Chapter 5

Diversification In Recommendation System: Graph Neural Network based Approaches

5.1 Introduction

In the previous chapter, we were able to improve the diversity of recommendation results by using the ClusDR model, but due to the accuracy-diversity trade-off, we face the problem of reduced accuracy. In this chapter, we explore Graph Neural Networks (GNN) in an effort to achieve a better balance between accuracy and diversity. In recent years we have seen a lot of research on Graph Neural Networks (GNN), leading to substantial improvements in structured graph data, which serves as the basis of many of today's recommender systems. The core idea behind GNN is to aggregate feature information of graph structure and its spatial information into various learning models. Graph structure information is learned using different node representation learning methods that map each node into a low-dimensional vector representation.

These node embedding methods were developed for homogeneous graphs (graphs having a single type of nodes). In a heterogeneous information network, multiple types of nodes and structural relations exist between them. Recommendation systems have been proposed which follow the concept of the heterogeneous graph where there are multiple kinds of nodes (users and items) and structural relations (user-item interaction), among them [153]. In this chapter, we propose a Diverse Heterogeneous Network Embedding-based Recommendation (Div-HERec) technique that uses metapaths for diverse recommendation generation.

Most existing heterogeneous information networks define a series of metapaths that randomly select neighbours for the target node (user) and then calculate similarity and relevance between nodes to make recommendations. Instead of randomly selecting neighbours for target users, we include a graph colouring algorithm in a weighted user-user homogeneous graph to generate similar and dissimilar user pairs. We developed a joint representation of a series of metapaths in the node representation learning and recommendation generation model. Then we integrated them into a framework for edge prediction tasks for the user and unknown item relations for the target user.

The graphical abstract of our proposed approach is shown in Figure 5.1. The graphical abstract shows the User set (U_1, U_2, U_3), Item set (I_1, I_2, I_3) and Group Color Node (G_1, G_2). The proposed model uses two network graphs: A heterogeneous graph containing user, item, and group colour nodes and a second graph containing a homogeneous weighted graph for the users' node. Further user-user homogeneous graph is preprocessed from the heterogeneous graph of user-item interaction for metapath selection for the user. Still, the user is selected based on the graph colouring algorithm instead of random selection. For every HIN graph, we used more than four metapaths with the constraint that the first and the last node of the metapath would be the same. The metapaths are then used for generating the embedding of user and item nodes using metapath2vec. The detailed methodology for embedding generation is discussed in section 5.2.2.

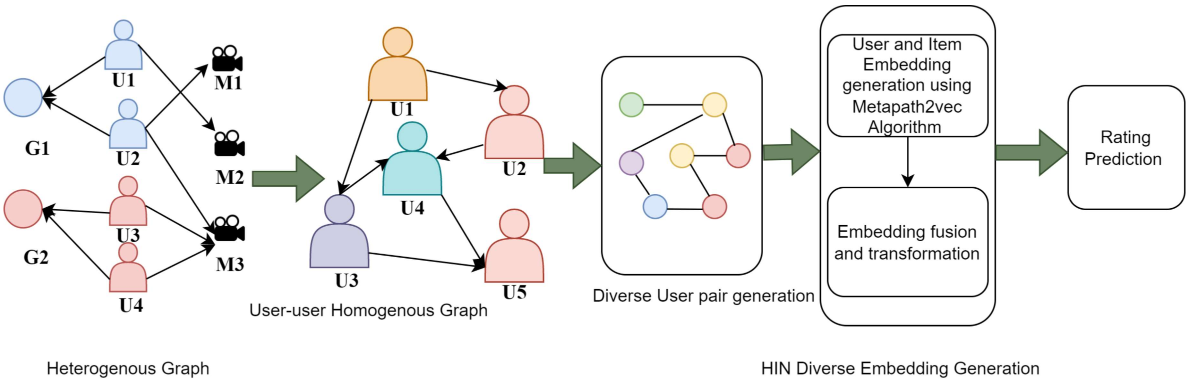


Figure 5.1: Graphical abstract of our proposed approach

5.2 Problem Formulation

This section explains the basic terminologies and definitions of heterogeneous information networks (HIN) and formally describes the application of HIN in recommendation generation.

5.2.1 Heterogeneous Information Network

Heterogeneous Information Network (HIN) is a relatively simple way of modelling distinct types of nodes and edges in a graph. HIN graphs are different from homogeneous graphs, where the nodes and edges are of a single type, respectively. A representative HIN graph is shown in Figure 5.2. Recently heterogeneous information networks have been used for the characterization and representation of auxiliary data, which also complies with their structural properties.

A HIN is defined as a directed graph as $\mathcal{G} = (\mathcal{V}, \mathcal{E}, T)$, where $\mathcal{V}$ is a set of vertices (represents user and item node) and $\mathcal{E}$ is a set of edges (represents rating relation between user and item) taking pairs of vertices from $\mathcal{V}$. In HIN, each node v and each edge e are associated with their mapping function. The mapping function for vertex mapping is defined as $\phi : \mathcal{V} \rightarrow T_v$ and the edge mapping function as $\psi : \mathcal{E} \rightarrow T_e$, where T_v, are set of nodes and T_e are defined as set of link types respectively, such that $|T_v| + |T_e| > 2$. A network schema $T = (T_v, T_e)$ is used to define the meta template describing the nodes and edges in a given heterogeneous information network. The network schema (T) is defined as directed graph described over object types T_v with edges as relation information T_e denoted as $T = (T_v, T_e)$. Figure 5.2 shows an example of a heterogeneous graph.

Example: In Figure 5.2 represents three types of nodes such that $\mathcal{V} = \{C_g, U, I\}$ where C_g is the set of group colour nodes (nodes having a same colour will be one group), U is the set of user nodes and I is the set of item nodes. We also have two different types of relations $\mathcal{E} = \{R_c, R_r\}$, where R_c is the set of colour relations between users and the R_r is the set of rating relations between a user and an item. The user-user homogeneous graph shows the colour relation between different users. The heterogeneous graph constitutes all user, group, and colour nodes related to rating relations between users and movies. In Figure 5.2, we have a heterogeneous graph for Users (U), Movies (M) and Color-Group (G) as node types and edges indicated rating relation (U-M) between User→ Movie and Color relation (U-G) between User→Color-Group.

User-User Graph: It is the graph $G_u = (V_u, E_u)$ generated from HIN considering only the set of users as vertices $V_u = \{v_1, v_2, \ldots v_n\}$. An edge $e = (v_i, v_j)$ between two users v_i and v_j (both $\in V_u$) exists in G_u only when similarity score between users is more than threshold (λ). The similarity score is calculated based on the ratings provided by the users on the common items in their past interactions.

5.2.2 Meta-path based embedding

The relationship among different nodes in a HIN is more complex than any other homogeneous network graph. The proximity between nodes in a HIN is not based on similarity or closeness (distance) between nodes but on the semantics of various nodes. The metapath2vec algorithm is proposed for modelling the proximity in HINs [38]. The Metapath2vec is the random walk-based skip-gram model for low-dimensional vector representation method for node representation. The random walk in a HIN is

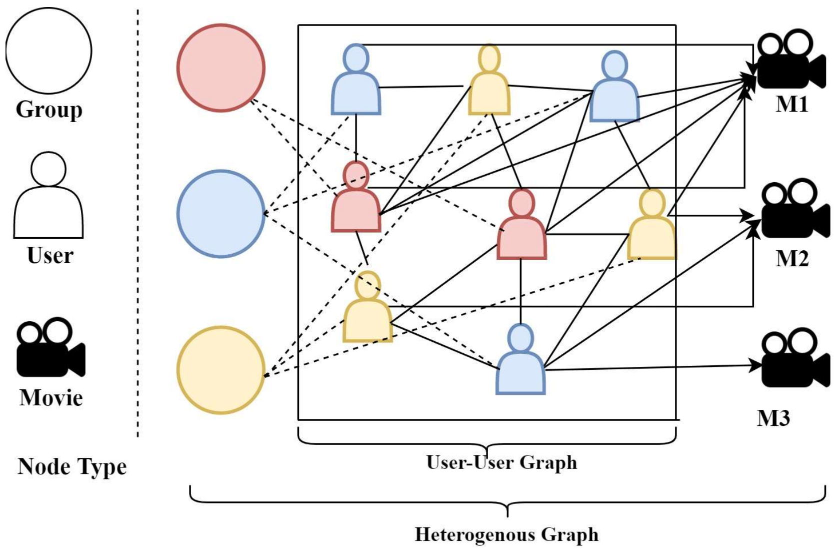


Figure 5.2: An illustrative example of a heterogeneous graph (MovieLens) where three different types of nodes are users, movies, and groups and two kinds of connections: a) rating relations between users and movies, and b) colour relation (edges between any two users from the same group) between users.

to develop metapaths of multiple types of nodes. A meta path is an order of node types (users/items) with edge types for modelling a distinct relationship. For example, a metapath scheme P is defined on the graph $\mathcal{G}$ of network schema $T = (T_v, T_e)$ as $P : V_0 \xrightarrow{R_1} V_1 \xrightarrow{R_2} V_2 \dots V_t \xrightarrow{R_t} V_t + 1 \dots \xrightarrow{R_l} V_l$. The nodes $V_0, V_1, \dots V_l$ can be of different types as defined in T_v, and similarly, edges $R_1, R_2, \dots R_l$ are also one of the edge-types defined in T_e.

Every instance p of a metapath P can be defined as a sub-sequence of different nodes and edge types in the metapath P. The composite relation between various node types is defined as $R_1 \circ R_2 \circ R_3 \dots R_l$: the composite relation between various nodes. The random walk in a HIN graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, T)$ and network schema T for embedding generation is defined using the transition probability $p(v^{i+1}|v_t^i)$, is as described below:

$$p(v^{i+1}|v_t^i, P) = \begin{Bmatrix} \frac{1}{|\mathcal{N}_{t+1}(v_t^i)|} & (v^{i+1}, v_t) \in \mathcal{E}, \phi(v^{i+1}) = t + 1 \\ 0 & (v^{i+1}, v_t) \in \mathcal{E}, \phi(v^{i+1}) \neq t + 1 \\ 0 & (v^{i+1}, v_t) \notin \mathcal{E} \end{Bmatrix} \quad (5.1)$$

In Equation 1 $v_t^i \in V_t$, $\mathcal{N}_{t+1}(v_t^i)$ represents V_{t+1} node type of neighbourhood of node v_t^i. The condition here is that the flow of the random walk is dependent on the previous metapath and the first and last node of the metapath is the same. In addition, $v^{i+1} \in V_{t+1}$ represents the random walk flow depends on the meta-path P and also follows the condition that the starting node type V_1 is similar to the ending node type V_l of the l length metapath means its recursive guidance for a random walker is:

$$p(v^{i+1}|v_t^i) = p(v^{i+1}|v_t^i) \text{ if } t = l \tag{5.2}$$

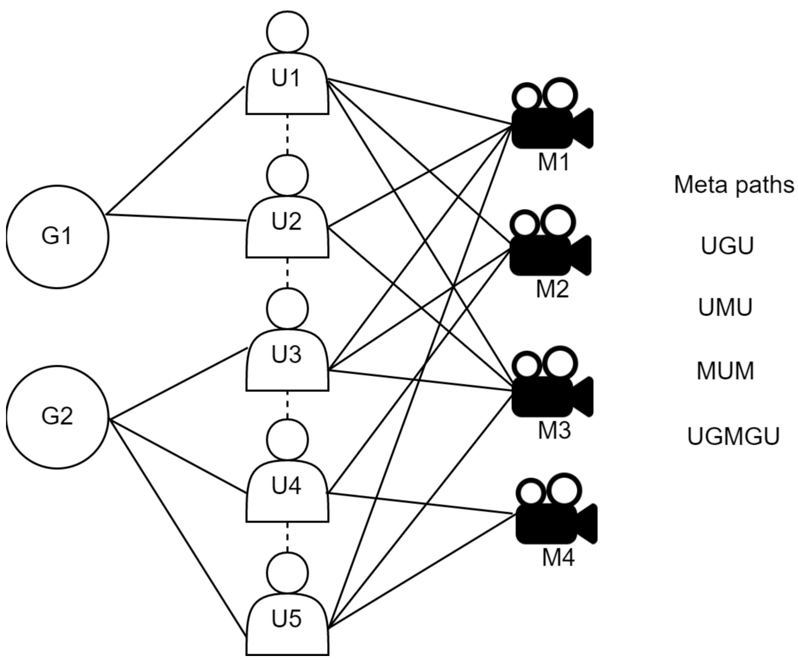


Figure 5.3: An illustrative example is a heterogeneous graph (MovieLens) where we have a network of five users interacting with four different movies. The users are categorized into two-colour nodes.

The random walk-based metapath-based embedding technique provides semantic relation between various node types in HIN which is incorporated into embedding. Each metapath we use in our proposed model ensures its own relationship like User-Movie-User-Movie denotes the social relationship for users who watched similar movies in the HIN. The length of a metapath P is the number of node relations present in it. The metapath "User-Movie-User" have a length of 2. Every metapath has its own semantics in it. For example metapath "User-Movie-User" shows the semantics of users who watched the same movies.

For a better understanding of metapath2vec embedding generation for heterogeneous graphs, we have an illustrative example of MovieLens in Figure 5.3, where we have three types of nodes: User, Movie, Group, and their interaction relation. The first is the user and group relation which is obtained using graph colouring algorithm discussed in Algorithm 3. The second interaction is the user-item interaction obtained from the user-item interaction history. For example Figure 5.3 we have $|V| = 11$ and we need to predict items (movies) for user U_4. The metapaths defined for the heterogeneous graph are $P = (UGU, UMU, MUM, GUMUG)$, where for example path UMU represents User-Movie-User. So based on the metapaths defined for heterogeneous graph, we need to find the neighbourhood for target node U_4, the neighbourhood of node at the time step i $\mathcal{N}^i_{t+1}$ are $(G_2, U_3, U_2, U_5, M_1, M_2, M_3, M_4)$ because G_2 is the same color node for users U_3, U_5. The equation 5.1 represents the transition probability at step i for metapath schema P. The Metapaths are used in that the path's first and last node is the same, which facilitates recursive guidance for the random walk.

The heterogeneous skip-gram objective function for the effective representation of the network structure. We use metapath2vec with a skip-gram model for effective node representation in the heterogeneous graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, T)$. For heterogeneous graph with $T_v > 1$ (more than one node type) by maximising the node's probability of retaining the heterogeneous context $\mathcal{N}_t(v)$, $t \in T_v$ for target node v.

$$\arg\max_{\Theta} \sum_{v \in \mathcal{V}} \sum_{t \in T_v} \sum_{c_t \in \mathcal{N}_t(v)} logP(c_t|v;\Theta) \tag{5.3}$$

Here is equation 5.3 $N_t(v)$ are the neighbours of node v and $logP(c_t|v;\Theta$ is softmax function.

$$P(c_t|v;\Theta) = \frac{e^{X_{ct}}.X_v}{\sum_{u\in\mathcal{V}} e^{X_u}.X_v} \tag{5.4}$$

Here, equation 5.4 X_v denotes the embedding vector of node v.

5.2.3 Graph colouring

Recently graph colouring problem has received substantial attention. Broadly this interest develops from the fact that many real-world optimization problems can be solved using graph colouring algorithms. There are two types of graph colouring problems: edge colouring and vertex colouring. In our proposed approach, we only consider the vertex colouring algorithm. In the vertex colouring problem, the aim is to find a colouring solution such that any two adjacent vertices do not have the same colours. In our proposed work, we use graph colouring to minimize the number of colours at one edge distance of any vertex. The traditional vertex colouring algorithm for a provided graph G_u is to give colours to all the graph vertices with the minimum possible colours so that no two adjacent vertices have the same colour. In graph theory a proper vertex colouring with minimum k colours is a mapping $f : V(G_u) \rightarrow N$ such that $\forall_{v_i,v_j\in V(G_u),i\neq j}\exists(e_i, e_j) \Rightarrow f(i) \neq f(j)$.

In our proposed model Div-HERec we use a heterogeneous graph where we have various nodes representing users and movies, and the edge represents their relationship. We formulated our problem for diverse user selection in the homogeneous graph of the user. Graph colouring aims to widen the user's selection scope to encompass diverse items in the recommendation system. The concept behind using graph colouring in user-user similarity graphs is to produce user pairs with the same colour. In metapath selection for a target node, the node will traverse for most similar users and specific lengths.

So to include diversity, we widen the user pairs by having graph colour so that in a minimum length metapath, the user with a similarity score and the user having the same colour node will also be included. An example of this illustration is shown in Figure 5.4. For applying the vertex colouring algorithm in our proposed method Div-HERec, we need to pre-process our data to a user-user similarity graph (Homogeneous graph). In this graph, the node represents a user, and an edge represents a similarity value between them. We used a Greedy k-colouring Algorithm because we want a short-sighted way of making proper colouring with as few colours as possible. The idea behind the graph colouring algorithm is to colour each user node with minimum k possible colours. The graph between users is constructed using the similarity matrix of users. The similarity between users is calculated using a cosine similarity measure. Once we have the similarity between users, we build a homogeneous graph $G_u = (V_n, E_m)$ where V_n represents total user nodes and E_m represents edges between them.

5.2.4 Problem Definition

For HIN based recommendation system, the information can be modelled by heterogeneous network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, T, \phi, \psi)$ having different users and item nodes and their relation in terms of rating information. Let $\mathcal{V}_u$ denotes the user set and $\mathcal{V}_i$ denotes the item set and $R = (u, i, r_{ui})$ denotes the rating relation (r_{ui}) for user u and item i. The implicit feedback matrix (R) where we set the entry r_{ui} as follows:

$$r_{ui} = \left(\begin{smallmatrix} 1 & \text{if user u has rated the item i} \\ 0 & otherwise \end{smallmatrix} \right) \tag{5.5}$$

In our proposed approach, we have two types of relations. First: rating relation R_r between user and items and second: colour relation between users for corresponding heterogeneous network $\mathcal{G}$ and user homogeneous graph representation G_u. The objective of our proposed approach is to predict the preference for the non-rated item set $v_j \in \mathcal{V}_i$ for target users $u_i \in \mathcal{V}_u$. The proposed model is based on a personalized recommendation problem where the target user will get relevant and diverse recommendations in their recommendation list. We formulate this problem in a heterogeneous information network with graph colouring and a metapath algorithm. Here the graph colouring algorithm is used to include the diverse user in the metapath node embedding generation. After that, these node embeddings are implicitly used in a hybrid matrix factorization framework for recommendation generation.

5.3 Proposed Methodology

This section explains our proposed methodology of diverse recommendation generation. First, we present the node embedding generation method for users and items, and then we obtain their combined representation for recommendation generation.

5.3.1 Graph colouring based diverse user selection

The graph colouring algorithm is a particular case of graph labelling where we assign labels to the graph in terms of colours with some specific constraint. The graph colouring problem is divided into two problems: the vertex colouring problem and the edge colouring problem. Here in our proposed approach, we use a vertex colouring problem where the vertex is the user. In the vertex colouring problem, we use a greedy colouring algorithm to colour the graph in which two adjacent vertices do not have a similar colour.

A great many algorithms have been proposed for graph colouring [154–157]. In our proposed approach, we adopted the greedy graph colouring algorithm. We adopted the vertex greedy colouring algorithm for the user-user undirected graph. The user-user undirected graph is created using a similarity matrix. In greedy vertex colouring, the vertex of a graph is selected consecutively. The sequence order is decided by some set of rules and determines the set of minimal possible colours from the colour set and allocates it to vertices such that the colour of two adjacent vertices is different. In our proposed method, vertex order is chosen based on the similarity value between two nodes (users). The higher values between two user nodes are selected first, then the vertices are ordered in decreasing order of similarity value between nodes. For user-user graph we need to set threshold value for edge formation between two users. In our proposed method we set threshold value 0.7. The users having more than 0.7 similarity value will be included into the user-user graph.

An arbitrary sequence of the users is given like (U_1, U_2, U_3, U_4) of all the user node vertices of graph $G_u = (V_n, E_m)$. A greedy colouring algorithm assigns a color to each vertex from U_1 to U_n, using the minimal possible color value that has not already been assigned to one of its adjacent vertexes. The color assignment for the sample user graph is shown in Figure 5.4. In Algorithm 3 we present the pseudocode for the greedy vertex colouring algorithm for graph colouring.

Algorithm 3: Greedy Vertex colouring Algorithm

Input: $G_u = (V_n, E_m)$

Output: Colored Graph $G_u = (V_n, E_m)$

Calculate k ; `// chromatic number`

1. Give an order to the set of k colours.
2. Select the first vertex from the vertex order from the graph, assign it the first color from the colour set.
3. Considering the next vertex from the graph, assign it to the lowest ranked color that has not been assigned to any adjacent vertex for the target node.
4. Repeat step 3 until the graph is coloured.

5.3.2 Meta-path based diverse embedding generation

Figure 5.4 shows the demonstration of the graph colouring used for the user-user undirected graph. An essential part of graph colouring is making diverse user pairs for embedding generation. First, we discuss how the traditional metapath algorithm will choose the paths for an embedding generation. Then, we will discuss how we include a graph colouring algorithm to select the diverse metapaths for an embedding generation. The traditional metapath-based random walk algorithm metapath2vec is used for the representation of graph nodes [38]. The metapath2vec uses the skip-gram model methodology in the heterogeneous information network $\mathcal{G}(\mathcal{V}, \mathcal{E}, T)$ with $|T_V \geq 1|$ by

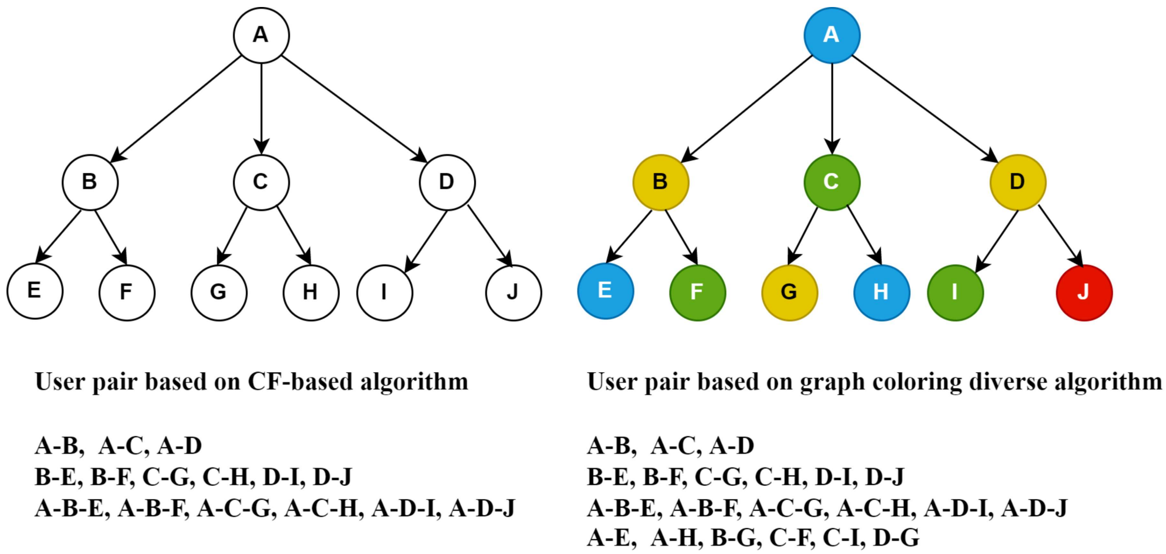


Figure 5.4: Demonstration of user pair selection using graph colouring based diverse user selection algorithm

maximizing the heterogeneous context's probability for target node v. In HIN, we use the random walk method to generate the representation of the nodes using the skip-gram model. The random walk methods aim to calculate the similarity between nodes by considering random walks between nodes. Let a random walk starts with node i and recursively proceed to a random neighbourhood node with probability α and return to node i with probability $(1-\alpha)$. Let r_j^t be the transition probability that the node is at j^{th} position at time t. Then, an iterative equation is defined as follows:

$$r_{t+1} = \alpha P.r^{(t)} + (1-\alpha)t \tag{5.6}$$

In this equation, P denotes the transition probability matrix and t denotes the teleport vector and $t = r_{(0)}$ indicates that the element of t is all zeros besides the starting node elements i for which $t_i = 1$. In the meta path-based method, metapath is used for HIN to estimate the similarity between nodes to generate recommendations for target users. For example, in Figure 5.4, the first tree graph is used for a traditional meta path-based algorithm. In traditional metapath, selection will be based on random path selection. This is demonstrated in Figure 5.4, where the first tree graph shows the general graph of the user and when it applies to the HIN network path is chosen randomly. Let us take an example of a user and movie bipartite network where two nodes are present. Example metapaths are U-M-U, and M-U-M, which follow the traditional collaborative filtering approach concept where similar users will get equivalent recommendations [158]. The metapath U-M-U shows the social connection between two users who rated the same movie, and M-U-M shows movies rated by a common user.

Considering only similar users and movies in metapath selection will lead to similar recommendations based on their past interactions with the user. To address these issues in the recommendation, we included graph colouring in the metapath selection. We select the neighbour based on the colouring algorithm instead of randomly selecting the next neighbour for the target user to have diversity in our model. For illustration, the second tree graph is shown in the Figure 5.4.

For metapath selection, we need to pre-process our model for diverse user pair selection. Unlike in collaborative filtering or any other recommendation algorithm, we only consider the nearest neighbour node (similar users) for node embedding generation. In our proposed model, we include user pairs using an algorithm 4 where the neighbour of the target user is not only a most similar user but other similar users with the same colour from the graph colouring algorithm. This will include most similar as well as less similar users in the metapath selection.

The Meta-path selection and diverse user pair selection for the diverse embedding are shown in Algorithm 5.3.

Since our goal is to improve the performance of our recommendation model in terms of accuracy and diversity both at the same time, we include some constraints in metapath selection. The selection of metapath is based on the assumption that it will consist of similar neighbors and dissimilar neighbors in the metapath section. First, we select metapath having similar node types (users/movie/color) at the start and the

Algorithm 4: Pseudo-code for diverse user pair selection

Input: The user-user similarity graph $G_u = (V_n, E_m)$, τ, Similarity Matrix (S_{u_i,u_j})
Output: paths=[]

paths=[];
for $(i = 0, i \leq N, i++)$ **do**
  for $j = 0, j \neq i, j++)$ **do**
    if $S_{u_i,u_j} \geq \tau$ **then**
      $S_{u_i,u_j} = 1$
    else
      $S_{u_i,u_j} = 0$
    end
  end
  return u_i, u_j
end
Function `Greedy-Color`(u_i, u_j)**:**
  if $color(node(u_i)) = color(node(u_j))$ **then**
    path=(u_i, u_j);
  else
    Discard
  end
return path;

Algorithm 5: The heterogeneous information network embedding generation

Input: The heterogeneous information network $\mathcal{G}(\mathcal{V}, \mathcal{E}, T)$, The user-user similarity graph $G_u = (V_n, E_m)$, walk-length (wl), target node type T_v the neighbourhood size (N_v), the number of walks per node r
Output: User Embedding E_U and Item Embedding E_I for recommendation

paths=[];
for *each node* $v \in V$ *and* $\phi(v) == T_v$ **do**
  for *i= 1 to r do* **do**
    paths =[];
    while $wl > 0$ **do**
      walk for target node x using equation 5.5. **if** $\phi(x) == T_v$ **then**
        append node x into path; $wl \leftarrow wl - 1$
      end
    end
    paths[]= path.append()
  end
end
$E_U, E_I = SGD(paths, d, ns)$
return E_U, E_I

end of the metapaths. Next, we select users' node types that adhere to the same colour scheme as that discussed in section 5.4 for metapaths. Apart from a random selection of neighborhoods in metapaths, we apply one more constraint, the neighbor node has a similar color with the target node. The objective function for representation of node in this scenario is defined in equation 5.7.

$$max_f \sum_{\substack{u \in v \\ (color(u)=color(v))}} logP(\mathcal{N}_u | f(u)) \tag{5.7}$$

where f is function mapping node to d dimensional embedding representation and $\mathcal{N}_v$ is the neighbour of target node v. In our proposed methodology, the combined representation of different node types for recommendation is obtained using different operators which is discussed in section 5.3.3.

5.3.3 Embedding Integration and Recommendation Generation

Since our goal is to generate a recommendation using node embedding obtained from the Algorithm 5, we need a joint representation for the node to give a meaningful presentation for recommendation generation. The architecture of our proposed methodology (Div-HERec) is shown using a flow-chart in the Figure 5.5. After pre-processing the dataset for the HIN graph, we first generate a user-user homogeneous graph using similarity score and then generate diverse user pairs using graph colouring algorithm. The selection of diverse user pair using graph colouring and metapath generation for our proposed model discussed in Section 5.3.1, 5.3.2 are the pre-processing steps for our proposed model Div-HERec. Further we use these metapaths for embedding generation of the user and item using metapath2vec algorithm. The only restriction in metapath is that the first and last node will be the same. If the first node is a user, then the last node will also be a user and similarly for the item metapath if first node is item node then last node will also be the item node. We obtain user E_u and item E_i embedding

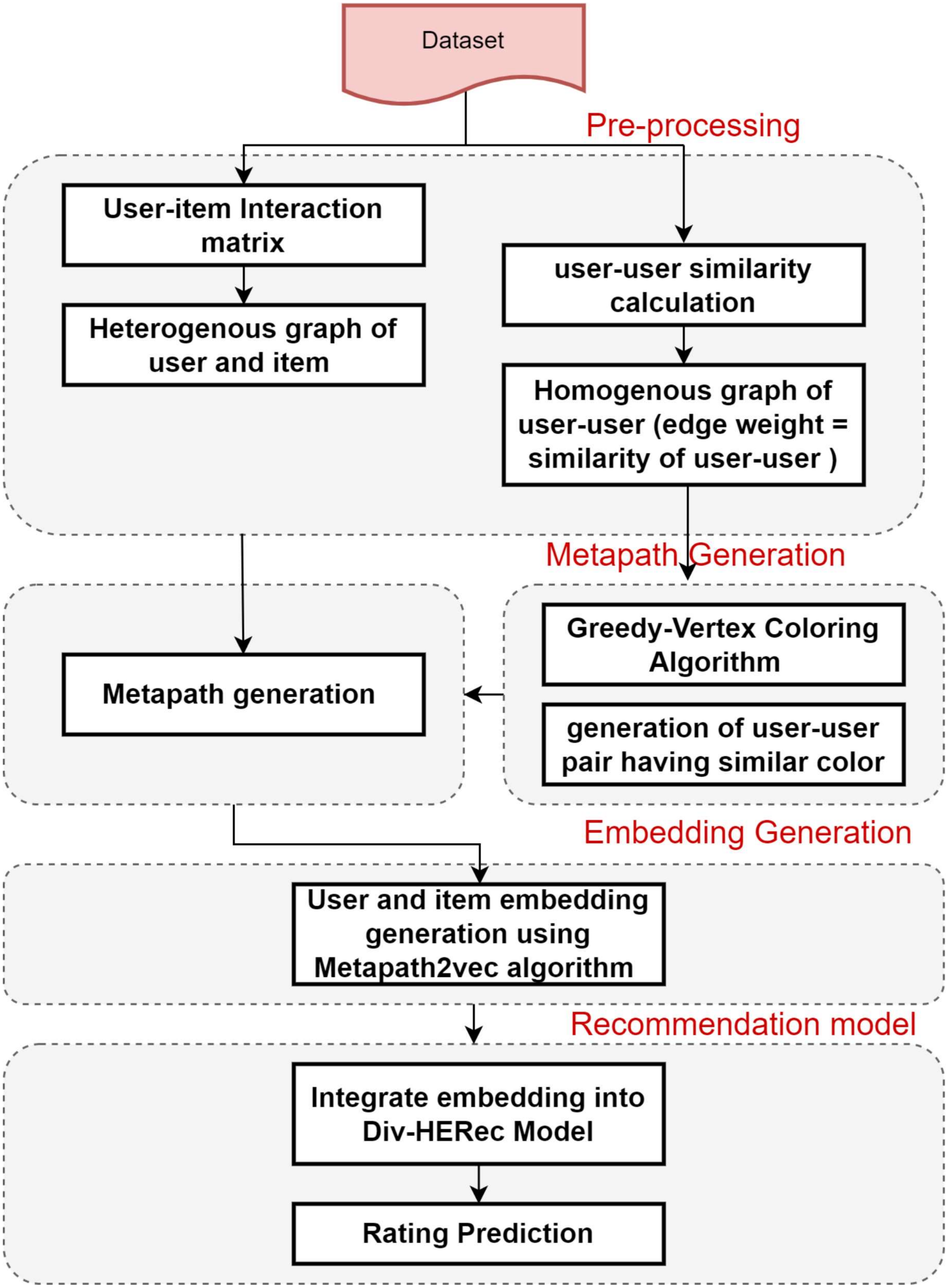


Figure 5.5: Flow-chart for our proposed Div-HERec model

for heterogeneous graph embedding using various operators. Therefore we use a general integration method to use these user embeddings and items embeddings into a joint representation. The details of these operators are as follows:

1. concat - Concatenation
2. hadamard - element wise multiplication of embedding of node (u_i, v_i)
   $h(u_i, v_i) = u_i, v_i$
3. L1 operator- $L1(u_i, v_i) = |u_i, v_i|$
4. L2 operator- $L21(u_i, v_i) = (|u_i, v_i|)^2$

After generation of user and item embeddings, we utilize these embedding into classic matrix factorization framework to obtain rating prediction [159]. The basic groundwork for the matrix factorization is that it includes user and item embeddings for final rating prediction. Let there be m users and n items, and the rating matrix be R_{ui}. The basic matrix factorization methodology is used to divide the rating matrix R into a low-rank d-dimensional matrix described as follows:

$$R = U.I^T \tag{5.8}$$

where $U \in \mathbb{R}^{m \times d}$ and $I \in \mathbb{R}^{d \times n}$. Here U denotes users and I are the representation of item in a low-dimensional space. The rating prediction of user i for an item is computed as follows:

$$\hat{r}_{i,j} = U_i \times I_j^T \tag{5.9}$$

Here U_i denotes user latent factor and I_j denotes item latent factors. After obtaining the latent factors of the users and item, we obtain the rating prediction of our model using these latent factors and HIN embedding into a joint model:

$$\hat{r}_{i,j} = U_i \times I_j^T + \alpha.\sigma(E_u^U E_i^I) \tag{5.10}$$

Here α is the hyper-parameter used for integration of these latent factor and HIN embeddings into a model. The σ is the sigmoid function in our proposed model. The $\hat{r}_{i,j}$ is the predicted rating of our proposed model.

Objective Function

In our proposed method, we use the implicit feedback information of users for recommendation generation. The method for recommendation generation is similar to the BPR (Bayesian Personalized Ranking) methodology. We endeavor to maximize area under the curve (AUC-Score) in our proposed method. The AUC score is used to predict the likelihood of the random relevant items that are ranked higher than random irrelevant items. A higher AUC score means a better recommendation system.

$$AUC - Score(u) = \frac{1}{|I_u^+| \left| \frac{I}{|I_u^+|} \right|} \sum_{i \in I_u^+} \sum_{i \in \frac{I}{I_u^+}} \delta(r_{uij} > 0) \tag{5.11}$$

Here in equation 5.11, i is for the positive items from the set I_u^+ and j is for the negative items from the set $\frac{I}{I_u^+}$:

$$r_{uij} = r_{ui} - r_{uj} \tag{5.12}$$

where r_{uij} is the preference for item i over j for user u.

The optimization of the AUC-score is done using the squared loss function to overcome the over-fitting problem of model [40].

5.4 Experiments

This section describes the results for the proposed model through various experiments on diverse datasets. To quantify the accuracy of our proposed model, we use AUC Score and NDCG measures. For diversity, we adopted the aggregate diversity measure, and for coverage, we use item coverage. The metrics we used for evaluation are explained in the section 2.4.

5.4.1 Datasets and pre-processing

We use various datasets from different domains to prove our model's significance. We used the MovieLens dataset [1] for the movie domain and the LastFM dataset from the music domain. For the book domain, we use Douban dataset [2] and the Yelp dataset [3] from the business domain. Our proposed method works on implicit feedback of the user so we can apply this method on any domain. The datasets [4] we used in our approach are the same as used by [115].

We use binary rating data for adjacency matrix creation where if the user rated that movie/book/song, then we have 1 else 0 in the matrix. The MovieLens 100k dataset which includes 943 users and 1682 distinct movies with 100000 interactions. We used the LastFM dataset, where 1892 users have listened to 17563 songs with 92834 interactions. We also used the Douban Book dataset which contains 13024 users and 22347 books with 792026 rating details ranging from 1 to 5. Another dataset is the Yelp business dataset of 16239 users, and 14282 businesses and 198397 ratings are available. Statistics of the datasets used in our proposed model are described in table 5.1.

Table 5.1: Features for considered dataset

Dataset	Users	Items	Ratings
MovieLens-100k	943	1682	100000
Douban Book	13024	22347	792026
LastFM (2k)	1892	17632	92834
Yelp	16239	14282	198397

In our proposed method, we need to preprocess our dataset for the graph colouring

[1] https://grouplens.org/datasets/movielens/
[2] https://book.douban.com/
[3] https://github.com/librahu/HIN-Datasets-for-Recommendation-and-Network-Embedding
[4] https://github.com/librahu/HIN-Datasets-for-Recommendation-and-Network-Embedding

component, where we need a user-user homogeneous graph. For the user-user homogeneous graph, we need to preprocess our dataset to apply the graph colouring algorithm discussed in section 5.3.1 in Algorithm 3. The users' pairs are selected based on the similarity score between them. For similarity calculation, we used the Pearson co-relation coefficient. The statistics of user pairs in the dataset are discussed in Table 5.2.

Table 5.2: Statistics of user pairs in datasets

Dataset	pair	counts
MovieLens(100k)	user-user	47150
Douban book	user-user	169150
LastFM(2k)	user-user	18802
Yelp	user-user	158590

Table 5.3: The selected metapaths and entity description for various datasets

Dataset	**Entity**	**Metapaths**
MovieLens	User-U	UGU, UMU, MUM,GUMUG
	Movie-M	UGU, UMU, MUM,
	Color-G	UGU, UMU
		UGU
LastFM	User-U	UGU, UMU, MUM,GUMUG
	Music-M	UGU, UMU, MUM,
	Color-G	UGU, UMU
		UGU
Yelp	User-U	UGU, UMU, AUA,GUAUG
	Business- A	UGU, UAU, AUA,
	Color-G	UGU, UAU
		UGU
Douban Book	User-U	UGU, UBU, BUB,GUBUG
	Book-B	UGU, UBU, BUB,
	Color-G	UGU, UBU
		UGU

5.4.2 Div-HERec Structure

The overall structure of our proposed Div-HERec model consists of four components, followed by pre-processing of our model for heterogeneous and homogeneous graph construction. The structure of our proposed model Div-HERec is shown in Figure 5.6. The components of our model are preprocessing of the data for Diverse user-pair selection from the homogeneous weighted user-user graph, Metapaths generation, Metapath2vec embedding generation for the user and item node, and then training our model for preference prediction for unrated items.

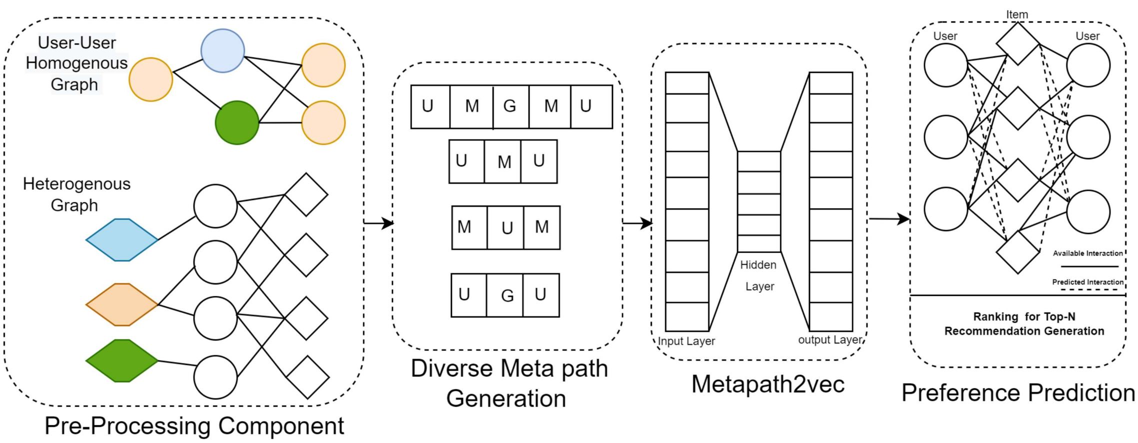


Figure 5.6: Structure for our proposed Div-HERec model

5.4.3 Baseline and hyper-parameter setting

The proposed Div-HERec model is based on a heterogeneous information network used for diverse recommendation generation using various metapaths and a deep matrix factorizing approach. The effectiveness of our proposed model is measured using various evaluation metrics used for the recommendation system evaluation. The various measure used in our model performance is discussed in Section 2.4. We use the AUC score and NDCG measure for our model's performance evaluation, and for diversity, we use the intralist diversity measure and catalogue coverage for each target user's recommendation.

For comparative analysis of our model, we use the baseline algorithm discussed below:

1. **User-based Collaborative Filtering (User CF)** [47]: User-based CF predicts

user's preferences for items based on the assumption that similar users will like similar items in their recommendation list. The preferences are nothing but the rating of the non-rated item, which is estimated as $r_{ui} = \sum_{u \epsilon U} sim(u, u') \times r_{u'i}$. The similarity between users is calculated using cosine similarity.

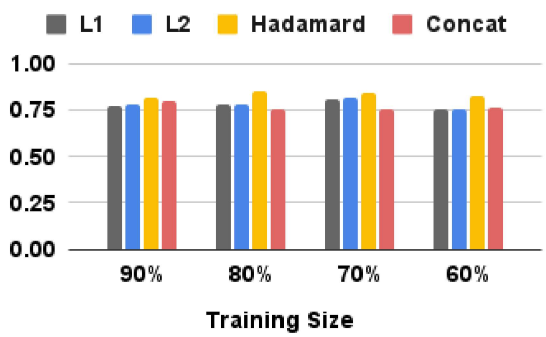
2. **Item-based Collaborative Filtering (Item CF)** [160]: The general idea of item-based CF is mostly similar to the user-based CF, the only difference is that we calculate item similarity instead of user similarity. The ratings calculated as $r_{ui} = \sum_{i \epsilon I} sim(i, i') \times r_{ui'}$.
3. **Non-Negative Matrix Factorization (NMF)** [161]: Based on the concept of matrix factorization, NMF calculates two non-negative latent factor matrices of the user and item whose product is more or less related to the original matrix.
4. **Bayesian Personalized Ranking (BPR-MF)** [162]: BPR-MF is a ranking optimization method proposed for a recommendation system where we learn to rank the personalized recommendation generated using MF using implicit feedback.
5. **Path-constraint Random Walk** [163]: The path constraint random walk method is based on a single metapath in a heterogeneous information network. We followed the same pre-processing proposed in the paper for our methodology comparison. The model uses a single metapath for the recommendation, and each metapath has its own semantic meaning, so this method is also called a semantic recommendation model.
6. **xQuAD**: Explicit Query Aspect Diversification proposed by Abdollahpouri et al.: [36] proposed a methodology to improve recommendation and popularity bias using personalized diversification. The model uses re-ranking methodology using a sub-profile diversification to balance the accuracy-diversity tradeoff.
7. **Div2vec** [41]: The method includes an embedding technique for node representation for diverse recommendation generation. The div2vec method is based on sampling nodes with probability inversely proportional to its degree so that in

the random walk, every node will appear adequately.

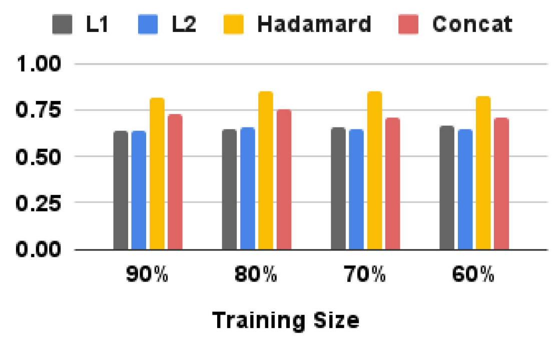
8. **Div-HeteRec** [44]: The proposed Div-HeteRec model uses a non-Mrkovian random walk over a heterogeneous network to introduce diversity in the recommendation system. The random walk is used to boost the weights of the influential nodes and shrinks the weights of the other node.

9. **DivRank**: A framework for diverse ranking in networks was introduced by DivRank, which was developed by Mei et al [164]. A random walk model based on a Markovian process was taken into consideration in earlier strategies like Personalised Page-Rank. In contrast, DivRank takes into account a non-Markovian process-based vertex-reinforced random walk model. With more visits, the likelihood of arriving at a node increases. As a result, a single node absorbs the score of its densely linked neighbours, resulting in a network of diversified nodes with high ranks.

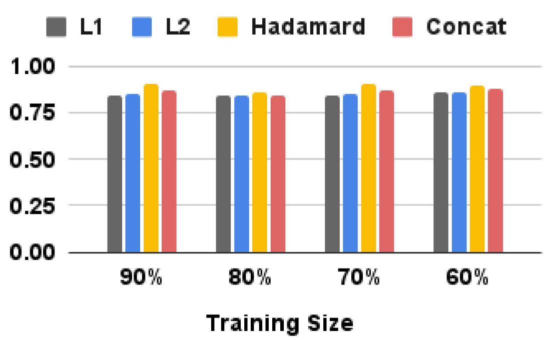
Our proposed model uses implicit feedback data for recommendation generation, so our model is dataset independent. We tested our model performance on four datasets belonging to various domains, e.g., movie, music, business, and book, for experimental analysis. Our model considers it a graph where the user entity is connected to the item entity, and the connection between these entities is their relation. We need to find the missing links between the user and the item entity for recommendation generation. The implementation of our model is based on three parts. First, we need to create a homogeneous user graph and use the graph colouring algorithm discussed in the Algorithm 3, and then we form user-user pairs so that we get both similar and dissimilar users at one go. Second, we need to find the embedding representation of every node (user and item) using the metapath2vec algorithm discussed in the Algorithm 5.2. For node representation, we use various user-generated metapaths, the detailed description for metapaths used for a different dataset is shown in Table 5.3. Next, we use the metapath2vec algorithm with metapaths and diverse user-user pairs to obtain each node's embedding representation. For diverse user pair selection, the threshold for the similarity value is 0.70%. In the user-user homogeneous graph, we included the user's relation having more than 0.7 similarity value. In node representation learning, we use the random walk node2vec embedding technique with various metapaths. We set walk length = $[10, 20, 50, 100, 150]$, epoch = 10, embedding dimension = $[10, 20, 50, 100, 150, 200]$ and the number of walks = 5 for our complete experimental analysis. The context window size for embedding generation is 10.



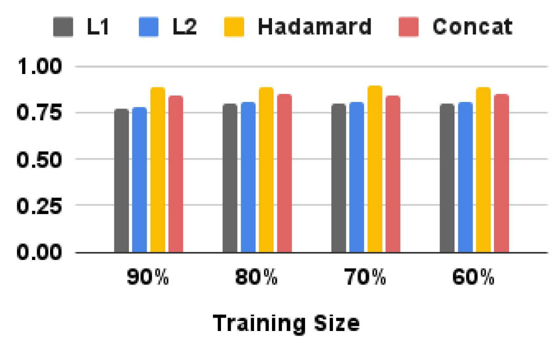
(a) Movielens Dataset



(b) LastFM Dataset



(c) Yelp Dataset



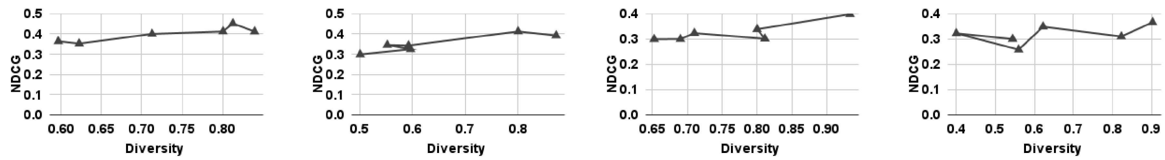
(d) Douban Book Dataset

Figure 5.7: Performance analysis of Div-HERec model for various datasets with different training sizes

Table 5.4: Comparative analysis of our proposed Div-HERec model

	Method	Metapath	AUC	Precision	Recall	NDCG	Diversity	Coverage
ML-100k	[47]		0.6911	0.1572	0.1639	0.3321	0.4875	43.22
	[160]		0.6844	0.1361	0.1593	0.3647	0.5322	49.22
	[161]		0.6221	0.1602	0.1163	0.3441	0.4882	39.45
	[162]		0.7885	0.2668	0.1822	0.3955	0.5122	42.25
	[163]	UM	0.5788	0.2108	0.1282	0.1335	0.2362	15.66
		UMU	0.5622	0.1969	0.1473	0.1225	0.2293	14.22
		MUM	0.5336	0.2183	0.1406	0.1445	0.3002	16.53
		UMUMU	0.5889	0.2667	0.1346	0.1142	0.2881	17.53
	[36]		-	0.1792	0.1034	0.1822	0.4771	48.55
	[41]		0.8011	0.3435	0.1987	0.3002	0.7522	65.44
	[44]		0.8273	**0.3046**	**0.2642**	0.4355	0.7258	70.65
	[164]		0.8552	0.2998	0.2322	0.4442	0.7122	69.51
	Div-HERec		**0.8487**	0.2422	0.1996	**0.4122**	**0.7933**	**71.11**
Douban Book	[47]		0.5592	0.1900	0.1776	0.2944	0.3922	40.22
	[160]		0.5966	0.3214	0.2877	0.3001	0.3447	41.23
	[161]		0.6127	0.2215	0.2102	0.3884	0.3998	45.33
	[162]		0.7028	0.1792	0.2007	0.3117	0.4225	40.22
	[163]	UM	0.4002	0.1209	0.0987	0.1336	0.2247	13.22
		UMU	0.4955	0.1542	0.1042	0.1255	0.2336	19.55
		MUM	0.5001	0.1598	0.1153	0.1445	0.1998	14.55
		UMUMU	0.5224	0.2155	0.1998	0.1142	0.2011	16.85
	[36]		-	0.2422	0.1258	0.2042	0.5773	39.88
	[41]		0.7622	0.3665	0.2766	0.3101	0.7866	68.22
	[44]		0.8689	0.4958	0.3017	0.4644	0.6982	73.28
	[164]		0.8821	0.2776	0.2013	0.4223	0.6887	66.11
	Div-HERec		**0.8911**	**0.5532**	**0.3220**	**0.4662**	**0.8221**	**76.23**
LastFM	[47]		0.6014	0.2456	0.1170	0.3001	0.4772	32.21
	[160]		0.6447	0.2841	0.0271	0.3122	0.4115	38.22
	[161]		0.7336	0.231	0.1218	0.3365	0.3922	47.22
	[162]		0.7445	0.3212	0.1658	0.4021	0.4339	52.11
	[163]	UM	0.6001	0.2029	0.1997	0.1889	0.2554	19.22
		UMU	0.5993	0.2367	0.2154	0.1998	0.2441	20.22
		MUM	0.5799	0.2377	0.2269	0.2003	0.2336	21.98
		UMUMU	0.6112	0.2394	0.2388	0.2201	0.2011	20.33
	[163]		-	0.1877	0.1243	0.1324	0.5446	52.87
	[41]		0.7901	0.304	0.2404	0.32214	0.7122	70.12
	[44]		0.8668	0.4395	0.2674	0.4587	0.7132	69.58
	[164]		0.8332	0.2666	0.1998	0.3992	0.6882	56.22
	Div-HRec		**0.8585**	**0.5592**	**0.3628**	**0.4225**	**0.8221**	**85.33**
Yelp	[47]		0.5778	0.2058	0.211	0.32214	0.5622	41.22
	[160]		0.6001	0.2182	0.198	0.2887	0.4922	38.23
	[161]		0.6822	0.2359	0.2014	0.2447	0.511	33.25
	[162]		0.5922	0.2288	0.2235	0.3221	0.5569	38.93
	[163]	UA	0.4722	0.2345	0.2311	0.2001	0.3665	20.33
		UAU	0.4722	0.2229	0.2558	0.2114	0.3554	22.27
		AUA	0.4992	0.2192	0.2366	0.1993	0.3328	23.22
		UAUAU	0.4822	0.2386	0.2531	0.2366	0.3012	24.62
	[163]		-	0.2398	0.1326	0.2239	0.5773	39.88
	[41]		0.8033	0.2593	0.3233	0.2239	0.7922	56.91
	[44]		0.7596	0.2547	0.3015	0.3988	0.7669	61.38
	[164]		0.8902	0.3012	0.2122	0.3997	0.5667	52.91
	Div-HERec		**0.9000**	**0.2691**	**0.3535**	**0.4338**	**0.8336**	**76.81**

Note: xQuAD is a post-processing method for diversity



(a) Movielens Dataset (b) LastFM Dataset (c) Yelp Dataset (d) Douban Dataset

Figure 5.8: Accuracy-diversity trade-off of Div-HERec model for various datasets

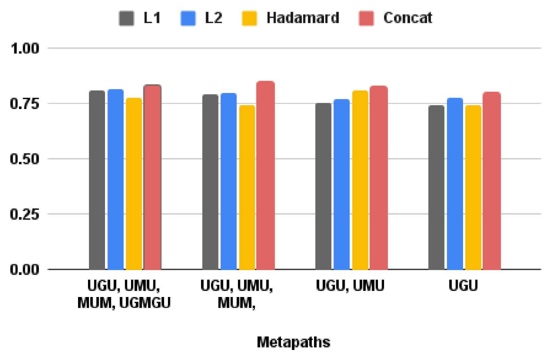
5.4.4 Results

This section will discuss the details of our proposed model Div-HERec and comparative results with various baseline approaches discussed in Section 5.4.3. We use AUC-Score to check the accuracy of our proposed model. The AUC-Score of our proposed Div-HERec model is shown in Table 5.4. Apart from the accuracy analysis of our proposed model, we also compared our model performance with several state-of-the-art methods of recommendation generation. The experimental analysis of our proposed model Div-HERec is shown in Table 5.4. We used four datasets from different domains to check the effectiveness of our proposed model. We get the best results on the Yelp business dataset, where we obtained almost 12% more accurate results. For comparative analysis, we use two different state-of-the-art approaches. First, we use accuracy-centric approaches for recommendation generation, e.g. UserCF, ItemCF, NMF and BPR-MF. These approaches are the best-performing approaches for accurate item recommendations. Second, we use a diversity-based approach for the recommendation system, i.e., div2vec, where the primary objective of the proposed approach is to generate a diverse recommendation for the target user.

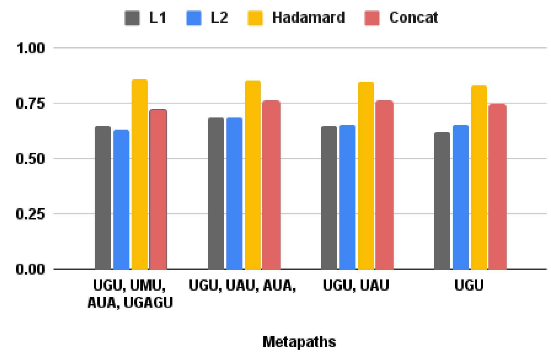
For comparative analysis, we use the AUC score as the performance measure of the system and the NDCG score for accurate item recommendations. In our proposed model Div-HERec, the primary objective is to provide an accurate and diverse recommendation to the target user. The proposed Div-HERec model performance for various datasets with different training ratios are shown in Figure 5.7. We used four different operators for joint embedding representation of users and items in which the Hadamard operator performs better for every dataset. For performance evaluation of our proposed model Div-HERec, we use a train-test split ratios of 90%, 80%, 70%, and 60% to evaluate the model performance. Again, the performance of our model is best for the Yelp dataset, where we got the AUC-Score 0.9000. The AUC score is suitable for both the accuracy-centric state-of-the-art approach and the diversity-centric recommendation approach. Apart from AUC-Score, we also use the NDCG measure to evaluate the recommendation quality. In our proposed model Div-HERec, instead of relying only on accuracy, the primary focus of our model is to provide an accurate and diverse recommendation. So we also tested our model's performance in terms of diversity and coverage. The complete result is shown in Tables 5.4. The results shown in Tables 5.4 show our model's effectiveness in diversity and coverage as well we got almost 21.4% improvement in coverage for a diversity-centric baseline and 63.45% improvement in coverage for accuracy-centric baseline. Similarly, in terms of diversity, we obtained 5% for diversity centric baseline and 58.8% improvement for accuracy centric baseline.

5.4.5 Accuracy-diversity trade-off

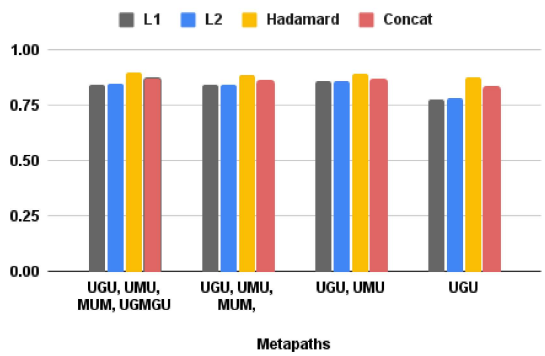
We quantified accuracy through the AUC-Score (higher is better) and the NDCG measure (higher is better), and diversity through the intralist diversity measure (higher is better) to evaluate our proposed approach. We also study the trade-off of accuracy diversity for our proposed model Div-HERec. In our proposed method, we achieved accuracy as well as diversity in recommendation generation. We followed the same hyper-parameter setting discussed in Section 5.4.3. To test the trade-off of the accuracy-diversity in our model, we use NDCG and diversity score for the top-n recommendations. We use the top-n value to (5,10,20,50,100). The trade-off for various datasets for our proposed model is shown in Figure 5.8.



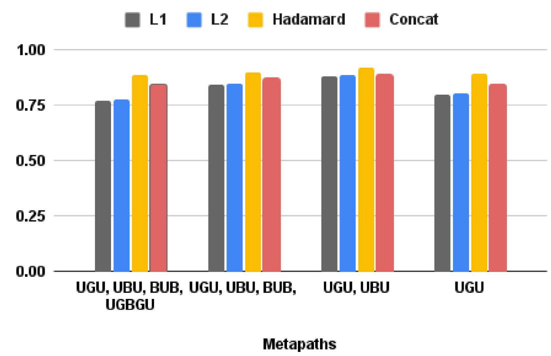
(a) Movielens Dataset



(b) LastFM Dataset



(c) Yelp Dataset

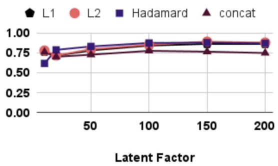


(d) Douban Dataset

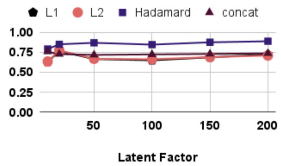
Figure 5.9: Performance analysis of Div-HERec model for various datasets with gradually decreasing metapaths constraints

Table 5.5: Performance analysis of Div-HERec for metapath2vec (without diverse user pair selection)

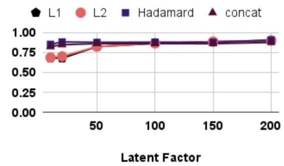
Dataset	AUC Score	Precision	Recall	NDCG	Diversity	Coverage
MovieLens	0.7093	0.4332	0.2665	0.4125	0.8002	65.44
LastFM	0.5914	0.4221	0.2336	0.3456	0.5524	70.12
Yelp	0.6906	0.3397	0.2879	0.3489	0.6221	56.91
Douban	0.79641	0.4008	0.3052	0.3014	0.8112	68.22



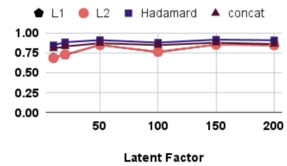
(a) MovieLens Dataset



(b) LastFM Dataset

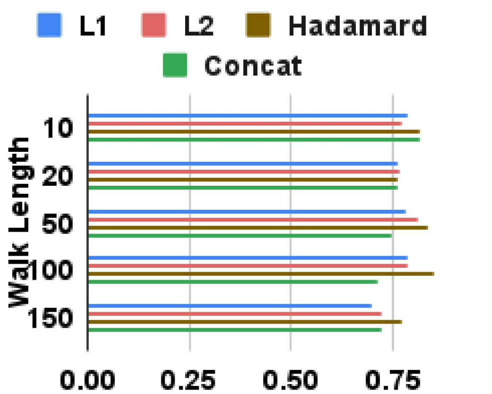


(c) Yelp Dataset

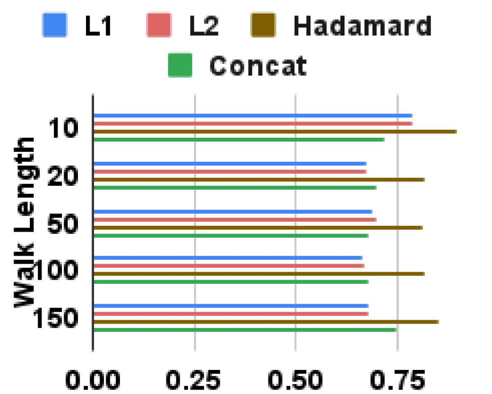


(d) Douban Book Dataset

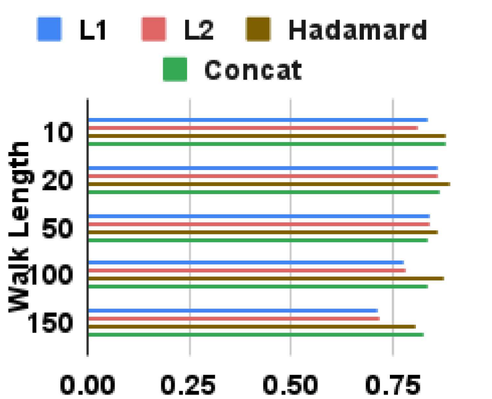
Figure 5.10: Performance analysis of Div-HERec model for various datasets with varying latent factor sizes



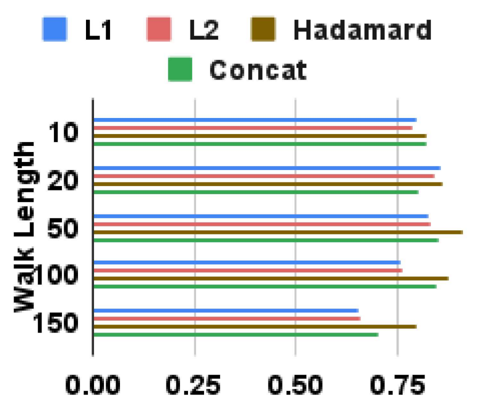
(a) Movielens Dataset



(b) LastFM Dataset



(c) Yelp Dataset



(d) Douban Dataset

Figure 5.11: Performance analysis of Div-HERec model for various datasets with different walk length constraints

5.4.6 Implication of Research

In this study, we proposed a model for diversified recommendation generation for users, keeping the model's accuracy at the same time. The necessity for diverse and unknown recommendation generation for users has become a prominent research direction in recommendation systems. An earlier state-of-the-art model like collaborative filtering, where the generation of similar items from the users' past interaction history became outdated where target users are not satisfied with the items that are merely similar to what they preferred before. This raises a research question about how to tailor a diversified recommendation for the target users to their requirements. Inclusion of diversity in the recommendation model leads to an accuracy-diversity tradeoff, where diversity enhancement in the model leads to a decrease in accuracy for the recommendation generation. To overcome this issue, many state-of-the-art approaches have been proposed, including diversity by including dissimilar and similar users for recommendation generation of the target user. Including distinct users for recommendation generation will fail to provide the right balance for the accuracy-diversity tradeoff. We tried to have diverse user pairs using graph colouring algorithms to get a better balance in the accuracy-diversity tradeoff, instead of including dissimilar users for the diverse recommendation generation. Our hypothesis behind containing diverse user pairs is to dilate the item domain for the target user. We followed that hypothesis in our proposed methodology and evaluated our process over four different datasets. The experimental results are shown in Table 5.4 in support of our claim.

5.5 Ablation Study

In this section, we study the effect of various hyper-parameters on the performance of our proposed Div-HERec model.

1. **Effect of diverse user pairs**: We have evaluated our model's performance

for the non-diverse user pairs. We followed the general paradigm for metapath selection without applying the graph colouring component in our proposed model. The results are shown in Table 5.5.

2. **Effect of lengths of metapaths**: One of the major drawbacks of the metapath2vec algorithm is that a larger path length reduces the model's generalization ability. We evaluated our model's performance for various metapaths in Figure 5.9. We evaluated our model's performance for four different metapaths settings described in Table 5.3. The performance of our model is evaluated using AUC-Score and the score for various metapath is shown in Figure 5.9. The longest path we use in of length 5, which gives the best AUC Score. So we concluded that gradually increasing the metapaths in our proposed model increases our model's performance.

3. **Effect of walk-length**: The next ablation study we perform with the number of random walks for embedding generation of each node and their impact on our model Div-HERec. We tested our model for walk length (10,20,50,100,150). The results in term of AUC-Score for our model for various integration operators is shown in Figure 5.11. From Figure 5.11 we concluded that walk-length 50 is the best performing for our model.

4. **Effect of latent dimensions**: To further analyze the effect of the latent factor size in recommendation performance in our proposed approach. We tested our proposed model Div-HERec performance our different latent factor sizes (10,20,50,100,150,200). The performance of our model is best for latent factor 150. The complete performance of our proposed model is shown in Figure 5.10.

5. **Effect of training ratio**: To further analyze the effect of training ratio on our proposed model Div-HERec and other state-of-the-art models, we analyze the performance on various split ratios for training and testing ratios (90-10, 80-20, 70-30). The AUC-Score for different training and testing ratios is shown in figure

5.12. From figure 5.12, we notice for most of the approaches, the results are highest for the training ratio of 70-30 for our pre-processed datasets.

6. **Model performance for changing item-item homogeneous graph**: In our suggested model, Div-HERec, we pre-process the model to generate diverse user pairs. We have replaced this step by changing the diverse item pairs instead of generating diverse user pairs. The following preprocessing is used to create diverse item pairs and include diverse item pairs:
   (a) First, we preprocessed our item-item homogeneous graph for adjacency matrix creation for item-item dissimilarity calculation. Where nodes in the graph represent items and edges connecting these nodes represent the co-occurrence between items. The co-occurrence is the number of users who rated both items [113].
   (b) Next, we calculate the similarity between items using the cosine similarity measure.
   (c) After that, we followed the same procedure as discussed in Section 5.3.
   (d) Detailed experimental results are shown in Table 5.6.

Table 5.6: Performance analysis of Div-HERec for metapath2vec for item-item graph

Div-HERec	**Dataset**	**AUC Score**	**NDCG**	**Diversity**	**Coverage**
With user-user graph	ML-100k	0.8487	0.4122	0.7933	71.11
	Douban Book	0.8911	0.4662	0.8221	76.23
	LastFM	0.8585	0.4225	0.8221	85.33
	Yelp	0.9000	0.4338	0.8336	76.81
With item-item graph	ML-100k	0.8001	0.4099	0.6966	66.32
	Douban Book	0.8256	0.3887	0.7424	70.53
	LastFM	0.8223	0.4123	0.7825	74.47
	Yelp	0.8383	0.4223	0.8002	71.67

5.6 Statistical Significance Test

We also conducted a statistical significance test as part of evaluation of the recommendation method. We conducted a Friedman test at a significance level of 0.05. The

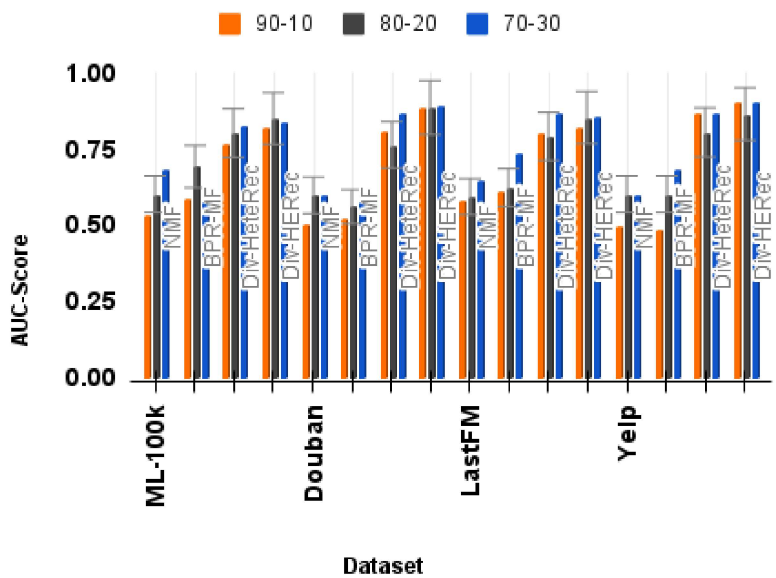


Figure 5.12: Comparative analysis for the different training ratios for the ML-100k dataset.

Table 5.7: Friedman test results on Div-HERec for significance test for MovieLens dataset

Div-HERec	Rank	UserCF	Rank	ItemCF	Rank	BPR-MF	Rank	div2rec	Rank
0.8112	5	0.6911	3	0.6889	2	0.6221	1	0.7555	4
0.7952	5	0.5366	1	0.6441	3	0.6011	2	0.7405	4
0.7555	5	0.5922	2	0.5966	3	0.5596	1	0.7555	4
0.7405	4	0.6011	1.5	0.6100	3	0.6011	1.5	0.7441	5
0.8133	5	0.6112	2	0.6822	3	0.6001	1	0.7922	4

analysis was conducted for the AUC-Score of the top-5 approaches. The proposed Div-HERec model performed significantly better for all co-occurrence-based methods, including UserCF, ItemCF, NMF, and BPR-MF, with the diversity-based method div2vec. For five samples of AUC-Score with multiple executions of our model and baseline algorithms. The Friedman statistical analysis for our model, Div-HERec on MovieLens dataset, is shown in Table 5.7. The test results show our model is significant for $p < 0.5$.

The post hoc examination of the Friedman test is conducted when the test produces significant results. The post hoc breakdown of the Friedman test is done using two different methods.

- Nemenyi Test
- Conover Test

Nemenyi Test: The Nemenyi test is a post hoc analysis test intended to find the data groups that vary from any global statistical test. The test is described as unplanned comparisons and controls for family-wise error. The test makes pair-wise tests of performance.

$$q = \frac{R_{max} - R_{min}}{s.e.} \tag{5.13}$$

$$s.e. = \sqrt{\frac{nk(k-1)}{12}} \tag{5.14}$$

k = the number of groups and n = the size of each of the group samples. The sample size for groups is equal for all. The result for Nemenyi posthoc analysis for the Friedman test is shown in Table 5.8.

5.7 Summary

This paper proposes a model for diversified recommendations using a graph colouring algorithm. The proposed model Div-HERec uses a metapath2vec algorithm for node

Table 5.8: Nemenyi Post-hoc test for Friedman statistical test

	0	1	2	3	4
0	1.0000	0.0306	0.2659	0.0042	0.9000
1	0.0306	1.0000	0.8945	0.9000	0.1446
2	0.2659	0.8945	1.0000	0.5540	0.6108
3	0.0042	0.9000	0.5540	1.0000	0.0306
4	0.9000	0.1446	0.6108	0.0306	1.0000

representation learning. We further used these node representations for recommendation generation. The proposed model addresses the problem of the recommendation system accuracy-diversity tradeoff. We did not lose on accuracy for diversity enhancement in the recommendation generation. This paper proposes a model for diversified recommendations using a graph colouring algorithm. The proposed model Div-HERec uses a metapath2vec algorithm for node representation learning. We further used these node representations for recommendation generation. In the future, we plan to investigate how other node representation methods with content information can be applied to increase the diversity of the model.