



Diversity-Ensured Semantic Movie Recommendation by Applying Linked Open Data

Uma Shankari Srinivasan^{1*} Chidambaram Mani²

¹*Department of Computer Science, Bharathiar University, Coimbatore, India*

²*Department of Computer Science, Rajah Serofiji College, Thanjavur, India*

* Corresponding author's Email: umabalajes@gmail.com

Abstract: The recommender system becomes a significant research area due to the popularity of the social web. Traditional semantic recommender systems deliver poor performance when balancing the recommendation accuracy and diversity. Also, the rank-based recommendation methods lack to obtain the coverage of the entire preferences of the user in the top-N recommendation list. Thus, this paper presents the Diversity-Ensured Semantic-aware Item REcommendation (DESIRE) that deals with the consistent and reliable knowledge source to significantly improve the quality and provide the diversity-ensured top-N recommendation list. The DESIRE approach builds the semantically relevant graphs such as movie-centric and user rating-centric graph with the help of both the Linked Open Data (LOD) and the explicit ratings of the users. By extracting the semantic-path based features from the user rating-centric graph, it executes the ranking algorithm for the top-N movie recommendation. Moreover, the diversity-aware re-ranking tends to maintain the trade-off between the diversity and accuracy in the top-N recommendation.

Keywords: LOD, Movie-centric graph, User rating-centric graph, Semantic-path, learning to rank, Top-N movie recommendation, Diversity.

1. Introduction

The high popularity of World Wide Web and Social networks has led to the dramatic growth of information. The Web information overload challenges the users during the process of decision-making. As a consequence, to cope with the information overload, the users increasingly demand an efficient filtering technology such as recommender systems [1] to assist the users to obtain the desired information in a personalized manner. In recent years, Recommender systems have become an essential tool in a variety of online services such as e-commerce and online entertainment. It intends to filter out potentially essential items such as movies from a vast amount of information based on the user behaviors and feedback. Several recommendation techniques have attempted to achieve better recommendation accuracy such as collaborative, content-based

filtering, and hybrid filtering [2]. However, the conventional movie recommender systems [3, 4] still encounter the problems in accurately predicting the unknown rating and recommending the relevant movies to the users. To overcome this obstacle, several research works have presented the semantic information based movie recommender systems. Semantic recommender systems play a crucial role in enhancing the traditional recommendation approaches with the help of ontologies. Ontologies effectively discover the semantic features in the recommendation process due to the existence of underlying relations of every movie.

Even though ontology-based semantic recommender systems explores the latent features, the cold start and data sparsity constraints are still in its infancy stage while predicting the preferences of the users. Recently, to tackle such constraints, few of the recommendation efforts focus on exploiting the freely available Web data such as Linked Open

Data (LOD) [5]. LOD [6] is a huge decentralized knowledge base that comprises the updated information having links to related data. With the explosive information availability of LOD source, recommender systems attain greater benefits rather than the domain ontology and taxonomy restricted conventional recommender systems. However, extracting the latent features from a vast LOD source is an arduous as well as time-consuming task. Also, accurately ranking the top-N movies according to the preferences of the user is a challenging task while modelling the ranking as the recommendation problem [7]. Recently, the concept of Learning to Rank (LtR) plays a crucial role in precisely developing such a user-item recommendation model. The LtR techniques are categorized into three types such as point-wise, pair-wise, and list-wise ranking methods. Collaborative filtering method with list-wise LtR suggests a list of items for each user based on the list-wise preferences. It intends to directly optimize several list-wise ranking evaluation measures such as Mean Reciprocal Rank (MRR), Mean Average Precision (MAP), and Normalized Discounted Cumulative Gain (NDCG). Even though list-wise LtR techniques [8] improve the ranking accuracy, there is a tremendous need for maintaining the trade-off between the diversity and accuracy in the top-N recommendation list. Thus, the proposed approach constructs the movie-centric graph along with the effective utilization of LtR method, which facilitates the semantic-path based user preference identification and also maintains the trade-off between the diversity and accuracy.

The main contributions of the Diversity-Ensured Semantic-aware Item REcommendation (DESIRE) approach are as follows.

- The DESIRE approach combines the latent user preferences from the LOD and user rating information through generating two graphs such as movie-centric and user rating-centric graph.
- By exploring user rating-centric graph, the DESIRE approach identifies the semantically relevant path between the users and movies and determines the inherent user preferences on movies.
- Instead of applying the prediction based ranking method, the DESIRE approach directly focuses on the ranking method to effectively suggest the top-N movies through the learning to rank algorithm.
- Moreover, the DESIRE approach addresses the constraint of maintaining the trade-off between the diversity and accuracy through optimally re-ranking the list of movies with the help of learning to the ranking algorithm.

- The experimental results demonstrate that the DESIRE approach significantly outperforms the existing approach to create the diverse as well as a relevant set of top-N movie recommendations.

1.1 Paper organization

The remainder of this paper is structured as follows. Section 2 reviews the previous works related to the semantic recommender systems. The proposed semantic movie recommender system is presented in Section 3. Section 4 describes the experimental evaluation to test the performance of the DESIRE approach. Finally, Section 5 presents the conclusion of the presented approach.

2. Literature survey

Several previous approaches have been attempted to incorporate the semantic information in the recommender systems, especially LOD. This section reviews the previous works and the most recent efforts on the semantic recommender systems and the ranking based recommender systems.

2.1 Semantic recommender systems

With the intention of addressing the cold-start problem in the collaborative recommender systems, an approach in [9] develops the first prototype of the open recommender system using the linked data. The knowledge-based framework in [10] leverages the DBpedia to compute the cross-domain recommendations. Several researchers [11, 12] have presented a LOD based mode-based and memory-based methods to support the content-based recommendations. To match the graph-based item representations, LOD content-based recommendation [13] suggests a recommendation models based on the neighborhood-based graph kernel. Context-aware movie recommendation model [14] is based on the DBpedia source, presents a movie recommendation tool for mobile applications. A hybrid approach [15] has been presented top-N recommendation model based on the implicit feedback information with the assistance of linked data sources. An approach [16] addresses the cold-start issues with the items through a vertex kernel that helps to get the knowledge about the unrated semantic categories. Contextual eVSM model in [17] exploits the LOD to facilitate the content-based recommendation system along with the adoption of semantic representation through

distributional models and entity linking methods. Collaborative recommendation system [18] employs the dynamic content based filtering, association rules, and opinion mining to monitor the dynamically changing user behaviors on the shopping site. With the intention of identifying the current market trend, it applies the association rule mining on the items that are preferred by the users. Moreover, it facilitates the determination of item popularity among the users based on its weighted opinion miner with respect to the customer reviews. Hybrid collaborative movie recommender system [19] improves the recommendation quality by reducing the scalability issue with the support of enriched clustering method such as Fuzzy C Means Clustering (FCM) with Bat optimization. It easily identifies the cluster centroid and nearest neighbors of the active user due to its Bat optimization method and Pearson correlation method respectively. A hybrid recommender system [20] exploits k-means clustering algorithm with bio-inspired Artificial Bee Colony (ABC) optimization method ensures higher movie recommendation accuracy, reliability, and personalization while testing on the Movielens dataset. By applying the dimensionality reduction and ontology techniques, collaborative filtering based recommender system resolves the sparsity and scalability related issues while ensuring the recommendation accuracy. It employs the Singular Value Decomposition (SVD) method as the dimensionality reduction technique to determine the most similar users and items in the clustered users and items [21].

2.2 Ranking based recommender systems

In recent years, ensuring top-N recommendation is the critical issue in the context of collaborative filtering. SLIM [22] employs a sparse linear method to learn a sparse aggregation coefficient matrix and to compute the top-N recommendations. A novel collaborative ranking method [23] optimizes the NDCG ranking metric during the recommendation through matrix factorization. Several existing research works such as CofiRank [24], CLiMF [25] and TFMAP [26] relies on the list-wise LtR based Collaborative filtering method and also focuses on the top-N recommendation items [27]. A collaborative filtering algorithm, CLiMF [25] directly optimizes the evaluation metric of Mean Reciprocal Rank while assessing the recommendation process. Context-aware movie recommendation [28] has presented a social, temporal collaborative ranking based matrix factorization to ensure the time-aware

recommendation based on the explicit and implicit feedback of the users. To tackle the unexpectedness in the observed data, the serendipitous personalized ranking method suggests the effective recommendation with the help of matrix factorization and improves the serendipity as well as recommendation accuracy [29]. LambdaMART (LMART) [30] is an extended ranking model of LambdaRank by exploiting the boosted tree optimization method of MART. This listwise approach employs the derivative estimation of NDCG during training of LtR algorithm to assign the parameters of regression trees. Learning to rank model based recommender system [31] utilizes the trust and distrust relationships of the users and presents the most relevant items of the users and their friends at the top of the list. It presents the weighting strategy and captures the correlations between the user preferences based on the trust of the friends and distrust of the foes. Focused learning technique [32] improves the recommendations in terms of prediction accuracy for the cold-start items based on the objective of the customized matrix factorization and hyper parameter optimization. A Joint Representation Learning (JRL) framework [33] enhances the top-N recommendation with several heterogeneous information sources such as review text, numerical rating, product image, and so on. It can learn a complex prediction network for fast online calculation in the recommendation model. Pairwise Factored Mixed Similarity Model (P-FMSM) captures the locality of the interactions between the users and items based on the predefined symmetric similarity, computes the global correlations across the items based on the asymmetric learned similarity, and understand the uncertain implicit feedback based on the pairwise preference assumption [34]. Currently, the researchers [35, 36] focus on recommending the items with diversity and exploit various evaluation metrics to measure the recommendation performance in the aspect of accuracy as well as diversity. Semantic-path-based ranking (SPrank) [37] provides atop-N recommendation of movies using learning to rank algorithm along with the support of LOD source. However, it fails to ensure the diversity in the top-N recommendation list. Modelling the top-N ranking methods and recommending the items with the optimal level of dependencies within the recommendation list are the significant issues. Moreover, these research works lack in recommending the top-N items with the increased diversity for each user since the consideration of increased diversity for each user

alone can ensure the balance between the accuracy and diversity.

3. An overview of the proposed methodology

Owing to the overwhelming amount of information, it is difficult for the users to determine the new pertinent movies matching the individual preferences. Recommender systems suggest the potentially interesting items to the users to overcome this issue. It is obtained by analyzing the user preferences through the feedback information about the users on the items and computing the similarity between specific user preference and the item. Although, the conventional recommender systems often meet the data sparsity issue. The semantic techniques overcome this problem through the support of graph-based knowledge representation and the heterogeneous data integration. Several prior semantic movie recommender systems employ the LOD source to attain the benefits from the aggregated knowledge sources in various aspects while computing recommendations. However, inferring the preferences of the users without predicting the user ratings is a challenging task due to the existence of multiple categories in the LOD. Hence, the proposed semantic movie recommender system intends to build two specialized graphs such as movie-centric graph using LOD and user rating-centric graph using user ratings to recommend the top-N movies with the diversity directly.

The DESIRE approach incorporates two main phases such as building the LOD based movie-centric graph and movie-centric graph based Top-N movie recommendation. It employs the LOD as the primary source for semantically finding the preferred path and recommending the top-N movies to the users. It enhances the ranking algorithm to recommend the most desired movies to the users effectively. The proposed model builds two semantic graphs to facilitate the movie recommendation along with the diversity ensured top-N movie recommendation.

Building the LOD based movie-centric graph:

Initially, the DESIRE approach applies the semantic distance on the LOD source based on the movie category to extract the movie relevant data in a linked graph structure. Thus, it comprises the movie-centric graph including a variety of primary and subcategories of movies. For instance, the main category represents the crime, thriller, and comedy category and the subcategories refer to the names of the starring, director, and producer.

Movie-centric graph based Top-N recommendation:

The DESIRE approach separately builds the user rating-centric graph from the movie-centric graph by semantically capturing the user-preferences on movies from the rating information. This constructed user rating-centric graph facilitates the recommender system to find the user preferences. Then, the DESIRE approach inherently determines the semantic-path relevant list of movies and suggests the Top-N movies to the users using learning to rank method while ensuring the trade-off between diversity and accuracy.

4. The proposed methodology

The DESIRE approach employs the LOD source and the explicit feedback of the user to build two different graph structures to precisely suggest the list of top-N movies to the users. Recommending the desired movies of the DESIRE approach rely on the two phases such as the contextual path exploration in a semantic graph and multi-relational graph structure based personalized ranking. Figure 1 shows the overall process of the proposed methodology.

4.1 Building LOD based movie-centric graph

The DESIRE approach exploits the LOD to support the semantic movie recommendation owing to the extensive offering of the structured data in multiple domains. The prime advantage of the LOD is that broadly comprises the open datasets including multi-domain information with their relations. Moreover, its standard interfaces ease the task of recommender systems while analyzing the inherent knowledge and also eliminate the need for processing the additional raw data. The most significant benefit of accessing such ontological schema based entities is that they are dynamically augmented, synthesized, and enriched. LOD contains the relational data related to different domains such as music, geographic locations, art, movies, general common-sense knowledge, and facts. Wherein, the availability of encyclopedia datasets includes DBpedia and Freebase, incorporating a huge amount of factual knowledge. Even though its ontological schema provides the semantically relevant data, exploring semantic-path from a vast and multi-domain knowledge graph is a difficult task when there is a need for a single domain relevant data alone such as 'movie.' Hence, the DESIRE approach targets to build the movie-centric graph from a large graphical representation of LOD.

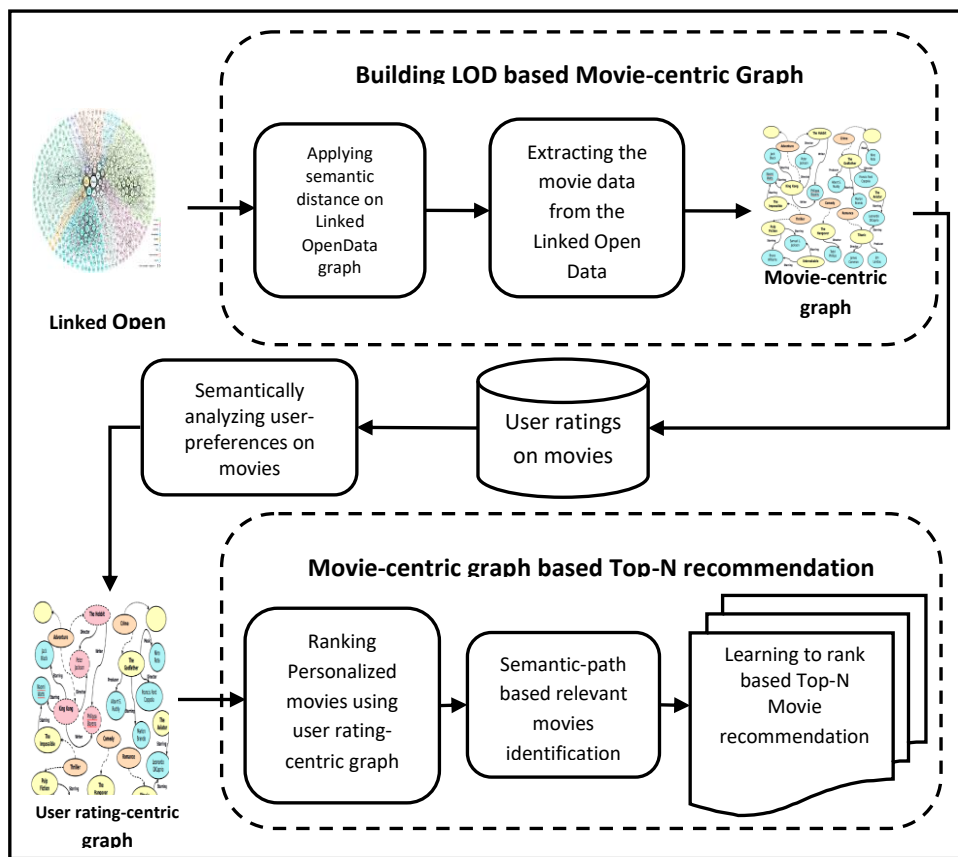


Figure. 1 LOD based Semantic movie recommendation

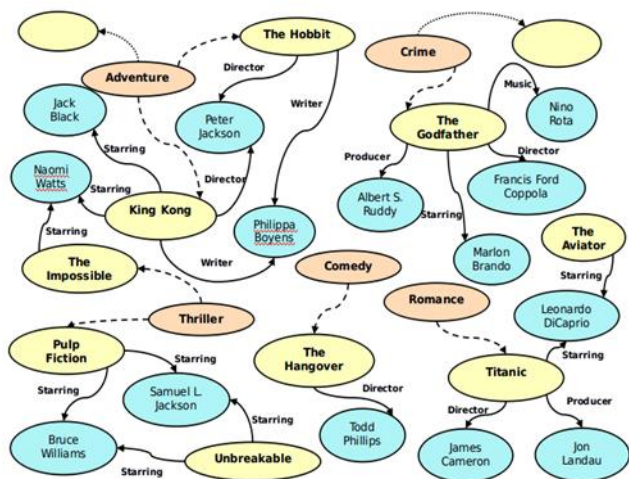


Figure. 2 Movie-centric graph

By applying the movie features based semantic distance on the LOD graph, the DESIRE approach acquires a part of graph structure such as movie-centric graph with multi-relational data. The prerequisite of the DESIRE approach is the utilization of movie-centric graph, wherein the construction of the movie-centric graph is an offline process of recommendation. Although, the constructed movie-centric graph has the updated knowledge in terms of entities and properties, is independent of the recommendation process. To

build a movie-centric graph, the DESIRE approach inherently analyzes the movie database to retrieve the graph structure within the optimal range accurately. The optimal range of movie-centric graph is based on the relational links on the list of movies. Figure 2 demonstrates the portion of the Movie category in the LOD graph. The DESIRE approach effectively utilizes constructed graph structure with the help of user-specific rating information to ensure the personalized recommendation of the movies.

4.2 Semantically creating user rating-centric graph

With the need of semantic representation in the field of recommender systems, the DESIRE approach targets to build the user semantic model in the form of user rating-centric graph through inferring the semantic preferences from the ratings of the users and inherent features of the movie. The representation of the user rating-centric graph is based on the users-features matrix providing latent user ratings for the features of the movies. By accessing the information in the LOD based movie-centric graph, the DESIRE approach effectively extracts the movie-features and constructs the

knowledge graph. The constructed knowledge graph refers the user rating-centric graph which is the resemblance of the LOD design, comprising the feature-values as the nodes that are semantically connected to each other via features like the edges.

4.2.1. Extracting latent user preferences from user feedback

To semantically create the user rating-centric graph, there is an essential need of extracting the latent features of the movies for each user. Accordingly, the DESIRE approach focuses on the development of the movie representation based on the user feedback for recommender systems. To accomplish this objective, the proposed system initially maps the term of movie name with the constructed movie-centric graph and extracts the latent features which are linked with a specific movie in a one-hop distance. By applying string similarity measure, the system tends to determine the co-occurring movie names in the movie-centric graph. With the intention of creating user rating-centric graph, the DESIRE approach has separately acquired a list of movies that are rated by each user. The acquired user-specific list of movies facilitates the system to generate the user-rating centric graph. According to the interests, users provide the rating values as 1, 2, 3, 4, and 5 on the movies. Wherein, the DESIRE approach filters the movies having high rating value which is assigned by the threshold value (Ra). The threshold value refers the rating value 3 or 3.5. If the movie rating score is higher than the rating threshold (RS>Ra), the DESIRE approach takes that movie into the consideration of corresponding user (Ui) preferences list. From the retained movie list, determining the co-occurring features are necessary for each user in which co-occurring features indicate either main class of movies such as comedy, and thriller or other features (fn) such as direction, and acting.

Fig. 3 illustrates the user rating-centric graph wherein, the feature-values in dotted lines represent that two movies such as ‘King Kong’ and ‘The Hobbit’ have the common pair of feature-values such as ‘Peter Jackson’ and ‘PhillippaBoyens.’ It tends to the computation of the Co-occurrence Score between these two feature-values for the corresponding user. To determine the co-occurring movie features for each user, the DESIRE approach measures the Semantic-path Score (SPS) between the two feature-values in a user rating-centric graph.

$$SPS(f_1^V, f_2^V)_{U_i} = \omega^P \times \sum_{l=1}^L |P(f_1^V, f_2^V)^l| \quad (1)$$

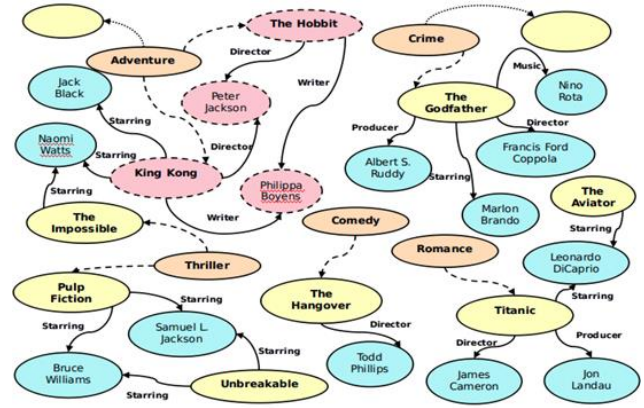


Figure. 3 User rating-centric graph

In Eq. (1), ‘L’ denotes the length of the path referring the total number of movies having both the feature values f_1^V and f_2^V , where, $l \in L$. $|P(f_1^V, f_2^V)^l|$ represents that the importance score of a pair of feature-values in the corresponding movie of the user with respect to the explicit rating score and ω^P indicates the damping factor, $0 < \omega \leq 1$. By applying Eq. (1), the semantic-path score is determined for all the combination of a pair of feature-values that are identified within the user rating-centric graph alone. Subsequently, the DESIRE approach computes the score for all the feature-value pairs according to the semantic connectivity to reduce the feature-value pairs. After filtering a set of feature-value pairs for a specific user ($\{(f_1^V, f_2^V)\}_{U_i}$), the DESIRE approach computes the co-occurrence score on the user rating-centric graph to determine the user preferences related to the latent features. Each feature (f_n) consists of numerous feature-values (f_n^V). For instance, feature ‘director’ has the feature-values of ‘Peter Jackson’ and ‘James Cameron’. The Co-occurrence Score (CS) between the pair of features (f_1^V, f_2^V) is computed with the three cases regarding the frequency of the feature-values, as

$$CS(f_1^V, f_2^V)_{U_i} = \begin{cases} 0, & \text{if } freq(f_1^V) = 0 \text{ or } freq(f_2^V) = 0 \\ 1, & \text{if } freq(f_1^V) = freq(f_2^V) = freq(f_1^V, f_2^V) = freq(f_1^V, f_2^V)_G \\ \eta(f_1^V, f_2^V) \times \left[\frac{freq(f_1^V, f_2^V)}{freq(f_1^V, f_2^V)_G} \right], & \text{Otherwise} \end{cases} \quad (2)$$

Where,

$$\eta(f_1^V, f_2^V) = \frac{\log(freq(f_1^V, f_2^V))}{\log(freq(f_1^V))} \cdot \frac{\log(freq(f_1^V, f_2^V))}{\log(freq(f_2^V))} \quad (3)$$

Where $freq(f_1^V)$ refers the number of movies of f_1^V feature-value appeared in the user rating-centric graph ($G(U_R)$). $\eta(f_1^V, f_2^V)$ represents the weighted score of a specific user regarding the particular pair of features. $freq(f_1^V, f_2^V)$ and $freq(f_1^V, f_2^V)_G$ denotes the

number of movies that contain the presence of both the f_1^V and f_2^V features in the user rating-centric graph and the movie-centric graph respectively. $freq(f_1^V, f_2^V)_G$ measurement computes the total number of movies which are all comprising the occurrence of the pairs in the movie-centric graph. By applying Eqs. (2) and (3), each user has the different score for their set of feature-value pairs. According to the user rating, the DESIRE approach effectively extracts the latent preferences of the users on the movies.

4.3 Recommending Top-N movies

The ultimate of the recommender systems is to suggest a list of items to the users. Accordingly, with the assistance of the extracted latent preferences of each user, the DESIRE approach recommends a list of desired movies to the users. It directly concentrates on finding the top-ranked list of movies for each user rather than accurately predicting the ratings on movies. The DESIRE approach effectively applies the ranking algorithm based on the semantic connectivity in user rating-centric graph.

4.3.1. Ranking Top-N movies from the list of user preferences

The DESIRE approach learns the ranking function by applying the learning to rank techniques to automatically build a ranking model with the help of training data through machine learning techniques. It assists in sorting the new movies according to the user preference. The personalized ranking method targets to provide a list of movies to the corresponding user with a high level of satisfaction. The DESIRE approach investigates the latent preferences of each user on the movies to ensure the personalized top-N movie recommendation. By analyzing the user rating-centric graph, the DESIRE approach extracts the semantically most relevant features based on the path to describe the user-movie interaction list. After extracting the relevant path, it applies learning to a rank method to compute the top-N movie recommendation.

The primary objective of the DESIRE approach is to recommend the movies to the users through the analysis of their latent features expressed in the semantic graph. The multi-relational nature of the data between the path of users and movies tends to inherently compute a distinct relevance score of each path for the user. For instance, the intention behind the preferences of a user relies on the specific actors. In this case, starring relation path based movie selection is more beneficial to the user.

Hence, there is an essential need of recommending the movies based on the discriminative features of the user. According to the sequence of relations in the user rating-centric graph, the DESIRE approach extracts the rich feature-value pairs for every user and consequently, entrusts to the learning to rank algorithm with the most relevant path information for each user. By utilizing Eq. (4), the training data of user-movie pairs with the relevance score is computed, and it is given to the learning to rank algorithm of the LambdaMART algorithm. The DESIRE approach employs the LambdaMART algorithm to generate a list of the top-N recommendation of movies.

$$W_{U_i, M_j} = \left[\frac{N(P)_{i,j} - N_{min}(P)_{i,k}}{N_{max}(P)_{i,k} - N_{min}(P)_{i,k}} \right] \times \left[\sum_{(f_1^V, f_2^V)_{U_i=1}}^m \beta \times CS(f_1^V, f_2^V)_{U_i} \right] \tag{4}$$

Eq. (4) calculates the weighted score of each user on each movie based on the relationship between the user and movie, and co-occurrence score between a pair of feature-values in the movie. In equation (4), $N(P)_{i,j}$ refers the total number of the path between the i^{th} user and j^{th} movie. $N_{min}(P)_{i,k}$ and $N_{max}(P)_{i,k}$ denotes the minimum and a maximum number of paths between i^{th} user over k^{th} movies respectively. Wherein, ‘ k ’ implies the total number of highly rated movies expressed in the corresponding user rating-centric graph. ‘ m ’ represents the number of the feature-value pairs belongs to the i^{th} user **interest list**, $m \in n$. ‘ β ’ is the indicator that will be ‘1’, if the j^{th} movie contains the corresponding feature-value pair. Otherwise, it will be ‘0’. Accordingly, the DESIRE approach suggests a list of top-N movies for each user with the help of LambdaMART learning to rank algorithm.

4.3.2. Ensuring diverse ranking of Top-N movies

To avert the overfitting, the DESIRE approach further focuses on ensuring the diversity of the movies in the list of top-N recommendation. It measures the novelty of the movies in the personalized recommendation list. To ensure the diversity of suggested movies, the DESIRE approach reconstructs the list of top-N movies with the optimal diversity level among the movies. The DESIRE approach utilizes the whole set of movies with the personalized ranking score during re-ranking or reconstructing a list of movies to provide the top-N recommendation alone. If a list of suggested movies does not comprise the optimal diversity score ($Diversity(M_L^i)$), the DESIRE

approach performs the re-ranking within the threshold ranking score ($Rank_a$) through the learning to rank algorithm. It delivers the diversity ensured training set to the LambdaMART algorithm, wherein the training set includes the preference score of the user-movie pair, rich feature-value pairs, and the optimal diversity-accuracy trade-off. The optimal diversity-accuracy score level includes two threshold levels of ranking score such as user-specific ranking threshold and $Rank_a$. User-specific ranking threshold ($Rank_a^i$) is based on the optimal ranking score of any movie in the recommendation list, ($Rank_a > Rank_a^i$) and $Rank_a^i \in [Rank_a, Rank_{max}^i]$. Optimal ranking score denotes that the point of the score when all the rich feature-value pairs based movies are involved in the top-N recommendation list.

$$Diversity(M_{RL}^i) = \left[\frac{(f_1^V, f_2^V)_{U_i^m} \cap (f_1^V, f_2^V)_{M_{RL}^i}}{(f_1^V, f_2^V)_{U_i^m}} \right] \times \left[\frac{\sum_{RL=1}^N D(RS_{M1}, RS_{M2})_{RL}}{N} \right] \quad (5)$$

In Eq. (5), the first term refers to the importance of the user-preferred feature-value pairs in the recommendation list and the second term denotes the computation of the diversity among the list of movies. $(f_1^V, f_2^V)_{U_i^m}$ and $(f_1^V, f_2^V)_{M_{RL}^i}$ refers a set of feature-value pairs in the preference list of i^{th} user and a set of feature-value pairs involved in the suggested movies respectively. $D(RS_{M1}, RS_{M2})_{RL}$ represents the distance between the two consequent pair of movies in the recommendation list in which distance measurement is based on the ranking score, wherein RS_{M1} and RS_{M2} refer the ranking score of $M1$ and $M2$ movie respectively. ‘N’ indicates the total number of recommendations in terms of a total number of movies in the Recommendation List (RL). According to the proposed definition, even when there are less than ‘N’ movies above $Rank_a^i$, the DESIRE approach initially recommends all those movies and then, recommends the remaining movies until reaching top-N recommendation based on W_{U_i, M_j} . This process provides a fair performance when comparing the DESIRE approach with the baseline techniques in the experimental analysis point of view.

5. Experimental evaluation

To exemplify the recommendation performance of the DESIRE approach, this paper compares the proposed algorithm with the SPrank approach [37] through the evaluation with the Movielens dataset.

The evaluation framework considers that the existing approach employs the LambdaMart learning to the rank algorithm to provide the top-N recommendation.

5.1 Experimental setup

The experimental evaluation of the proposed and the comparative approaches are conducted on Linux Ubuntu 12.04 LTS 64-bit machine with a 2.9 GHz Intel CPU and 8GB memory. It runs the experiments using the Java version 1.8.0 from OpenJDK and employs the LOD to generate the semantically associated graph structure.

5.1.1. Dataset

To evaluate the movie recommendation approaches, the evaluation framework utilizes the data from the Movielens dataset [38]. This dataset contains 27278 movies and 20000263 ratings from 138493 users in which rating in the range from 1 to 5. Moreover, the evaluation model exploits the LOD cloud to extract the inherent features of the movies.

5.1.2. Evaluation metrics

The evaluation model employs both the binary and graded metrics, wherein precision and recall are binary metrics that rely on the measurement of relevance or irrelevance and nDCG consider the graded values in terms of the ranked position of movies.

Precision: It is the ratio between the number of top-N recommended movies that are relevant to i^{th} user and the number of top-N recommended movies for the i^{th} user.

Recall: It is the ratio between the number of top-N recommended movies that are relevant to i^{th} user and the number of relevant observed movies for the i^{th} user.

Normalized Discounted Cumulative Gain (NDCG): It mainly relies on the ranked position of movies in the top-N recommendation list. ‘k’ refers the position of movies for the i^{th} user in the recommendation list and IDCG denotes the ideal value of DCG.

$$NDCG = \frac{DCG}{IDCG} \quad (6)$$

Where,

$$DCG = \sum_{j=1}^N \left(\frac{2^{rel_{ij}-1}}{\log_2(1+k)} \right) \quad (7)$$

Diversity: It refers to the greater dissimilarity of all the movie pairs in the recommendation list of i^{th} user.

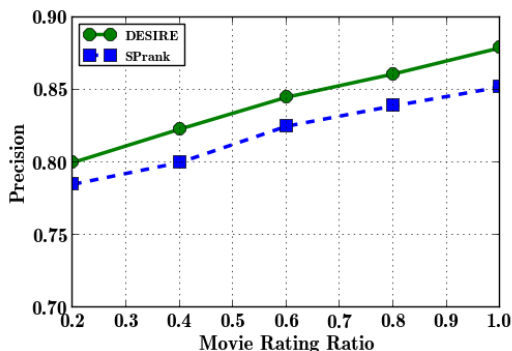


Figure. 4 Movie rating ratio vs. precision

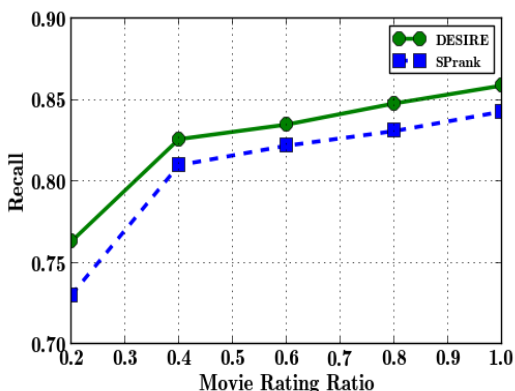


Figure. 5 Movie rating ratio vs. recall

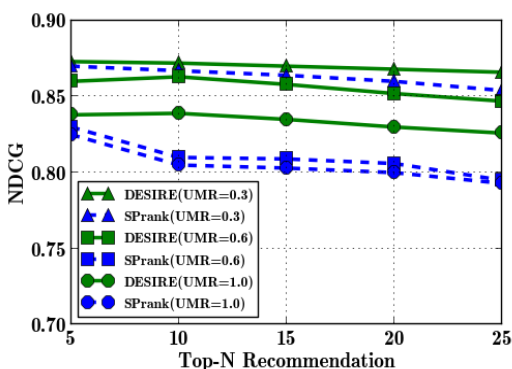


Figure. 6 Top-N recommendation vs. NDCG

5.2 Experimental results

5.2.1. Movie rating ratio vs. precision

Fig. 4 shows the precision of both the DESIRE and the existing SPrank approach while varying the Movie Rating Ratio from 0.2 to 1.0. The precision value increases with the increase of Movie Rating Ratio. Accordingly, the DESIRE approach linearly increases the precision value by 1.34% than the existing SPrank approach while increasing the Movie Rating Ratio from 0.2 to 1.0, which indicates that it suggests the desired movies to the corresponding user concerning the knowledge of rated movie features. It is because the DESIRE approach explores the intended feature-value pairs behind the rating value and selects the rich feature-

value pairs for each user with the help of LOD knowledge source. Moreover, the DESIRE approach applies the rich feature-value pair based learning to the therank algorithm to recommend the top-N movies to the corresponding users. Thus, the DESIRE approach achieves 80% of precision value even when the movie rating ratio is minimum as 0.2 but, the SPrank approach only obtains 78.5% of precision value.

5.2.2. Movie rating ratio vs. recall

Fig. 5 compares the results regarding the recall of the DESIRE approach and the existing SPrank approach with the variation of Movie Rating Ratio from 0.2 to 1.0. The DESIRE approach considers that the feature-value pair extraction is necessary to find the original intention of the users. The DESIRE approach achieves higher recall value by 4.52% than the existing SPrank approach when the movie rating ratio is 0.2. It is because, the DESIRE approach constructs two levels graphs such as movie-centric graph and user-rating centric graph, which assists to inherently determine the most personalized interest of each user. It extracts the inherent features and recognizes the feature-value pairs instead of considering the features alone using the interlinked resource of the LOD source. Thus, the proposed and existing approaches obtain the quite closer recall value with a monotonic difference by 1.97% when the movie rating ratio is 0.4. However, the SPrank approach performs poorly in ensuring the diversity ensured movie recommendation when compared to the DESIRE approach.

5.2.3. Top-N recommendation vs. NDCG

Fig. 6 shows the performance of NDCG value for both the proposed and existing approaches when varying the number of top-N recommendations from 5 to 25 and varying the unrated movie ratio as 0.3, 0.6, and 1.0. Unrated Movie Ratio (UMR) is the ratio between the number of unrated movies and the total number of movies. The NDCG value decreases while increasing the UMR. When UMR=0.3, the DESIRE approach decreases its NDCG value by 0.8% even when increasing the top-N recommendation list from 5 to 25. However, at the same scenario, the existing SPrank approach suddenly degrades its performance by 1.83% due to the lack of feature-value pair and diversity score consideration during ranking and re-ranking respectively. Moreover, the DESIRE approach accomplishes the great NDCG value even when there is the maximum value of UMR. Accordingly, it obtains 3.98% higher NDCG value than the SPrank

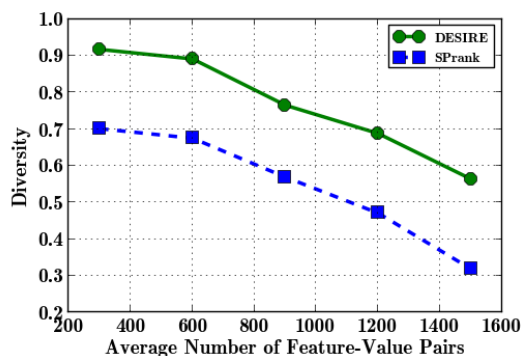


Figure. 7 Number of feature-value pairs vs. diversity

approach when the number of top-N recommendation list is 15, and unrated movie ratio is 1.0. It is because, the DESIRE approach optimally reduces the number of feature-value pairs for each user based on the LOD source based feature extraction and user-rating specific intention, which averts the inaccurate movie recommendation. When UMR is 0.6, the DESIRE approach attains the NDCG value as 0.847 which is quite close to the NDCG value of the SPrank approach when UMR=0.3 at the point of 25 top-N recommendation list.

5.2.4. Average number of feature-value pairs vs. diversity

The diversity of both the DESIRE and the SPrank approaches are depicted in Fig. 7 with the variation of the number of feature-value pairs per user. When the average number of feature-value pairs varies from 300 to 1500, the DESIRE approach obtains the optimum diversity and then, it attempts to maintain the optimal diversity value in the top-N recommendation list. When the average number of feature-value pairs is 300, the DESIRE approach decreases diversity value by 38.49%. In contrast, the SPrank approach rapidly decreases the diversity by 54.49% when increasing the average number of feature-value pairs from 300 to 1500, which seems the DESIRE approach recommend the diversity-ensured movies with the consideration of an optimal average number of feature-value pairs alone. Thus, the DESIRE approach maintains the optimal diversity even when increasing the number of feature-value pairs due to the consideration of the rich feature-value pair based re-ranking of top-N movies concerning the ranking score of each movie.

6. Conclusion

This paper has presented the DESIRE approach, a top-N movie recommendation algorithm able to precisely exploit the substantial knowledge from the

LOD source with the support of user rating information. With the intention of reducing the time complexity during recommendation, the DESIRE approach initially builds the movie-centric graph in offline mode. The DESIRE approach enables the extraction of inherent path based preferences between the users and movies from the constructed user rating-centric graph. The ranking and recommendation problem is formulated through the personalized ranking and diversity enabled re-ranking in learning to the ranking algorithm. The DESIRE approach is modelled to improve the performance of top-N movie recommendations by keeping both the accuracy and diversity at an optimal level. The experimental results demonstrate that the DESIRE approach offers the significant improvements in terms of 30.81% higher diversity over the existing SPrank approach in Movielens dataset. In future, this work will be extended to sensitively satisfy the current taste of the users over the dynamically changing users' interest day-by-day. Furthermore, the future work will focus on recommending the millions of items to millions of users in optimal computational time and power with the support of scalability.

References

- [1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions", *IEEE transactions on knowledge and data engineering*, Vol.17, No.6, pp.734-749, 2005.
- [2] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey", *Knowledge-based systems*, Vol.46, pp.109-132, 2013.
- [3] W. Carrer-Neto, M.L.Hernández-Alcaraz, R. Valencia-García, and F. García-Sánchez, "Social knowledge-based recommender system. Application to the movies domain", *Expert Systems with applications*, Vol.39, No.12, pp.10990-11000, 2012.
- [4] P. Winoto and T.Y. Tang, "The role of user mood in movie recommendations", *Expert Systems with Applications*, Vol.37, No.8, pp.6086-6092, 2010.
- [5] T. D. Noia and V.C. Ostuni, "Recommender systems and linked open data", In: *Proc. of Reasoning Web. Web Logic Rules, 11th International Summer School*, pp.88-113, 2015.
- [6] C. Bizer, T. Heath, and T. Berners-Lee, "Linked data-the story so far", *International*

- Journal on Semantic Web and Information Systems*, Vol. 5, No. 3, pp 1-22, 2009.
- [7] H. Steck, "Evaluation of recommendations: rating-prediction and ranking", In: *Proc. of 7th ACM Conf. on Recommender systems*, pp.213-220, 2013.
- [8] A. Karatzoglou, L. Baltrunas, and Y. Shi, "Learning to rank for recommender systems", In: *Proc. of 7th ACM Conf. on Recommender systems*, pp. 493-494, 2013.
- [9] B. Heitmann and C. Hayes, "Using Linked Data to Build Open, Collaborative Recommender Systems", In: *Proc. of AAAI spring symposium: linked data meets artificial intelligence*, pp.76-81, 2010.
- [10] I. Fernández-Tobías, I. Cantador, M. Kaminskas, and F. Ricci, "A generic semantic-based framework for cross-domain recommendation", In: *Proc. of 2nd International Workshop on Information Heterogeneity and Fusion in Recommender Systems, ACM*, pp.25-32, 2011.
- [11] T. Di Noia, R. Mirizzi, V.C. Ostuni, and D. Romito, "Exploiting the web of data in model-based recommender systems", In: *Proc. of sixth ACM Conf. on Recommender systems*, pp.253-256, 2012.
- [12] T. Di Noia, R. Mirizzi, V.C. Ostuni, D. Romito, and M. Zanker, "Linked open data to support content-based recommender systems", In: *Proc. of 8th International Conf. on Semantic Systems, ACM*, pp.1-8, 2012.
- [13] V.C. Ostuni, T. Di Noia, R. Mirizzi, and E. Di Sciascio, "A linked data recommender system using a neighborhood-based graph kernel", In: *Proc. of International Conf. on Electronic Commerce and Web Technologies, Springer, Cham*, pp.89-100, 2014.
- [14] V.C. Ostuni, G. Gentile, T. Di Noia, R. Mirizzi, D. Romito, and E. Di Sciascio, "Mobile movie recommendations with linked data", In: *Proc. of International Conf. on Availability, Reliability, and Security, Springer, Berlin, Heidelberg*, pp.400-415, 2013.
- [15] V.C. Ostuni, T. Di Noia, E. Di Sciascio, and R. Mirizzi, "Top-n recommendations from implicit feedback leveraging linked open data", In: *Proc. of 7th ACM Conf. on Recommender systems*, pp.85-92, 2013.
- [16] M. Rowe, "Transferring semantic categories with vertex kernels: Recommendations with semanticsvd++", In: *Proc. of International Semantic Web Conf., Springer, Cham*, pp.341-356, 2014.
- [17] C. Musto, G. Semeraro, P. Lops, and M. de Gemmis, "Combining distributional semantics and entity linking for context-aware content-based recommendation", In: *Proc. of International Conf. on User Modeling, Adaptation, and Personalization, Springer, Cham*, pp.381-392, 2014.
- [18] V. Vellaichamy and V. Kalimuthu, "Hybrid Collaborative Movie Recommender System Using Clustering and Bat Optimization", *International Journal of Intelligent Engineering and Systems*, Vol.10, No.5, pp. 38-47, 2017.
- [19] A.S. Tewari and A.G Barman, "Collaborative Recommendation System Using Dynamic Content based Filtering, Association Rule Mining and Opinion Mining", *International Journal of Intelligent Engineering and Systems*, Vol.10, No.5, pp. 57-66, 2017.
- [20] R. Katarya, "Movie Recommender system with metaheuristic artificial bee", *Neural Computing and Applications, Accepted for publication*, 2018.
- [21] M. Nilashi, O. Ibrahim, and K. Bagherifard, "A recommender system based on collaborative filtering using ontology and dimensionality reduction techniques", *Expert Systems with Applications*, Vol.92, pp.507-520, 2018
- [22] X. Ning and G. Karypis, "Sparse linear methods with side information for top-n recommendations", In: *Proc. of sixth ACM Conf. on Recommender systems*, pp.155-162, 2012.
- [23] S. Balakrishnan and S. Chopra, "Collaborative ranking", In: *Proc. of fifth ACM international Conf. on Web search and data mining*, pp.143-152, 2012.
- [24] M. Weimer, A. Karatzoglou, Q.V. Le, and A.J. Smola, "Cofi rank-maximum margin matrix factorization for collaborative ranking", In: *Proc. of 20th International Conference on Neural Information Processing Systems*, pp.1593-1600, 2008.
- [25] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, N. Oliver, and A. Hanjalic, "CLiMF: learning to maximize reciprocal rank with collaborative less-is-more filtering", In: *Proc. of sixth ACM Conf. on Recommender systems*, pp.139-146, 2012.
- [26] Y. Shi, A. Karatzoglou, L. Baltrunas, M. Larson, A. Hanjalic, and N. Oliver, "TFMAP: optimizing MAP for top-n context-aware recommendation", In: *Proc. of 35th International ACM SIGIR Conf. on Research and development in information retrieval*, pp.155-164, 2012.

- [27] P. Cremonesi, Y. Koren, and R. Turrin, "Performance of recommender algorithms on top-n recommendation tasks", In: *Proc. of fourth ACM Conf. on Recommender systems*, pp.39-46, 2010.
- [28] N.N. Liu, L. He, and M. Zhao, "Social temporal collaborative ranking for context aware movie recommendation", *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol.4, No.1, pp.15, 2013.
- [29] Q. Lu, T. Chen, W. Zhang, D. Yang, and Y. Yu, "Serendipitous personalized ranking for top-n recommendation", In: *Proc. of the IEEE/WIC/ACM International Joint Conf. on Web Intelligence and Intelligent Agent Technology*, IEEE Computer Society, Vol.1, pp.258-265, 2012.
- [30] Q. Wu, C.J. Burges, K.M. Svore, and J. Gao, "Adapting boosting for information retrieval measures", *Information Retrieval*, Vol.13, No.3 pp.254-270, 2010.
- [31] D. Rafailidis and F. Crestani, "Learning to Rank with Trust and Distrust in Recommender Systems", In: *Proc. of Eleventh ACM Conf. on Recommender Systems*, pp.5-13, 2017.
- [32] A. Beutel, EH. Chi, Z. Cheng, H. Pharm, and J. Anderson, "Beyond Globally Optimal: Focused Learning for Improved Recommendations", In: *Proc. of 26th International Conf. on World Wide Web*, pp.203-212, 2017.
- [33] Y. Zhang, Q. Ai, X. Chen, and W. Croft, "Joint representation learning for top-n recommendation with heterogeneous information sources", In: *Proc. of International conference on Information and Knowledge Management*, pp.1449-1458, 2017.
- [34] M. Liu, W. Pan, M. Liu, Y. Chen, X. Peng, and Z. Ming, "Mixed similarity learning for recommendation with implicit feedback", *Knowledge Based Systems*, Vol.119, pp.178-185, 2017.
- [35] N. Hurley and M. Zhang, "Novelty and diversity in top-n recommendation--analysis and evaluation", *ACM Transactions on Internet Technology (TOIT)*, Vol.10, No.4, p.14, 2011.
- [36] S. Vargas and P. Castells, "Rank and relevance in novelty and diversity metrics for recommender systems", In: *Proc. of fifth ACM Conf. on Recommender systems*, pp.109-116, 2011.
- [37] T.D. Noia, V.C.Ostuni, P. Tomeo, and E.D.Sciascio, "SPrank: Semantic-path-based ranking for top-n recommendations using linked open data", *ACM Transactions on Intelligent Systems and Technology (TIST)*, Vol.8, No.1, p.9, 2016.
- [38] <https://grouplens.org/datasets/movielens/20m/>