
Joint Representation of Entities and Relations via Graph Attention Networks for Explainable Recommendations

Rima Boughareb^{1,*}, Hassina Seridi-Bouchelaghem¹
and Samia Beldjoudi²

¹*Department of Computer Science, Badji Mokhtar – Annaba University, Annaba, Algeria, LabGED Laboratory, Po box 12, Annaba, Algeria*

²*National Higher School of Technology and Engineering, LTSE (Laboratoire de Technologies des Systèmes Energétiques), Annaba, Algeria*

E-mail: rima.boughareb@univ-annaba.org; seridi@labged.net; s.beldjoudi@esti-annaba.dz

**Corresponding Author*

Received 07 April 2023; Accepted 15 June 2023;
Publication 24 October 2023

Abstract

The latest advances in Graph Neural Networks (GNNs), have provided important new ideas for solving the Knowledge Graph (KG) representation problem for recommendation purposes. Although GNNs have an effective graph representation capability, the nonlinear transformations over the layers cause a loss of semantic information and make the generated embeddings hard to explain. In this paper, we investigate the potential of large KGs to perform interpretable recommendation using Graph Attention Networks (GATs). Our goal is to fully exploit the semantic information and preserve inherent knowledge ported in relations by jointly learning low-dimensional embeddings for nodes (i.e., entities) and edges (i.e., properties). Specifically, we feed the original data with additional knowledge from the Linked Open Data (LOD) cloud, and apply GATs to generate a vector representation for each node on the graph. Experiments conducted on three real-world

Journal of Web Engineering, Vol. 22.4, 615–638.

doi: 10.13052/jwe1540-9589.2243

© 2023 River Publishers

datasets for the top-K recommendation task demonstrate the state-of-the-art performance of the system proposed. In addition to improving predictive performance in terms of precision, recall, and diversity, our approach fully exploits the rich structured information provided by KGs to offer explanation for recommendations.

Keywords: Recommender systems, knowledge graphs, graph attention networks, graph embedding, machine learning, graph representation learning.

1 Introduction

A Knowledge Graph is a type of graph-based data model that represents data as a set of nodes and edges, with nodes representing entities or concepts, and edges representing the relationships between them. KGs offer a structured approach to organizing and linking information efficiently, facilitating data integration [Cudré-Mauroux,2020; Paneque, et al., 2023; Diamantini, et al., 2022; Asprino, et al., 2023], analysis [Ryen et al., 2022; Wu et al., 2021], discovery [Zeng et al., 2022], and supporting reasoning and inference [Chen et al., 2019; Lin et al., 2019]. This versatility makes KGs beneficial for numerous applications, such as natural language processing [Schneider et al., 2022, Zhu et al., 2023], semantic search [Bettina and Thomas, 2010], knowledge management [De Donato et al., 2020; Jiang et al., 2012], and recommendations [Boughareb et al., 2020; Boughareb et al., 2023; Oramas et al., 2017; Zhao et al., 2023, Wang et al., 2022; Arabi et al., 2020]. A Recommender System (RS) is a type of algorithmic system that provides personalized recommendations to users based on their past preferences. The integration of KGs into RSs can help provide more relevant and context-aware recommendations to users. Indeed, KGs are used to enrich the data with additional semantic information, such as concept hierarchies, relationships, and attributes. This can help to capture more nuances and subtleties in the data, which can lead to more accurate and personalized recommendations [Guo et al., 2022].

The success of recent advances in Deep Graph Representation Learning methods (DGLR) in various domains such as computer vision [Bouguettaya et al., 2021; Haghighat and Sharma, 2023; Ramezani et al., 2023] and social networks [Deng et al., 2017; Wu et al., 2020, Islam et al., 2020; Tesfagerish et al., 2022] has shed light on their potential for use in KG-aware RSs [Boughareb et al., 2023, Zhang et al., 2019; Liu et al., 2021, Pham et al., 2023]. DGLR methods such as Graph Convolutional Networks (GCNs) [Kipf

and Welling, 2017] and Graph Attention Networks (GATs) [Veličković et al., 2018], can learn more expressive and powerful representations of the entities and their relationships in a KG. This is particularly useful in knowledge-aware systems, where the relationships between items and users are often more nuanced than in traditional recommender systems. By encoding the graph structure into low-dimensional vector representations, these methods can capture rich and complex information about the entities and their relations, leading to more accurate recommendations. GAT is one of the most commonly applied GNN architectures for recommendation, it consists of learning a set of attention coefficients for each node, which determine the importance of its neighbor in the representation learning process. For example, the system KGAT [Wang et al., 2019] is the first to expand GATs to knowledge graphs for the purpose of recommendations. By utilizing a self-attention mechanism, KGAT produces explainable recommendations by taking into account high-order relationships. [Wang et al., 2020] proposed the GAATs model, which utilizes an attenuated attention mechanism to generate new embeddings for both entities and relations. The attenuated attention mechanism permits assigning varying weights to different relation paths, which helps in gathering information from the neighborhoods. Recently, [Jin et al., 2023], proposed a Meta-path guided graph attention network to provide the explainable medical herb recommendations.

Although DGRL techniques particularly GATs have made significant progress in the recommendation research field, there are still some outstanding issues that need to be addressed. First, DGLR methods often lack interpretability, making it difficult to understand how the embeddings are learned and how they contribute to the recommendation process. Specifically, the nonlinear transformations over the layers cause a loss of semantic information and make the generated embeddings hard to explain. Furthermore, GATs suffer the problems of high complexity and computational cost. Regarding KG-aware RS, the state-of-the-art models only treat the property embedding as a secondary aspect of the entity embedding. They do not take a deeper look into embedding properties, ultimately causing a loss of semantic information carried by these properties, especially when used in conjunction with RSs to process large amounts of semantic linked data. In this paper, a novel entity and property knowledge-based graph attention network is proposed to achieve better performance on recommendation. A knowledge representation that emphasizes semantic preservation during graph attention network learning is built, which includes four types of node: user nodes, item nodes, property nodes, and semantic entity nodes. This is preceded

by a feature selection phase to reduce the complexity of the GATs learning process. The experimental results on three real-world recommendation scenarios demonstrate that the proposed approach improves state-of-the-art performance, provides effective explanations, and ensures diversity of recommendations.

To summarize, the key contributions of this research are:

- (i) The proposal of a novel knowledge representation that emphasizes semantic preservation by leveraging rich background knowledge from the LOD cloud and applying it to real-world user-item datasets to perform top-N recommendation tasks.
- (ii) The development of a new algorithm called property-to-entity, which transforms property edges into entity nodes to enhance the representation of properties in the graph, which are often overlooked in traditional knowledge graph-based recommendation approaches.
- (iii) A semantic-based information gain property selection is exploited to reduce prior GATs algorithm complexity used for the learning process.
- (iv) The proposal of a semantic-based information gain property selection method to reduce the complexity of the GAT learning process. The property selection enhances the efficiency of the learning process and enables the model to capture more relevant semantic relationships between entities and properties, resulting in better performance and more diverse recommendations.
- (v) The experimental results conducted on real-world datasets for the top-K recommendation task demonstrate that the proposed approach achieves state-of-the-art performance in terms of accuracy, diversity, and explanation. These results validate the effectiveness of the proposed algorithm and highlight its potential for practical applications in real-world recommendation scenarios.

The structure of the paper is as follows: Section 2 provides a review of related works; Section 3 presents the proposed approach in detail; Section 4 outlines the research methodology, and presents and analyzes the experimental results; finally, Section 5 concludes the paper and suggests potential avenues for future research.

2 Related Work

GNNs have introduced innovative solutions for tackling the problem of graph representation learning in various domains, including image classification

[Dong et al., 2017; Marino et al., 2017], link prediction [Lei et al., 2019; Mudiyansele et al., 2022], and recommendation tasks [Boughareb et al., 2023; Zhang et al., 2019; Liu et al., 2021, Pham et al., 2023]. Among the GNN architectures, GCNs and GATs are the most commonly employed in recommendation systems. While GCN aggregates the feature vectors of a node's neighborhood with equal weights, this approach may not always be appropriate since some nodes carry more significance than others. GAT addresses this issue by employing a learnable function of weights to determine each node's importance, allowing for more nuanced and personalized recommendations. KGAT [Wang et al., 2019] is the first system to extend GATs to knowledge graphs for recommendation purposes. Based on a self-attention mechanism, KGAT provides explainable recommendations considering high-order relationships. In the KGAT model, the explanations can be obtained directly from the model by interpreting the attention weights. [Wang et al., 2020] proposed the GAATs model, which applies an attenuated attention mechanism to obtain new embeddings on both entities and relations. The attenuated attention mechanism allows assigning different weights in different relation paths and acquires the information from the neighborhoods. In [Shimizu et al., 2022], a new approach to explainable recommendation is presented, leveraging an advanced knowledge graph attention network model that takes into account item-specific side information to deliver highly accurate recommendations. The proposed framework enables direct interpretation of the reasoning behind each recommendation by visualizing the factors that contributed to it. [Dai et al., 2022] proposed a novel framework with collaborative and attentive graph convolutional networks for personalized knowledge-aware recommendation. Particularly, we model the user-item graph and the KG separately and simultaneously with an efficient graph convolutional network and a personalized knowledge graph attention network, where the former aims to extract informative collaborative signals, while the latter is designed to capture fine-grained semantics. [Tu et al., 2021] proposed knowledge-aware conditional attention networks, which is an end-to-end model to incorporate a knowledge graph into an RS. The authors use a knowledge-aware attention propagation manner to acquire the node representation, then knowledge-aware attentions are used to extract the knowledge graph into the target-specific subgraph. Recently, [Jin et al., 2023] introduced a meta-path guided graph attention network to provide the explainable medical herb recommendations.

The current GAT-based knowledge-aware recommendation models often lack interpretability, making it difficult to understand how the embeddings are learned and how they contribute to the recommendation process. These models often focus solely on feeding the original data with additional knowledge, while the semantic aspect is usually ignored. Specifically, they do not take a deeper look into embedding properties, leading to a loss of semantic information carried by these properties, especially when used in conjunction with RSs to process large amounts of semantic linked data.

In contrast, the proposed system is based on a knowledge representation of user-item and semantic metadata that emphasizes the preservation of semantic information. The proposed approach gives equal importance to the modeling of entities and properties, in contrast to current state-of-the-art GAT-based knowledge-aware recommendation models. This enables the model to better capture the underlying semantic relationships and improve the interpretability of the learned embeddings.

3 Proposed Approach

The system consists of four primary steps. The first involves constructing the KG by integrating rich semantic information. In the second step, a subset of properties is automatically selected using the information gain technique to perform property selection. The third step involves generating KG embeddings with GATs, which create relation-specific embeddings for each entity based on the semantic relations present in the KG. Finally, the fourth step entails providing an explainable ranked list of top-N items that each user is likely to be interested in, called the top-N explained item recommendation. Figure 1 illustrates the architecture of the proposed system, and the following subsections detail each step.

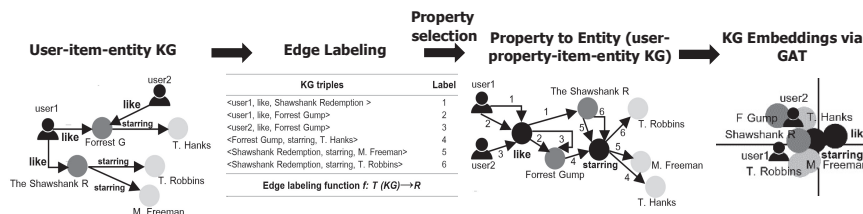


Figure 1 The proposed system architecture.

3.1 Building a Graph-based Knowledge Representation

In this study, three recommendation scenarios were chosen, namely movies, books, and music. To enrich the user-item data with semantic information, a three-step process was employed. The first step involved extracting user-item data from the dataset, which included information about users and item ratings. Next, the data was linked to external entities in the linked open data (LOD) cloud, specifically a cross-domain semantic dataset like DBpedia, and eventually a specialized semantic dataset (e.g., LinkedMDB for the movie domain). Once the data was linked, additional metadata were added to enhance the data. The utilization of the resources available in the LOD cloud improved the accuracy and completeness of the recommendation domain. It should be noted that, for readability purposes, the movie recommendation domain was used as an example in this section.

To create the knowledge graph, we interlinked the MovieLens dataset with two linked data sources, DBpedia and LinkedMDB. Each movie in MovieLens is matched to its corresponding DBpedia or LinkedMDB URI via a SPARQL query. For example, to obtain the mapping of the movie *Inception*, we submit a selection SPARQL query that returns as output the resource <http://dbpedia.org/resource/inception> whose name matches the title of the target movie. Then, we enriched the dataset with structural knowledge related to the movie domain. Initially, we extracted all the existing properties. As a result, the mapping contains 3862 DBpedia URIs and 50 properties. The data from which recommendations can be produced is typically derived from interactions between users $u \in U$ and items $i \in I$ with a rating $r_{ui} \in R$. We defined the collaborative property *like* (*user*, *item*) to model the fact that the user u likes the item i (e.g., $\langle user1, like, Avatar \rangle$). The property *like* describes on the explicit positive ratings, ranging from 1 to 5. Therefore, a knowledge graph $KG = (V, T)$ is created where (i) V denotes the set of nodes representing: users $U = \{u_1, \dots, u_n\}$, items $I = \{i_1, \dots, i_m\}$ and different entities $E = \{e_1, \dots, e_k\}$ like actors, directors, and producers; (ii) T denotes the set of labeled edges represented as RDF triples $\langle item, property, entity \rangle$.

3.2 Property Selection

GATs can be computationally expensive, especially when used in conjunction with RSs to process large amounts of semantic linked data. Therefore,

it is essential to identify the most useful properties that can enhance the effectiveness of our recommendation strategy and optimize prediction performance. To achieve this, we have applied a property selection technique that aims to reduce the number of properties in the dataset. Property selection is a data analysis mechanism that eliminates irrelevant properties from high-dimensional data, thus improving the performance of machine learning models. By selecting only the relevant properties, we can reduce computational costs and improve the accuracy of our predictions.

Information gain is an entropy-based feature selection method, widely used in the machine learning field. It works by measuring the reduction in entropy or disorder in the target variable by splitting data based on the value of a feature. Specifically, the method computes the entropy of the target variable before and after the split and then calculates the difference between the two entropies. The larger the difference, the more informative the feature is considered to be. In this way, information gain identifies the most relevant features for modeling and prediction tasks, thereby reducing the dimensionality of the input data and improving model performance.

More formally, the definition of information gain can be given as:

$$IG(X, Y) = Entropy(X) - Entropy(X|Y) \quad (1)$$

where $IG(X, Y)$ is the information gain obtained by knowing variable Y about variable X , $Entropy(X)$ is the entropy of variable X , and $H(X|Y)$ is the conditional entropy of variable X given variable Y . The entropy of a random variable X is a measure of the uncertainty in its probability distribution. It is defined as the sum of the negative logarithms of the probabilities of its possible outcomes, weighted by their probabilities. The entropy of X is defined as:

$$Entropy(X) = - \sum p(X) \log_2 p(X) \quad (2)$$

where $p(X)$ is the probability of the outcome X , and the sum is taken over all possible outcomes of X . The entropy is measured in bits, and it is maximized when all outcomes are equally likely, and minimized when one outcome is certain to occur with probability 1.

In the context of semantic property selection using information gain, we first need to define a target variable that we want to predict based on the semantic properties. Once we have a target variable, we can calculate the information gain of each semantic property with respect to that variable. We can then select the semantic properties with the highest information gain

and use them as features in our machine learning model. Indeed, information gain measure tells us how much information a property provides about the target variable.

We define the information gain of a property P_i as:

$$Gain(P_i) = Entropy(I) - \sum_{v \in P_i} \frac{|I_v|}{|I|} * Entropy(I_v) \quad (3)$$

$$Entropy(P_i) = - \sum_i P(i) \cdot \log_2 P(i) \quad (4)$$

where, $Entropy(I_v)$ represents the value of the entropy of the data, it is formally defined in Equation (4), where $P(i)$ is the probability of getting the i th value when selecting one from the set, I_v represents how many items have the property P_i with value v , and $Entropy(I_v)$ is the entropy calculated from data where property P_i assumes value v . Features are ranked in descending order according to their gain and the k highest-ranked ones are selected.

3.3 Property to Entity Algorithm

Most attempts to use GNN methods to leverage KGs for recommendation have focused on populating the data with the semantic content encoded in KGs and then learning entity embeddings. However, the importance of property in a knowledge graph cannot be overstated, as it is the primary element that determines the quality of knowledge reasoning. Indeed, only learning entity embeddings obstructs reaping all the benefits of KGs and makes the generated embeddings hard to explain.

Therefore, we propose a property to entity algorithm which transforms properties into entities to fully use network knowledge of both entities and properties. The algorithm transforms relations represented as labeled edges into nodes. We note that some properties can exhibit complex connectivity patterns, and applying such mapping will change the graph structure. For that, we used an edge labeling function $f: T(KG) \rightarrow R$ to assign a label for each RDF triple $\langle s, p, o \rangle$ in the graph. The f function uses consecutive integers $\{1, \dots, |N|\}$ for labeling triples with $|N|$ as the total number of triples. In the next step, we transform each property to a pair of edges with a new node n_p , where n_p will label the original edge-label, and the triple T becomes a pair of edges: $\langle s, n_p \rangle$ and $\langle n_p, o \rangle$. The pseudocode of the property to entity algorithm is given in the following code block.

Input: $KG(V, T)$: N triples T , M property $p \subset T$
Output: $KG'(V', E)$

1. for each triple $T < s, p, o > 1 \dots N$
 $f: T(KG) \rightarrow R$
2. for each property $p(1 \dots M)$
create node n_p
for $i: 1 \dots N$
if $p = n_p$
create edge $< s, n_p >$ with label i
create edge $< n_p, o >$ with label i

3.4 Attentive Embedding Layer

Graph attention networks are a type of neural network architecture designed to operate on graph-structured data. GAT incorporates attention mechanisms, which allows for the adaptive combination of features from neighboring nodes. The core idea behind GAT is to compute the attention coefficients between a node and its neighbors, which determine the importance of each neighbor node's features for the central node. The attention coefficients are computed by applying a neural network to a concatenation of the central node's feature vector and its neighbors' feature vectors. The resulting attention coefficients are then used to compute a weighted sum of the neighbor nodes' feature vectors, which is combined with the central node's feature vector to obtain a refined representation.

We ran GAT on the graph KG to represent both nodes and edges as vectors in the same space R_d . The aim is to generate for each input node $e_i = \{e_1, e_2, \dots, e_N\}$, an embedding $\vec{z}_{e_i} = \{\vec{z}_{e_1}, \vec{z}_{e_2}, \dots, \vec{z}_{e_N}\}$, where e_i represents the input embedding of the i^{th} node. \vec{z}_{e_i} represents the output embedding of the i^{th} node. N represents the total number of nodes.

The attention score α of a node e_i is given in Equation (5):

$$\alpha(e_i, e_j) = \text{softmax}(\sigma(a^T \cdot [W e_i || W e_j])) \quad (5)$$

where W represents a learnable linear projection matrix, σ represents a *LeakyReLU* activation function; it gives the final functional form for calculating raw attention scores which we can then pass into a *softmax* function to get the final normalized attention scores α . A is a learnable parameter that represents the direction that should be attended to. Indeed, the equation

applies a linear projection to each of the embedding separately, concatenates the results, and takes the dot product with a .

The final form of the GAT update equation can be defined as:

$$h_{e_i} = \sigma \left(\sum_{v \in e_i} \alpha(e_i, e_j) W e_j \right). \quad (6)$$

3.5 Item Recommendation

The item recommendation problem provides a ranked list of items that user u is likely to be interested in. Relying to the final outputs of the GAT model, each item i is represented as a global vector \vec{z}_i' . The relatedness between each candidate item i' and the items previously liked by u is computed using the score function of the translational method TransE [Bordes et al., 2013].

TransE is an embedding method for knowledge graphs, it learns representations of entities and relations so that $s + p \approx o$ where $\langle s, p, o \rangle$ is an RDF triple. TransE uses the following score function: $f(s, p, o) = d(s + p, o)$, where d is a distance function. For a triple $\langle s, p, o \rangle$, if the score $f(s, p, o)$ is closer to zero, the triple is considered as true. In our approach, we assign a score for each triple $\langle u, like, i \rangle$, where $u \in U$ is a user, $I \in I$ is an item. As an example, for the triple $\langle user1, like, Avatar \rangle$, the TransE score function uses the vectors for the three elements in the triple as follows:

$$f(user1, like, Avatar) = d(user1 + like - Avatar)$$

If $f(user1, like, Avatar) \approx 0$ then the film Titanic is not relevant for the user1 and will not be recommended for them, else i.e., $f(user1, like, Avatar) \approx 1$, *Avatar* is recommended for *user1*.

3.6 A Recommendation Regarding a Protocol for Diversification

Recommending items similar to the ones the user preferred before can achieve higher precision. However, users tend to be more satisfied with varying recommendations, i.e. being exposed to a content, which can promote discovery of something unexpected.

To address the diversity problem, we proceed to use property embeddings in order to understand the user tastes and determine how he chooses a movie to watch. By measuring user-property relatedness, we can determine which property (i.e., movie characteristics like actors, directors, and movie genres)

influences the user's choice. For example, one can select a movie according to its genre, another select a movie according to their favorite actors, also, some users may prefer movies directed by a specific director or produced by a particular producer.

Hence, considering these points allows controlling the diversification protocol. First, we define the user-property relatedness score as the cosine similarity between their vector representations. Then we rank the results in ascending order. Hence, for each user, we have l order preferences from lower to higher. Next, we select k items from the recommended-item list (RI-list). We process now the list from position $k + 1$. For each movie m in the RI-list, we measure its relatedness with each property p . According to this distance we add m to the corresponding property order.

4 Experimental Evaluation

In this section, we assess our proposed model in four real-world scenarios for movie, book, and music recommendations. Our objective is to address the following research questions:

- **RQ1.** Can selecting specific properties result in an increase in recommendation accuracy?
- **RQ2.** How do the proposed approach and baselines perform?
- **RQ3.** Can knowledge graphs be relied on to achieve diverse recommendations?
- **RQ4.** How effective is the proposed approach in providing reasonable explanations for user preferences towards items?

4.1 Experimental Settings

Datasets. Three benchmark datasets have been selected as the experimental datasets, and two linked datasets have been chosen to provide metadata to the system, (1) *MovieLens*, which is a movie dataset, it contains 943 users, 1682 movies, and 100,000 explicit ratings on a 5-star scale. Each user has rated at least 20 movies; (2) *LibraryThing*, which is related to the book domain and contains 7279 users, 37,231 books, and over 700,000 ratings; (3) *Last.fm*, which is a popular music dataset that contains approximately 1.1 billion play counts from 2.1 million users. The dataset includes metadata on artists and tracks, as well as user profiles and listening histories; (4) *DBpedia* (DBP in short), which is a cross-domain dataset in the linked data cloud; and

(5) *LinkedMDB* (LMDb in short), which is an open semantic database for movies.

Baselines. The recommendation results were compared to the following systems, respectively:

- **Item-based K-nearest neighbors (I-KNN)** is widely used for collaborative filtering. Its approach involves identifying K-items that are most similar to a given item to predict its rating for a user. By analyzing the ratings of those similar items, the algorithm infers the rating of the given item for that user. The similarity between items is determined using a metric like cosine similarity, which helps find the most similar elements.
- **Content-based filtering (CB).** Its fundamental concept is to suggest items that closely resemble what the user has previously shown interest in. In content-based recommendation systems, item similarity is determined by comparing their descriptions or characteristics. This allows the system to suggest items that share similar attributes to those that the user has already enjoyed.
- **Matrix factorization (MF)** is a mathematical technique used in recommendation systems to reduce the dimensionality of large matrices by decomposing them into lower-dimensional matrices. This decomposition helps identify latent factors or features that can be used to make personalized recommendations to users based on their preferences and behaviors.

Metrics. We utilize precision, recall and diversity metrics as performance indicators for item recommendations. To specify precision, recall and diversity of recommendations for a user u who obtains a list of N recommended items, we define Equations (7), (8) and (9), correspondingly:

$$P@N(u) = \frac{\text{number of relevant items in the top}N \text{ list}}{N} \quad (7)$$

$$R@N(u) = \frac{\text{number of relevant items in the top}N \text{ list}}{\text{total number of relevant items}} \quad (8)$$

$$ILD@N = \frac{\sum_{i=1}^n \sum_{j=i}^n (1 - \text{similarity}(c_i, c_j))}{N * (N - 1)/2} \quad (9)$$

where c_1, \dots, c_n are items in a set of recommendation list.

4.2 Comparative Results

4.2.1 Property selection impact for accuracy

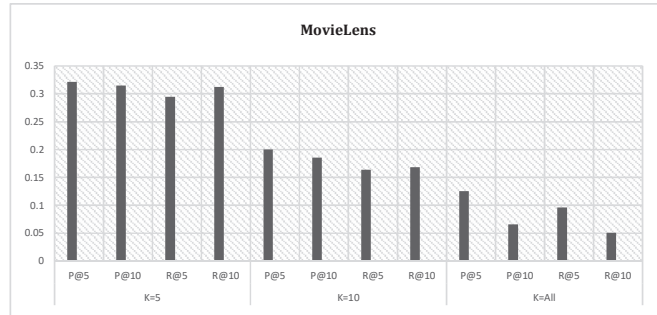
In this section, we respond to RQ1: *Can selecting specific properties result in an increase in recommendation accuracy?* To measure the impact of property selection on accuracy, we evaluate the performance of the proposed system based on the number of properties (K) used in the learning process, where K is set to 5, 10, and all (all means using all available properties). Our evaluation metrics are precision and recall, which are computed as the average of $P@N$ and $R@N$ across all users. The results of $P@N$ and $R@N$ with $N = 5$ and 10 on the MovieLens, LibraryThing, and Last.fm datasets are presented in Figure 2.

The results show that in the MovieLens domain, the proposed approach achieves the best performance when only 5 properties are used ($K = 5$). However, as the number of features used in training increases, the performance of the model decreases. Similarly, in the LibraryThing and Last.fm datasets, the proposed approach achieves the best performance when $K = 5$, which suggests that a smaller number of properties may be more effective in these domains.

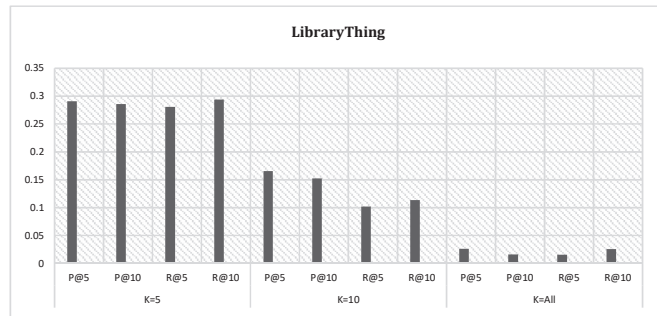
4.2.2 Comparison with baselines

In this section, we will address RQ2, which examines the performance of the proposed approach and baselines, and RQ3, which investigates the impact of knowledge graphs on the diversity of recommendations. Specifically, we compare proposed system variant with a parameter value of $K = 5$, which achieved the highest precision and recall scores in the previous test, against CB, I-KNN, and MF baselines. To assess the system's performance, we use precision, recall, and diversity metrics, which are computed as the average of $P@N$, $R@N$, and $ILD@N$ across all users. Tables 1(a), (b), and (c) present the experimental results for $P@N$, $R@N$, and $ILD@N$ with $N = 5$ and 10 on the MovieLens, LibraryThing, and Last.fm datasets.

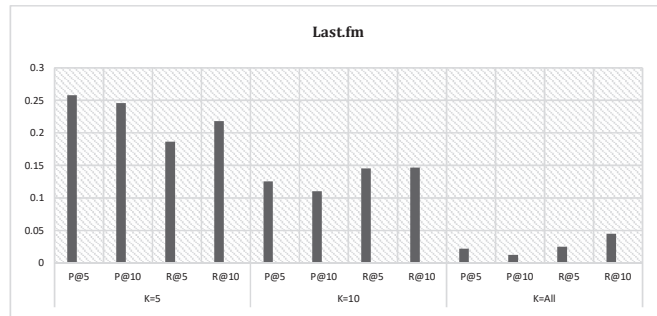
We can see from Tables 1(a), (b), and (c) that the proposed approach outperforms competing systems for all datasets in terms of precision, recall, and diversity. Based on Table 1(a) of the movielens dataset, it appears that the proposed approach achieves $P@5$ of 0.3214 and $P@10$ of 0.3150, while the other methods have much lower precision scores, with I-KNN being the lowest. In terms of recall at 5 and 10 recommendations ($R@5$ and $R@10$), the proposed approach also performs relatively well, achieving $R@5$ of 0.2944 and $R@10$ of 0.3127. Again, the other methods have much lower recall



(a)



(b)



(c)

Figure 2 Performance of the proposed according to the number of properties (K) used in the learning process. The three datasets (MovieLens, LibraryThing, and Last.fm) are evaluated using precision at N ($P@N$) and recall at N ($R@N$) with N set to 5 and 10. The available properties are split into three groups: $K = 5$, $K = 10$, and using all available properties (All). The results indicate that the proposed approach achieves the best performance when only 5 properties are used ($K = 5$) for all three datasets.

Table 1 Performance results for P@5, P@10, R@5, R@10, ILD@5, and ILD@10 on three datasets – MovieLens, Last.fm, and LibraryThing. The results show that the first variant of the proposed approach performs well on all datasets, with particularly impressive outcomes for the MovieLens dataset

System	P@5	P@10	R@5	R@10	ILD@5	ILD@10
(a) MovieLens						
Proposed approach V1	0.3214	0.3150	0.2944	0.3127	0.2623	0.2702
I-KNN	0.0763	0.0410	0.1569	0.1745	0.0045	0.0112
CB	0.1691	0.1433	0.1254	0.1405	0.0182	0.0633
MF	0.1164	0.1302	0.0261	0.1158	0.0216	0.0251
(b) LibraryThing						
Proposed approach V1	0.2912	0.2858	0.2810	0.2936	0.1691	0.2026
I-KNN	0.0323	0.0569	0.0612	0.0363	0.0123	0.0258
CB	0.0541	0.1360	0.1405	0.0586	0.0012	0.0023
MF	0.0472	0.1027	0.1137	0.0920	0.0140	0.0146
(c) Last.fm						
Proposed approach V1	0.2581	0.2459	0.1865	0.2180	0.1759	0.1791
I-KNN	0.0182	0.0177	0.1235	0.1244	0.0023	0.0036
CB	0.1562	0.1353	0.0658	0.0431	0.0014	0.0011
MF	0.0211	0.0180	0.0215	0.0324	0.0024	0.0074

scores, with I-KNN being the lowest. When it comes to the diversity of recommendations, measured by the intra-list diversity (ILD), the proposed approach still outperforms the other methods, but the gap is not as significant as in precision and recall. The proposed approach achieves ILD@5 of 0.2623 and ILD@10 of 0.2702, while the other methods have slightly lower ILD scores.

Similarly, for the LibraryThing dataset (see Table 1(b)), the proposed approach's first variant exhibited the highest performance across all evaluation metrics. It obtained a P@5 of 0.2581 and P@10 of 0.2459, followed by CB with a P@5 of 0.1562 and P@10 of 0.1353. Interestingly, the proposed approach performed less effectively in the ILD evaluation metric than in the recall and precision metrics.

Regarding the Last.fm dataset, the proposed approach performed well compared to the other methods but showed a lower performance than in the other datasets. It achieved a P@5 of 0.2581 and P@10 of 0.2459, followed by CB with a P@5 of 0.1562 and P@10 of 0.1353. Notably, the Item-KNN (I-KNN) method showed a relatively high performance in the recall evaluation metric (R@5 and R@10) but was significantly less effective in the precision and ILD metrics.

Overall, the results suggest that the proposed approach is effective in recommendation systems in terms of accuracy and diversity, particularly for the MovieLens and LibraryThing datasets.

4.3 Explainable Recommendation

In this section, we address RQ4. How effective is the proposed approach in providing reasonable explanations for user preferences towards items?

Users' confidence in a recommender system can be increased by providing them with an explanation of why a particular item is recommended, enabling them to make quick, informed decisions and promoting transparency. Our approach involves computing similarity scores between each recommended item and the user's profile using embedding vectors. The most relevant attributes are then selected to create explanations that clarify the user's preferences and the rationale behind the recommendations. In the movie domain, different templates for explainable sentences are employed based on the relationships, such as starring, director, genre, and subject. Examples of these templates include "You have previously liked movies featuring (Actor_Name)," "You have previously liked movies directed by (Director_Name)," "You have previously enjoyed (Genre_Label) movies," and "You have previously enjoyed (Subject_Label) movies." This method provides users with valuable explanations and facilitates the interpretation and analysis of the recommender system.

5 Conclusion

This paper proposed a novel entity and property knowledge-based graph attention network that leveraged rich background knowledge from the LOD cloud to achieve state-of-the-art performance on top-N recommendation tasks. The knowledge representation developed emphasized semantic preservation. The proposed algorithm, property-to-entity, transformed property edges into entity nodes to enhance the representation of properties in the graph, while a semantic-based information gain property selection method reduced the complexity of the GAT learning process. Experimental results on three real-world recommendation scenarios demonstrated the proposed approach's effectiveness in achieving better performance, providing effective explanations, and ensuring diversity of recommendations. There are several areas that could be explored in future work to further improve the effectiveness and practicality of the proposed approach. Future work includes

exploring the potential of incorporating additional background knowledge sources such as user-generated content or social networks. The proposed approach could also be extended to address cold-start recommendation scenarios, where limited data is available for new users or items. Further investigation is required to evaluate the performance of the proposed approach on datasets with varying levels of sparsity.

References

- A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, 'Translating embeddings for modeling multi-relational data', Proc. of the 26th Conference on Neural Information Processing Systems (NIPS), pp. 2787–2795, Lake Tahoe, Nevada, USA, ACM, 2013.
- A. Bouguettaya, H. Zarzour, A. M. Taberkit, A. Kechida, 'A review on early wildfire detection from unmanned aerial vehicles using deep learning-based computer vision algorithms', Signal Processing, 190, <https://doi.org/10.1016/j.sigpro.2021.108309>, 2022.
- A. Haghghat, A. Sharma, 'A computer vision-based deep learning model to detect wrong-way driving using pan-tilt-zoom traffic cameras', Computer-Aided Civil and Infrastructure Engineering, 38(1), <https://doi.org/10.1111/mice.12819>, 2023.
- B. Y. Lin, X. Chen, J. Chen, Ren, X, 'KagNet: knowledge-aware graph networks for commonsense reasoning', Proc. of the the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pp. 2829–2839, Hong Kong, China. ACL, 2019.
- C. Diamantini, A. Mircoli, D. Potena, E. Storti, 'Process-aware IIOT knowledge graph: a semantic model for industrial IOT integration and analytics', Future Generation Computer Systems, 139, 224–238, <https://doi.org/10.1016/j.future.2022.10.003>, 2023.
- C. Zhu, Y. Xu, X. Ren, B. Y. Lin, M. Jiang, W. Yu, 'Knowledge-augmented methods for natural language processing', Proc. of the 16th ACM International Conference on Web Search and Data Mining, pp. 1228–1231, <https://doi.org/10.1145/3539597.3572720>, 2023.
- D. Boughareb, A. Khobizi, R. Boughareb, N. Farah, H. Seridi, 'A graph-based tag recommendation for just abstracted scientific articles tagging', International Journal of Cooperative Information Systems, 29(3), doi: 10.1142/S0218843020500045, 2020.

- F. Bettina, L. Thomas, ‘Semantic Search on the Web’, *Semantic web*, 1(1,2), 89–96, 2010.
- F. Liu, Z. Cheng, L. Zhu, Z. Gao, L. Nie, ‘Interest-aware message-passing GCN for recommendation’, *Proc. of the Web Conference (WWW)*, pp. 1296–1305. Ljubljana, Slovenia, ACM, <https://doi.org/10.1145/3442381.3449986>, 2021.
- F. Ramezani, S. Parvez, J. P. Fix, ‘Automatic detection of multilayer hexagonal boron nitride in optical images using deep learning-based computer vision’, *Scientific Reports*, 13, <https://doi.org/10.1038/s41598-023-28664-3>, 2023.
- H. Arabi, V. Balakrishnan, N. L. Mohd Shuib, ‘A context-aware personalized hybrid book recommender system’, *Journal of Web Engineering*, 19(3-4), 405–428. <https://doi.org/10.13052/jwe1540-9589.19343>, 2020.
- H. Dong, T. Li, J. Leng, L. Kong, G. Bai, ‘GCN: GPU-based cube CNN framework for hyperspectral image classification’, *Proc. of the 46th International Conference on Parallel Processing (ICPP)*, pp. 41–49, Bristol, UK, IEEE, doi: 10.1109/ICPP.2017.13, 2017.
- J. Wu, H. Chen, F. Orlandi, Y. H. Lee, D. O’Sullivan, S. Dev, ‘Automated Climate Analyses Using Knowledge Graph’, *IEEE USNC-URSI Radio Science Meeting (Joint with AP-S Symposium)*, Singapore, 2021, pp. 106–107, doi: 10.23919/USNC-URSI51813.2021.9703620, 2021.
- J. Zhang, X. Shi, S. Zhao, I. King, ‘STAR-GCN: stacked and reconstructed graph convolutional networks for recommender systems’, *Proc. of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 4264–4270, Macao, China, 2019.
- K. Lei, M. Qin, B. Bai, G. Zhang, M. Yang, ‘GCN-GAN: A non-linear temporal link prediction model for weighted dynamic networks’, In *INFOCOM 2019 - IEEE Conference on Computer Communications*, pp. 388–396, Paris, France, IEEE, <https://doi.org/10.48550/arxiv.1901.09165>, 2019.
- K. Marino, R. Salakhutdinov, A. Gupta, ‘The more you know: Using knowledge graphs for image classification’. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20–28. Honolulu, HI, USA, 2017.
- K. Tu, P. Cui, D. Wang, Z. Zhang, J. Zhou, Y. Qi, W. Zhu, ‘Conditional graph attention networks for distilling and refining knowledge graphs in recommendation’, *Proc. of the 30th ACM International Conference on Information & Knowledge Management*, pp. 1834–1843, <https://doi.org/10.1145/3459637.3482331>, 2021.

- L. Asprino, E. Daga, A. Gangemi, P. Mulholland, 'Knowledge graph construction with a façade: a unified method to access heterogeneous data sources on the web', *ACM Transactions on Internet Technology*, 23(1), 1–31, <https://doi.org/10.1145/3555312>, 2023
- L. Wu, Q. Zhang, Chen, K. Guo, D. Wang, 'Deep Learning Techniques for Community Detection in Social Networks', *IEEE Access*, 8, 96016–96026, doi: 10.1109/ACCESS.2020.2996001, 2020.
- L. Xia, Y. Liang, J. Leng, P. Zheng, 'Maintenance planning recommendation of complex industrial equipment based on knowledge graph and graph neural network', *Reliability Engineering & System Safety*, 232, <https://doi.org/10.1016/j.res.2022.109068>, 2023.
- M. Paneque, M. Roldán-García, J. García-Nieto, 'e-LION: data integration semantic model to enhance predictive analytics in e-Learning', *Expert Systems with Applications*, 213(Part A), <https://doi.org/10.1016/j.eswa.2022.118892>, 2023.
- M. R. Islam, S. Liu, X. Wang, 'Deep learning for misinformation detection on online social networks: a survey and new perspectives', *Social Network Analysis and Mining*, 10(82), <https://doi.org/10.1007/s13278-020-00696-x>, 2020.
- N. Zhao, Z. Long, J. Wang, Z. Zhao, 'AGRE: A knowledge graph recommendation algorithm based on multiple paths embeddings RNN encoder', *Knowledge-Based Systems*, 259, doi: 10.1016/j.knosys.2022.110078, 2023.
- P. Cudré-Mauroux, 'Leveraging knowledge graphs for big data integration: the XI pipeline', *Journal of Semantic Web*, 11(1), 13–17, doi: 10.3233/SW-190371, 2020.
- P. Pham, L. T. T. Nguyen, N. T. Nguyen, R. Kozma, B. Vo, 'A hierarchical fused fuzzy deep neural network with heterogeneous network embedding for recommendation,' *Information Sciences*, 620, 105–124, <https://doi.org/10.1016/j.ins.2022.11.085>, 2023.
- P. Schneider, T. Schopf, J. Vladika, M. Galkin, E. Simperl, F. Matthes, 'A Decade of Knowledge Graphs in Natural Language Processing: A Survey', *Proc. of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing*, pp. 601–614, Online only, ACL, 2022.
- P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, Y. Bengio, 'Graph attention networks', In *6th International Conference on Learning Representations (ICLR 2018)*, Vancouver, BC, Canada, 2018.

- Q. Dai, X. Wu, L. Fan, Q. Li, H. Liu, X. Zhang, D. Wang, G. Lin K. Yang, 'Personalized knowledge-aware recommendation with collaborative and attentive graph convolutional networks', *Pattern Recognition*, 128(C), <https://doi.org/10.1016/j.patcog.2022.108628>, 2022.
- Q. Guo, F. Zhuang, C. Qin, H. Zhu, X. Xie, H. Xiong, Q. He, 'A survey on knowledge graph-based recommender systems', *IEEE Transactions on Knowledge and Data Engineering*, 34(8), 3549–3568, doi: 10.1109/TKDE.2020.3028705, 2022.
- R. Boughareb, H. Seridi and S. Beldjoudi, 'Explainable recommendation based on weighted knowledge graphs and graph convolutional networks', *Journal of Information and Knowledge Management*, 22(3), doi: 10.1142/S0219649222500988, 2023.
- R. De Donato, M. Garofalo, D. Malandrino, M. A. Pellegrino, A. Petta, 'Education meets knowledge graphs for the knowledge management', In: Kubincová, Z., Lancia, L., Popescu, E., Nakayama, M., Scarano, V., Gil, A. (eds) *Methodologies and Intelligent Systems for Technology Enhanced Learning*, 10th International Conference, Workshops (MIS4TEL 2020). *Advances in Intelligent Systems and Computing*, 1236, Springer, https://doi.org/10.1007/978-3-030-52287-2_28, 2021.
- R. Shimizu, M. Matsutani, M. Goto, 'An explainable recommendation framework based on an improved knowledge graph attention network with massive volumes of side information', *Knowledge-Based Systems*, 239, <https://doi.org/10.1016/j.knosys.2021.107970>, 2022.
- R. Wang, B. Li, S. Hu, W. Du, M. Zhang, 'Knowledge graph embedding via graph attenuated attention networks', *IEEE Access*, 8, 5212–5224, 2020.
- S. Deng, L. Huang, G. Xu, X. Wu, Z. Wu, 'On deep learning for trust-aware recommendations in social networks', *IEEE Transactions on Neural Networks and Learning Systems*, 28(5), pp. 1164–1177, doi: 10.1109/TNNLS.2016.2514368, 2017.
- S. G. Tesfagergish, R. Damaševičius, J. Kapočičūtė-Dzikienė, 'Deep learning-based sentiment classification of social network texts in amharic language', In: Zdravkova, K., Basnarkov, L. (eds) *ICT Innovations 2022, Reshaping the Future Towards a New Normal*, *Communications in Computer and Information Science*, 1740, Springer, https://doi.org/10.1007/978-3-031-22792-9_6, 2022.
- S. Oramas, V.C. Ostuni, T. Di Noia, X. Serra, E. Di Sciascio, 'Sound and music recommendation with knowledge graphs,' *ACM Transactions on Intelligent Systems and Technology*, 8(2), 1–21, doi: 10.1145/2926718, 2017.

- T. B. Mudiyansele, X. Lei, N. Senanayake, Y. Zhang, Y. Pan, 'Predicting CircRNA disease associations using novel node classification and link prediction models on Graph Convolutional Networks', *Methods Journal*, 198, 32–44, <https://doi.org/10.1016/j.ymeth.2021.10.008>, 2022.
- T. N. Kipf, M. Welling, 'Semi-supervised classification with graph convolutional networks', *Proc. of the 5th International Conference on Learning Representations (ICLR)*, pp. 2873–2879, Toulon, France, 2017.
- V. Ryen, A. Soylu, D. Roman, 'Building semantic knowledge graphs from (Semi-) structured data: A review', *Future Internet*, 14(5), <https://doi.org/10.3390/fi14050129>, 2022.
- W. Chen, W. Xiong, X. Yan, W. Wang, 'Variational knowledge graph reasoning', *Proc. of the 57th Annual Meeting of the Association for Computational Linguistics (ACL 2019)*, pp. 4185–4194, <https://doi.org/10.48550/arXiv.1803.06581>, 2019.
- X. Wang, K. Liu, D. Wang, L. W. Fu, X. Xie, 'Multi-level recommendation reasoning over knowledge graphs with reinforcement learning', *Proc. of the ACM Web Conference 2022*, pp. 2098–2108, <https://doi.org/10.1145/3485447.3512083>, 2022.
- X. Wang, X. He, Y. Cao, M. Liu, T. S. Chua, 'KGAT: knowledge graph attention network for recommendation', *Proc. of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*, pp. 950–958, Anchorage, USA, ACM, <https://doi.org/10.1145/3292500.3330989>, 2019.
- X. Zeng, X. Tu, Y. Liu, X. Fu, Y. Su, 'Toward better drug discovery with knowledge graph', *Current Opinion in Structural Biology*, 72, 114–126, <https://doi.org/10.1016/j.sbi.2021.09.003>, 2022.
- Y. Jiang, X. Gao, W. Su, J. Li, 'Systematic knowledge management of construction safety standards based on knowledge graphs: a case study in China', *International Journal of Environmental Research and Public Health*, 18(20), <https://doi.org/10.3390/ijerph182010692>, 2021.
- Y. Jin, W. Ji, Y. Shi, 'Meta-path guided graph attention network for explainable herb recommendation', *Health Information Science Systems*, 11(5), <https://doi.org/10.1007/s13755-022-00207-6>, 2023.

Biographies



Rima Boughareb is a PhD student at Badji Mokhtar – Annaba University, actively contributing to the research efforts of the esteemed Laboratory of Electronic Document Management (LabGED) Badji Mokhtar – Annaba University in Annaba, Algeria. She holds a Master’s degree in computer science from Annaba University (Algeria). She focuses on the semantic web, recommender systems, personalization, machine learning, and deep learning.



Hassina Seridi-Bouchelaghem is a full professor at the Computer Science department of Badji Mokhtar – Annaba University, Algeria and is affiliated to LABGED Laboratory. She has published several papers in international conferences and journals. Her research interests include information systems, recommender systems, e-learning, semantic web, social web, data mining and artificial intelligence.



Samia Beldjoudi is currently an Associate Professor at the National Higher School of Technology and Engineering, and a Researcher at LTSE Laboratory. She received her Ph.D. degree in computer science from Annaba University (Algeria) and is affiliated to LABGED Laboratory. She has published several papers in international conferences and journals. Her main research interests include social semantic web, personalization, recommender systems, e-learning, deep learning, prognostics, CMMS, predictive maintenance, and artificial intelligence. She is also collaborating on several national projects.