and the top 3 most relevant categories for each MC are incorporated into the KG. This process enriches the KG structure by expanding and exploring related knowledge while maintaining a manageable size.

KG Completion: Enriching the KG with related concepts and categories can lead to missing connections between existing entities. To address this issue, we refined the KG by including bidirectional and transitive relationships between the entities (main concepts, related concepts, and categories). This graph completion step ensures that the KG accurately represents entity relationships, facilitating deeper understanding of the knowledge domain.

3.1.2 Representation Learning of PKG Items Using GCN:. Following PKG construction, we leverage the rich structural and semantic information within the PKG to enhance the representation of its various entities using GCNs. Motivated by LightGCN [26], we also simplified GCN by omitting feature transformation and nonlinear activation while aggregating multi-hop neighbors and updating the representations. The enhancement process undergoes three key stages: (1) Construct the initial embedding matrix, (2) construct the adjacency matrix, and (3) construct the final embedding matrix.

Construct Initial Embedding Matrix: To enhance the representation of PKG entities (slides, main concepts, related concepts, and categories), we first create an initial embedding matrix. This matrix is derived from the textual content of each entity using Sentence-BERT (SBERT). For slide nodes, the content of the slide itself is used to generate its embedding. Embeddings for main concepts and related concepts are derived from the abstracts of their corresponding Wikipedia articles. Category nodes utilize their category name for embedding generation.

Construct the Adjacency Matrix: Traditionally, adjacency matrices in graph networks represent connections between nodes with a simple binary approach. A value of 1 indicates a direct connection (adjacency) between two nodes, while a 0 signifies no direct connection. This approach captures the basic structure of the network but doesn't consider the strength or influence of these connections. Our approach goes beyond this basic structure. We incorporate relationship weights into the adjacency matrix. These weights are calculated based on the cosine similarity between the embeddings of connected nodes (items) in the PKG. Higher cosine similarity indicates stronger semantic relatedness and, consequently, a stronger influence between the connected nodes. Finally, to determine the significance of connections within the PKG, we propose a simple "attention mechanism" that combines symmetric square root normalization inspired by LightGCN [26] with SBERT-based semantic similarity [7]. This mechanism incorporates the structure and semantic information of the PKG to assign weights to connections, reflecting the importance of the relationship between two directly connected items u and v in the PKG, which is calculated as given below:

$$\omega_{u,v} = \frac{cos(e_u, e_v)}{\sqrt{|\mathcal{N}_u||\mathcal{N}_v|}} \tag{1}$$

where e stands for the embedding of an item, while $\mathcal{N}_u$ and $\mathcal{N}_v$ refer to the set of items directly linked to u and v respectively. The term cos represents the cosine similarity between two embedding vectors. We calculate the adjacency matrix as follows:

$$ADJm = \begin{cases} \omega_{u,v}, \; if \; v \; \in \; \mathcal{N}_u \\ 0, \; otherwise \end{cases} \tag{2}$$

Construct the Final Embedding Matrix: We apply GCN to enhance the representations of items in the PKG. We use the embeddings obtained at the last layer (i.e., layer 2 in our case) as the final embeddings. At layer $l+1$, the GCN uses an aggregation function to derive the updated item representation. This function combines weighted embeddings from both the item's direct neighbors and the item itself (considered as a self-connection) from layer l as given below:

$$e_u^{(l+1)} = e_u^{(l)} + \sum_{v \in \mathcal{N}_u} \omega_{u,v} \, e_v^{(l)} \tag{3}$$

The new embedding matrix at layer 1 is computed by multiplying the adjacency matrix $ADJm$ with the initial embedding matrix. We repeat this process at layer 2. Here, the adjacency matrix is multiplied with the newly generated embedding matrix from layer 1. The final product of this multiplication at layer 2 becomes the final embedding matrix containing the enhanced item embeddings which will be utilized in the online phase to model the learner knowledge state.

3.2 Online Phase

The online phase focuses on the enhanced representations of learner models based on their DNU concepts (see Figure 1). In CourseMapper, learners have the option to explore the associated MCs on the slide through the DNU button at the bottom of the slide. If they don't understand an MC, they can directly indicate this by marking it as *DNU*. This allows learners to actively inform the system about their knowledge state for different concepts. This ensures transparency and scrutability in the learner model by enabling the learners to control their learner model. The learner-defined DNU concepts form the foundation of their learner model. The system constructs a learner model, denoted as L, to represent a learner's knowledge state based on their DNU concepts. The learner model L is a vector where a concept marked as DNU (U) is represented as 1 (0). The learner model is represented as a weighted average of their DNU concepts. Each concept's weight is determined by the cosine similarity score between its embedding and the embedding of the learning material. More relevant concepts receive higher weights, emphasizing their importance for the learner's knowledge state. By leveraging the enhanced embeddings of the learner's DNU concepts, alongside their corresponding weights, the embedding for the learner model is computed as follows:

$$e_L = \left[\frac{1}{\omega_{sum}} \sum_{c \in DNU} \omega_c \, e_c \right]; \; \omega_c = cos(e_c, e_{lm}); \; \omega_{sum} = \sum_{c \in DNU} \omega_c \tag{4}$$

where e_c is the enhanced embedding of concept c and ω_c is its weight in the learning material lm.

4 CONCLUSION

In this work, we presented an approach for learner modeling in the MOOC platform CourseMapper, particularly focusing on modeling learners' knowledge states. Our approach effectively combines Personal Knowledge Graphs (PKGs), Graph Convolutional Networks (GCNs), and transformer sentence encoders (SBERT) to construct a controllable, scrutable, and transparent learner model. The enhanced representation of a learner model can be used in future to improve performance in downstream tasks, such as adaptive systems, recommendation, and personalized search.

REFERENCES

[1] Mohamed Abdelrazek, Erasmo Purificato, Ludovico Boratto, and Ernesto William De Luca. 2023. Fairup: A framework for fairness analysis of graph neural network-based user profiling models. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3165–3169.

[2] Solmaz Abdi, Hassan Khosravi, Shazia Sadiq, and Dragan Gasevic. 2019. A multivariate Elo-based learner model for adaptive educational systems. In *Proceedings of the Educational Data Mining Conference*. 462–467.

[3] Solmaz Abdi, Hassan Khosravi, Shazia Sadiq, and Dragan Gasevic. 2020. Complementing educational recommender systems with open learner models. In *Proceedings of the tenth international conference on learning analytics & knowledge*. 360–365.

[4] Qurat Ul Ain, Mohamed Amine Chatti, Komlan Gluck Charles Bakar, Shoeb Joarder, and Rawaa Alatrash. 2023. Automatic Construction of Educational Knowledge Graphs: A Word Embedding-Based Approach. *Information* 14, 10 (2023), 526.

[5] Qurat Ul Ain, Mohamed Amine Chatti, Shoeb Joarder, Ilia Nassif, Benjamine Stella Wobiwo Teda, Mouadh Guesmi, and Rawaa Alatrash. 2022. Learning Channels to Support Interaction and Collaboration in CourseMapper. In *Proceedings of the 14th International Conference on Education Technology and Computers*. 252–260.

[6] Qurat Ul Ain, Mohamed Amine Chatti, Paul Arthur Meteng Kamdem, Rawaa Alatrash, Shoeb Joarder, and Clara Siepmann. 2024. Learner Modeling and Recommendation of Learning Resources using Personal Knowledge Graphs. In *Proceedings of the 14th Learning Analytics and Knowledge Conference*. 273–283.

[7] Rawaa Alatrash, Mohamed Amine Chatti, Qurat Ul Ain, Yipeng Fang, Shoeb Joarder, and Clara Siepmann. 2024. ConceptGCN: Knowledge concept recommendation in MOOCs based on knowledge graph convolutional networks and SBERT. *Computers and Education: Artificial Intelligence* 6 (2024), 100193.

[8] Krisztian Balog and Tom Kenter. 2019. Personal knowledge graphs: A research agenda. In *Proceedings of the 2019 ACM SIGIR International Conference on Theory of Information Retrieval*. 217–220.

[9] Krisztian Balog, Filip Radlinski, and Shushan Arakelyan. 2019. Transparent, scrutable and explainable user models for personalized recommendation. In *Proceedings of the 42nd international acm sigir conference on research and development in information retrieval*. 265–274.

[10] Jordan Barria and Peter Brusilovsky. 2019. Explaining educational recommendations through a concept-level knowledge visualization. In *Proceedings of the 24th International Conference on Intelligent User Interfaces: Companion*. 103–104.

[11] Jordan Barria and Peter Brusilovsky. 2019. Making educational recommendations transparent through a fine-grained open learner model. In *Proceedings of Workshop on Intelligent User Interfaces for Algorithmic Transparency in Emerging Technologies, Los Angeles, USA, March 20, 2019*.

[12] Satabdi Basu, Gautam Biswas, and John S Kinnebrew. 2017. Learner modeling for adaptive scaffolding in a computational thinking-based science learning environment. *User Modeling and User-Adapted Interaction* 27 (2017), 5–53.

[13] Robert Bodily, Judy Kay, Vincent Aleven, Ioana Jivet, Dan Davis, Franceska Xhakaj, and Katrien Verbert. 2018. Open learner models and learning analytics dashboards: a systematic review. In *Proceedings of the 8th international conference on learning analytics and knowledge*. 41–50.

[14] Peter Brusilovsky and Eva Millán. 2007. User models for adaptive hypermedia and adaptive educational systems. In *The adaptive web: methods and strategies of web personalization*. Springer, 3–53.

[15] Susan Bull and Judy Kay. 2016. SMILI: A framework for interfaces to learning data in open learner models, learning analytics and related fields. *International Journal of Artificial Intelligence in Education* 26 (2016), 293–331.

[16] Rui Chen and Qinglei Meng. 2023. A Knowledge Graph Preference Propagation Recommendation Algorithm Based on Introducing Relational Information. In *2023 IEEE 5th International Conference on Civil Aviation Safety and Information Technology (ICCASIT)*. IEEE, 1480–1484.

[17] Weijian Chen, Fuli Feng, Qifan Wang, Xiangnan He, Chonggang Song, Guohui Ling, and Yongdong Zhang. 2021. Catgcn: Graph convolutional networks with categorical node features. *IEEE Transactions on Knowledge and Data Engineering* 35, 4 (2021), 3500–3511.

[18] Weijian Chen, Yulong Gu, Zhaochun Ren, Xiangnan He, Hongtao Xie, Tong Guo, Dawei Yin, and Yongdong Zhang. 2019. Semi-supervised User Profiling with Heterogeneous Graph Attention Networks.. In *IJCAI*, Vol. 19. 2116–2122.

[19] Cristina Conati, Kaska Porayska-Pomsta, and Manolis Mavrikis. 2018. AI in Education needs interpretable machine learning: Lessons from Open Learner Modelling. *arXiv preprint arXiv:1807.00154* (2018).

[20] Yuli Deng, Duo Lu, Dijiang Huang, Chun-Jen Chung, and Fanjie Lin. 2019. Knowledge graph based learning guidance for cybersecurity hands-on labs. In *Proceedings of the ACM conference on global computing education*. 194–200.

[21] Lu Fan, Qimai Li, Bo Liu, Xiao-Ming Wu, Xiaotong Zhang, Fuyu Lv, Guli Lin, Sen Li, Taiwei Jin, and Keping Yang. 2022. Modeling user behavior with graph convolution for personalized product search. In *Proceedings of the ACM Web Conference 2022*. 203–212.

[22] D Graus, M Sappelli, and D Manh Chu. 2018. " let me tell you who you are"-Explaining recommender systems by opening black box user profiles. In *Proceedings of the 2nd Fatrec Workshop on Responsible Recommendation*.

[23] Mouadh Guesmi, Mohamed Amine Chatti, Yiqi Sun, Fangzheng Ji, Arham Muslim, Laura Vorgerd, and Shoeb Ahmed Joarder. 2021. Open, Scrutable and Explainable Interest Models for Transparent Recommendation.. In *Proceedings of the Joint Proceedings of the ACM IUI 2021 Workshops*.

[24] Mouadh Guesmi, Mohamed Amine Chatti, Alptug Tayyar, Qurat Ul Ain, and Shoeb Joarder. 2022. Interactive visualizations of transparent user models for self-actualization: A human-centered design approach. *Multimodal Technologies and Interaction* 6, 6 (2022), 42.

[25] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*.

[26] Xiangnan He, Kuan Deng, Xiang Wang, Yan Li, Yongdong Zhang, and Meng Wang. 2020. Lightgcn: Simplifying and powering graph convolution network for recommendation. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*. 639–648.

[27] Eleni Ilkou. 2022. Personal Knowledge Graphs: Use Cases in e-Learning Platforms. In *Companion Proceedings of the Web Conference 2022* (Virtual Event, Lyon, France) *(WWW '22)*. Association for Computing Machinery, New York, NY, USA, 344–348. https://doi.org/10.1145/3487553.3524196

[28] Hassan Khosravi, Kirsty Kitto, and Joseph Jay Williams. 2019. Ripple: A crowdsourced adaptive platform for recommendation of learning activities. *Journal of learning analytics* 6, 3 (2019), 91–105.

[29] Walter L. Leite, Samrat Roy, Nilanjana Chakraborty, George Michailidis, A Corinne Huggins-Manley, Sidney D'Mello, Mohamad Kazem Shirani Faradonbeh, Emily Jensen, Huan Kuang, and Zeyuan Jing. 2022. A novel video recommendation system for algebra: An effectiveness evaluation study. In *LAK22: 12th International Learning Analytics and Knowledge Conference*. 294–303.

[30] Danyang Liu, Yuji Yang, Mengdi Zhang, Wei Wu, Xing Xie, and Guangzhong Sun. 2022. Knowledge enhanced multi-interest network for the generation of recommendation candidates. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. 3322–3331.

[31] Pablo N. Mendes, Max Jakob, Andres Garcia-Silva, and Christian Bizer. 2011. DBpedia spotlight: shedding light on the web of documents. In *International Conference on Semantic Systems*.

[32] Erasmo Purificato, Ludovico Boratto, and Ernesto William De Luca. 2023. Tutorial on user profiling with graph neural networks and related beyond-accuracy perspectives. In *Proceedings of the 31st ACM Conference on User Modeling, Adaptation and Personalization*. 309–312.

[33] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* (2019).

[34] Markus Schröder, Christian Jilek, and Andreas Dengel. 2022. A Human-in-the-Loop Approach for Personal Knowledge Graph Construction from File Names. In *Proceedings of the 3rd International Workshop on Knowledge Graph Construction*.

[35] Yusuke Shinyama. 2013. PdfMiner. Retrieved May 7, 2023 from https://unixuser.org/~euske/python/pdfminer/

[36] Martin G Skjæveland, Krisztian Balog, Nolwenn Bernard, Weronika Lajewska, and Trond Linjordet. 2023. An Ecosystem for Personal Knowledge Graphs: A Survey and Research Roadmap. *arXiv preprint arXiv:2304.09572* (2023).

[37] Kyosuke Takami, Yiling Dai, Brendan Flanagan, and Hiroaki Ogata. 2022. Educational explainable recommender usage and its effectiveness in high school summer vacation assignment. In *LAK22: 12th International Learning Analytics and Knowledge Conference*. 458–464.

[38] Khushboo Thaker, Lei Zhang, Daqing He, and Peter Brusilovsky. 2020. Recommending Remedial Readings Using Student Knowledge State. *Proceedings of The 13th International Conference on Educational Data Mining (EDM 2020)* (2020).

[39] Nava Tintarev and Judith Masthoff. 2010. Designing and evaluating explanations for recommender systems. In *Recommender systems handbook*. Springer, 479–510.

[40] Nava Tintarev and Judith Masthoff. 2015. Explaining recommendations: Design and evaluation. In *Recommender systems handbook*. Springer.

[41] Hongwei Wang, Fuzheng Zhang, Jialin Wang, Miao Zhao, Wenjie Li, Xing Xie, and Minyi Guo. 2018. Ripplenet: Propagating user preferences on the knowledge graph for recommender systems. In *Proceedings of the 27th ACM international conference on information and knowledge management*. 417–426.

[42] Ziyang Wang, Wei Wei, Gao Cong, Xiao-Li Li, Xian-Ling Mao, and Minghui Qiu. 2020. Global context enhanced graph neural networks for session-based recommendation. In *Proceedings of the 43rd international ACM SIGIR conference on research and development in information retrieval*. 169–178.

[43] Shiwen Wu, Fei Sun, Wentao Zhang, Xu Xie, and Bin Cui. 2022. Graph neural networks in recommender systems: a survey. *Comput. Surveys* 55, 5 (2022), 1–37.

[44] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and S Yu Philip. 2020. A comprehensive survey on graph neural networks. *IEEE transactions on neural networks and learning systems* 32, 1 (2020), 4–24.

[45] Qilong Yan, Yufeng Zhang, Qiang Liu, Shu Wu, and Liang Wang. 2021. Relation-aware heterogeneous graph for user profiling. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 3573–3577.

[46] Shaojie Yan, Tao Zhao, and Jinsheng Deng. 2022. Interaction-aware Hypergraph Neural Networks for User Profiling. In *2022 IEEE 9th International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 1–10.

[47] Fan Yang, Yong Yue, Gangmin Li, Terry R Payne, and Ka Lok Man. 2023. KEMIM: Knowledge-enhanced User Multi-interest Modeling for Recommender Systems. *IEEE Access* (2023).

[48] Liangwei Yang, Shengjie Wang, Yunzhe Tao, Jiankai Sun, Xiaolong Liu, Philip S Yu, and Taiqing Wang. 2023. Dgrec: Graph neural network for recommendation with diversified embedding generation. In *Proceedings of the sixteenth ACM international conference on web search and data mining*. 661–669.

[49] Y Zhang, Q Ai, X Chen, and P Wang. [n. d.]. Learning over knowledge-base embeddings for recommendation. arXiv 2018. *arXiv preprint arXiv:1803.06540* ([n. d.]).